

George A. Tsihrintzis
Shiuh-Jeng WANG
Chih-Hung Wang *Editors*



Security and Information Technologies with AI, Internet Computing and Big-data Applications

Proceedings of SITAIBA 2023

Smart Innovation, Systems and Technologies

Volume 410

Series Editors

Robert J. Howlett, KES International, Shoreham-by-Sea, UK

Lakhmi C. Jain, KES International, Shoreham-by-Sea, UK

The Smart Innovation, Systems and Technologies book series encompasses the topics of knowledge, intelligence, innovation and sustainability. The aim of the series is to make available a platform for the publication of books on all aspects of single and multi-disciplinary research on these themes in order to make the latest results available in a readily-accessible form. Volumes on interdisciplinary research combining two or more of these areas is particularly sought.

The series covers systems and paradigms that employ knowledge and intelligence in a broad sense. Its scope is systems having embedded knowledge and intelligence, which may be applied to the solution of world problems in industry, the environment and the community. It also focusses on the knowledge-transfer methodologies and innovation strategies employed to make this happen effectively. The combination of intelligent systems tools and a broad range of applications introduces a need for a synergy of disciplines from science, technology, business and the humanities. The series will include conference proceedings, edited collections, monographs, handbooks, reference books, and other relevant types of book in areas of science and technology where smart systems and technologies can offer innovative solutions.

High quality content is an essential feature for all book proposals accepted for the series. It is expected that editors of all accepted volumes will ensure that contributions are subjected to an appropriate level of reviewing process and adhere to KES quality principles.

Indexed by SCOPUS, EI Compendex, INSPEC, WTI Frankfurt eG, zbMATH, Japanese Science and Technology Agency (JST), SCImago, DBLP.

All books published in the series are submitted for consideration in Web of Science.

George A. Tsihrintzis · Shiuh-Jeng WANG ·
Chih-Hung Wang
Editors

Security and Information Technologies with AI, Internet Computing and Big-data Applications

Proceedings of SITAIBA 2023

Editors

George A. Tsihrintzis
Department of Informatics
University of Piraeus
Piraeus, Greece

Shiuh-Jeng WANG
Department of Information Management
Central Police University
Taoyuan City, Taiwan

Chih-Hung Wang
Department of Computer Science
and Information Engineering
National Chiayi University
Chiayi, Taiwan

ISSN 2190-3018 ISSN 2190-3026 (electronic)
Smart Innovation, Systems and Technologies
ISBN 978-981-97-7785-3 ISBN 978-981-97-7786-0 (eBook)
<https://doi.org/10.1007/978-981-97-7786-0>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2025

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

If disposing of this product, please recycle the paper.

Organization

Honorary Chairs

Lakhmi C. Jain, KES International, UK

Chin-Chen Chang, IEEE Fellow, Feng Chia University, Taiwan

General Chairs

Shiuh-Jeng WANG, Central Police University Taiwan

George A. Tsihrintzis, University of Piraeus, Greece

International Advisory Committee

Nakao Koji, NICT, Japan

Javier Lopez, University of Malaga, Spain

Heung Youl Youm, Soonchunhyang University, Korea

Hiroaki Kikuchi, Meiji University, Japan

Witold Pedrycz, University of Alberta, Canada

Sakurai Kouichi, Kyusyu University, Japan

Sang-Soo Yeo, Mokwon University, Korea

Liudong Xing, University of Massachusetts, Dartmouth, USA

Laurence T. Yang, St. Francis Xavier University, Canada

Binod Vaidya, University of Ottawa, School of Electrical Engineering and Computer Science, Canada

Program Committee Chairs

Chih-Hung Wang, National Chiayi University, Taiwan
Jung-San Lee, Feng Chia University, Taiwan
Taeshik Shon, Ajou University, Korea
Damien Sauveron, University of Limoges, France
Chi-Yao Weng, National Chiayi University, Taiwan

Publicity Chairs

Aneesh Sharma, University of California, Berkeley, USA
Seungmin Rho, Sungkyul University, Korea
Kuo-Jui Wei, AAA Security Technology Co., Ltd., Taiwan
Chin-I Lee, Ling Tung University, Taiwan

Publication Chairs

Chi-Yao Weng, National Chiayi University, Taiwan
Cheng-Hsing Yang, National Pingtung University, Taiwan
Chia-Chen Lin, National Chin-Yi University of Technology, Taiwan

Finance Chairs

Chien-Lung Chan, Yuan Ze University, Taiwan
Jen-Chun Chang, National Taipei University, Taiwan
Cheng-Ta Huang, Yuan Ze University, Taiwan

Local Arrangement Chairs

Sean Hong, KEYXENTIC INC., Taiwan
Chung-Fu Lu, Chihlee University of Technology, Taiwan
Kuo-Yu Tsai, FengChia University, Taiwan
Cheng-Yi Lin, Lydsec Digital Technology Co., Ltd., Taiwan

Organizing Chairs

Chien-Lung Hsu, Chang Gung University, Taiwan

Jongsung Kim, Kookmin University, Korea

Chao-Lung Chou, Chung Cheng Institute of Technology, National Defense University, Taiwan

Program Committees

Winston Seah, Victoria University of Wellington, New Zealand

Chia-Chen Lin, National Chin-Yi University of Technology, Taiwan

Jung-Shian Li, National Cheng Kung University, Taiwan

Geyong Min, University of Exeter, United Kingdom

Wen-Chuan Wu, National Ilan University, Taiwan

Ming-Hung Wang, National Chung Cheng University, Taiwan

Athanasios V. Vasilakos, Lulea University of Technology, Sweden

Hsi-Chung Lin, Aletheia University, Taiwan

I-Ching Hsu, National Formosa University, Taiwan

Chun-Wei Yang, China Medical University, Taiwan

Chia-Wei Tsai, National Taichung University of Science and Technology, Taiwan

Chia-Chun Wu, National Quemoy University, Taiwan

Zhao Yao, Institute of Information Science, Beijing Jiaotong University, China

Bo-Chao Cheng, National Chung Cheng University, Taiwan

Hyunguk Yoo, University of New Orleans, USA

Chi-Chao Chang, Chang Jung Christian University, Taiwan

Liwei Chen, Zhengzhou University, China

Te-Yu Chen, National Tainan Junior College of Nursing, Taiwan

Ken Choi, Illinois Institute of Technology, USA

Chih-Ho Chou, Ministry of Digital Affairs, Taiwan

Rashid Mehmood, King AbdulAziz University, Saudi Arabia

Da-Ren Chen, National Taichung University of Science and Technology, Taiwan

Yen-Wen Chen, National Central University, Taiwan

Richard Chbeir, Universite de Pau et des Pays de l'Adour, France

Cheng-Chi Lee, Fu Jen Catholic University, Taiwan

Hung-Yu Chien, National Chi Nan University, Taiwan

Donghoon Chang, IIIT-Delhi, India

Narn-Yih Lee, National Cheng Kung University, Taiwan

Yujie Wang, University of Electronic Science and Technology, China

Chu-Hsing Lin, Tung Hai University, Taiwan

Ji-Han Jiang, National Formosa University, Taiwan

Taeshik Shon, Ajou University, Korea

Wei-Hua He, Soochow University, Taiwan

Ching-Chun Chang, Department of Computer Science, University of Warwick, Coventry CV4 7AL, UK
Chin-Feng Lee, Chaoyang University of Technology, Taiwan
Yeong-Sheng Chen, National Taipei University of Education, Taiwan
Chun-Ta Li, Fu Jen Catholic University, Taiwan
Yu-Chen Hu, Tung Hai University, Taiwan
Aneesh Sharma, University of California, Berkeley, USA
Jia-Ning Luo, National Defense University, Taiwan
Ali Shahrabi, Glasgow Caledonian University, UK
Jau-Ji Shen, National Chung Hsing University, Taiwan
Wei-Liang Tai, National Taichung University of Science and Technology, Taiwan
Seokjun Lee, Kennesaw State University, USA
Woei-Jiunn Tsaur, National Taipei University, Taiwan
Jonathan Kavalan, University of Florida, USA
Ming-Hour Yang, Chung Yuan Christian University, Taiwan
Ray-Lin Tso, National Cheng-Chi University, Taiwan
Luis Javier Garcia Villalba, Complutense University of Madrid, Spain
Chung-Ming Ou, Kainan University, Taiwan
Hai-Cheng Chu, National Taichung University of Education, Taiwan
Bingwen Feng, School of Cyberspace Security, Jinan University, China
Yu-Chi Chen, National Taipei University of Technology, Taiwan
Shih-Hao Chang, National Taipei University of Technology, Taiwan
Yue-Shan Chang, National Taipei University, Taiwan
Marco Aiello, University of Groningen, Netherlands
Kwok-Yan Lam, Tsinghua University, China
David Naccache, Gemplus, France
Xuejia Lai, Shanghai Jiao Tong University, China
Wei-Jen Wang, National Central University, Taiwan
Ryma Abassi, University of Carthage, Tunisia
Samia Bouzefrane, CEDRIC Lab Conservatoire National des Arts et Métiers, France
Serge Chaumette, LaBRI, University of Bordeaux, France
Emmanuel Conchon, University of Limoges, France
Gerhard Hancke, City University of Hong Kong, Hong Kong
Jose Onieva, University of Malaga, Spain
Henrich C. Pöhls, Institute of IT Security and Security Law, Universität Passau, Germany
Akka Zemhari, LaBRI, University of Bordeaux, France
Stefanos Gritzalis, University of the Aegean, Greece
Pierre-François Bonnefoi, University of Limoges, France
Andrew Kusiak, The University of Iowa, USA
Brahim Hamid, University of Toulouse, France

Preface

It is our Second International Conference on Security and Information Technologies with AI, Internet Computing and Big-data Applications, SITAIBA 2023. Before this conference, we held the four conferences, such as International Conference on Security with Intelligent Computing and Big-data Services (SICBS for short) in the years of 2017, 2018, and 2019, respectively. The First International Conference on Security and Information Technologies with AI, Internet Computing and Big-data Applications (SITAIBA for short) was held in 2021. The proceedings of four conferences, 2017, 2018, 2019, 2021, were published in Springer AISC 733, Springer AISC 895, Springer AISC 1145 series, and Springer SIST vol. 314, respectively. The conference of SITAIBA 2023 was held during 07–09 December 2023 at Chihlee University of Technology, New Taipei City, Taiwan.

SITAIBA 2023 provides a platform for researchers, engineers, academicians as well as industrial professionals from all over the world to present their research results and development activities in Security and Information Technologies with AI, Internet Computing and Big-data Applications. It aims at strengthening the international academic cooperation and communications, and exchanging research ideas.

We received a large number of papers from all around the world. After rigorous peer-review process, where each submission was reviewed by at least two TPC members, we selected 46 papers, covering the topics as follows: Multimedia Security, Network Security, Artificial Intelligence, Learning and Defense, Fuzz Testing and Analysis, Security Application, and Cryptography, Blockchain, and IoT. Moreover, our conference is further enriched by two keynotes. The first keynote speaker is Prof. Koji Nakao, National Institute of Information and Communications Technology, Japan, where the topic is “Research results of IoT/IIoT malware analysis and the countermeasures”. The second keynote speaker is Prof. Heung Youl Youm, Chairman of ITU-T Study Group 17, Security, Korea, where the topic is “Privacy and security for AI/ML including Privacy enhancing technologies”.

Organization of conferences is a hard work. First, we would like to thank our co-organizer: Chuan-Te Ho who is the President of the National Institute of Cyber Security (NICS). Next, we are appreciated all those who contributed to the advisory

committee, the technical program committee, the organizing committee, and the sponsors: Karen Ho who is the Chairperson of Cybersecurity Technology Institute, Institute for Information Industry (III) and Sean Po-Yueh Hung who is the CEO of the KeyXentic Inc. Besides this, there are also a lot of sponsors with the academic event of our ESAM (E-Security Analysis and Management Association) team for their efforts and support in the course of conference preparations. We are grateful to all authors and reviewers for their contributions. Our thanks are due to Springer for their assistance during the preparation phase of the manuscript.

Piraeus, Greece
Taoyuan City, Taiwan
Chiayi, Taiwan

George A. Tsihrintzis
Shiuh-Jeng WANG
Chih-Hung Wang

Contents

Multimedia

**Image Interpolation Utilizing Neighboring Pixels as References
in Conjunction with Multilayer Perceptron Neural Networks** 3
Tzu Chuen Lu, Thanh Nhan Vo, and Biswapati Jana

1 Introduction 3

2 Related Works 5

 2.1 The Enhanced Neighbor Mean Interpolation Technique
 (ENMI) 5

 2.2 Multilayer Neural Networks 6

3 Proposed Method 7

4 Experimental Results 9

5 Conclusion 11

References 12

**Reversible Data Hiding in Encrypted Image Based on Block
and Huffman Encoding** 13
Cheng-Hsing Yang, Chi-Yao Weng, Bo-Han Huang,
and Wei-Qing Huang

1 Introduction 13

2 The Proposed Approach 15

 2.1 Image Encryption 15

 2.2 Information Hiding 19

 2.3 Message Extracting and Image Recovery 19

3 Experimental Results 20

4 Conclusion 21

References 23

Authentication of Image Fidelity in Watermarking-QR-Code

Approving Copyright Systems 25

Hung-Jui Ko, Cheng-Ta Huang, and Shiuh-Jeng WANG

1 Introduction 25

2 Background and Related Works 27

 2.1 QR Code 27

 2.2 Perceptual Hash Algorithm 27

 2.3 Digital Watermarking 27

 2.4 Related Works 28

3 Proposed Method 28

 3.1 Preliminary Phase 28

 3.2 Embedding Phase 29

 3.3 Watermark Extraction and Verification Phase 30

4 Experimental Results and Discussions 31

5 Conclusions 32

References 33

A Zero-Watermarking Image Scheme in Normalized

Cross-Correlation with Robust Copyright Protections 35

Hung-Jui Ko, Cheng-Ta Huang, and Shiuh-Jeng WANG

1 Introduction 35

2 Background and Related Work 36

 2.1 Image Normalization 36

 2.2 Discrete Wavelet Transform 36

 2.3 Singular Value Decomposition 37

 2.4 Arnold Transform 37

 2.5 Related Work 38

3 Proposed Method 39

 3.1 Zero-Watermark Generation Phase 39

 3.2 Zero-Watermark Verification Phase 40

4 Experimental Results and Discussions 41

5 Conclusion 43

References 43

Network Security (I)

A Novel Approach for Detecting and Analyzing Cyber-Attacks

in Cyber-Physical Systems 47

Zhi-Ren Zheng and Bo-Chao Cheng

1 Introduction 48

2 Related Work 49

3 Proposed Method 51

 3.1 Pre-processing 52

 3.2 Autoencoder 52

 3.3 Bi-LSTM System 53

4	Evaluation	55
4.1	Dataset: Smart Grids System	55
4.2	Performance Metrics	56
4.3	Impact of Sequence Length	57
5	Conclusion	57
	References	58
	White-Box Penetration Testing for Hash Collision Attack on Web Applications	59
	Chao-Lung Chou and Hung-Wei Chiang	
1	Introduction	59
2	Literature Review	62
3	The Proposed Method	63
4	Experimental Results	66
5	Conclusions	68
	References	68
	Towards Zero Trust for Financial Sectors: A Proposed Framework on Trust Evaluation	71
	Yu-Chih Wei and Tak-Wai Yu	
1	Introduction	71
1.1	State of Cybersecurity in Financial Sectors	71
1.2	Zero Trust Architecture: A Paradigm Shift	72
1.3	Research Objective	72
2	Related Work	73
2.1	Cyber Threats in Financial Sectors	73
2.2	Cybersecurity in Financial Sectors	73
2.3	Zero Trust Architecture: The Concept and Core Components	74
3	Proposed Framework	75
3.1	Pillars of Trust Evaluation Framework	75
3.2	Environment Monitor Candidate Components	76
3.3	State of ML with Environment Monitor Components	77
4	Security Realization with Zero Trust	78
4.1	Flow Overview	78
5	Framework Evaluation	79
5.1	Participant Selection	79
5.2	Interview Preparation	80
5.3	Semi-structured Interviews	80
5.4	Interview Design	80
6	Limitation and Future Work	81
7	Conclusion	81
	References	82

FiT-DPKI: Decentralized Public Key Infrastructure with Flexibility and Transparency for IoT Networks 83

Ruei-Hau Hsu, Jia-Yin Guo, Bing-Cheng Ke, and Chia-Ho Tan

1 Introduction 83

2 Preliminaries 84

2.1 Bilinear Mapping 84

2.2 Gennaro et al.'s Threshold Signature Scheme 85

3 Related Work 85

4 System Overview 86

4.1 System Model 86

4.2 Threat Model 87

4.3 Security Requirements 87

5 Proposed Scheme 88

5.1 Service Setup 88

5.2 Registration 89

5.3 Query Whether ID Is Registered 90

5.4 Identity Verification 91

6 Security Analysis 91

7 Evaluation 91

7.1 Performance of Threshold Signature 91

7.2 Performance of Three Party Authentication 92

7.3 Comparison of Security Properties 93

8 Conclusion 94

References 94

Revolutionizing Healthcare: Case Studies of AI Algorithms Transforming the Field of Medicine 97

Aneesh Sharma, Rashee Sharma, and Cheng-Hsing Yang

1 Introduction 98

2 Overview 98

2.1 AI Algorithms for Detecting Eye Diseases 99

2.2 Genomic Sequencing and AI in Cancer Research 99

2.3 Machine Learning in Drug Discovery 100

2.4 Robotic Surgery and AI-Assisted Procedures 102

3 Conclusion 103

References 104

Analysis and Detection of Abnormal Transactions on Ethereum 107

Iuon-Chang Lin and Yu-Chun Chan

1 Introduction 107

2 Literature Review 108

3 Proposed Method 110

4 Experimental Results 112

5 Conclusion 116

References 117

Artificial Intelligence (I)

Feasibility Analysis Study on Constructing a Grid Intrusion Detection System Using Semi-supervised Learning Models 121
Chia-Wei Tsai, Jason Lin, Chun-Wei Yang, and Fu-Nie Loo

1 Introduction 122

2 Background 124

3 Proposed Semi-supervised Learning IDS and Experiment 126

4 Conclusion 128

References 128

Securing NLP Systems: A Comprehensive AI-Based Approach 129
Jheng-Jia Huang, Chih Yang, and Wei-Hsueh Wang

1 Introduction 130

2 Related Work 130

3 Overview of the Attack and Defense Techniques 131

3.1 Model Extraction Attacks 131

3.2 Backdoor Attacks 132

3.3 Evasion Attacks 134

4 Conceptual Defense Mechanism for Modern NLP Systems 135

4.1 Core Principles and Design 136

4.2 Techniques Deployment Across NLP System Lifecycle 138

5 Conclusion 139

References 140

A New Scheme Modeling Gym Membership Transactions with NFT Systems 143
Hao-Yu Tsai, Jen-Chun Chang, and Shiuh-Jeng WANG

1 Introduction 143

2 Background 145

2.1 NFT 145

2.2 Smart Contract 146

2.3 IPFS 146

2.4 Ethereum 146

2.5 Traditional Transaction Model 147

3 Proposed Scheme 148

3.1 First-Hand Transactions 149

3.2 Second-Hand Transactions 149

3.3 Bidding 150

4 Discussion 152

5 Conclusion 154

References 154

A Text to Human-Like Speech Using Tacotron-Based TTS Model 157
Wei-Chen Wu, Zhi-Xun Cai, and Chin-Feng Lee

1 Introduction 157

2 Related Work 158

3 Our Model 159

 3.1 Dataset 159

 3.2 Training Process 159

 3.3 Hyperparameter 160

 3.4 Audio Parameters 160

 3.5 Model Architecture 161

 3.6 Dataset–Fine-Tuning 161

4 Analysis 162

 4.1 Training Results 162

 4.2 Viewing of Training Results 163

5 Conclusion 163

References 164

A Novel Malware Classification Using CNN-SVM Deep Learning Method Based on Transfer Learning Architecture 165
Guan-Yu Wang, Hung-Jui Ko, and Wei-Jen Wang

1 Introduction 165

2 Related Work 167

3 Methodology 168

 3.1 VGG16 and Support Vector Machine 168

 3.2 Transfer Learning 168

 3.3 Proposed Model 170

4 Experimental Results 172

 4.1 Dataset Description 172

 4.2 Confusion Matrix 173

 4.3 Performance of the Proposed Model 174

5 Conclusion 175

References 176

Enhancing Project Programming Hour Prediction with Regression Analysis Techniques—A Case Study of Company D 179
Chin-Feng Lee, Shih-Chia Chang, and Jau-Ji Shen

1 Introduction 180

2 Proposed Method 180

 2.1 Introduction to Research Data 180

 2.2 Exploratory Data Analysis (EDA) 183

 2.3 Feature Selection 183

 2.4 Feature Selection Evaluation 184

3 Model Training and Validation 187

4 Model Testing and Evaluation 189

5 Conclusions 191

References 192

Network Security (II)

A New Traceable One-Time Address Scheme Secure Against Privilege Escalation Attack 195

Tse-Jui Chen, Jen-Chieh Hsu, and Raylin Tso

1 Introduction 195

1.1 Our Contribution 196

2 Related Work 197

2.1 Techniques Related to Privacy Protection 197

2.2 Traceability 197

2.3 Privilege Escalation Attack 198

3 Definitions of Partial Security Traceable One-Time Address Scheme Against Privilege Escalation Attack 198

3.1 System Architecture 198

3.2 Security Model 200

4 The Proposed Scheme 201

4.1 Preliminaries 201

4.2 Construction 201

5 Security 203

6 Comparison 205

7 Conclusion and Future Works 207

References 207

An Object-Based Multi-level Authentication Framework for AR e-Book 209

Chia-Chen Lin, Aristophane Nshimiyimana, Wei-Liang Tai, and Morteza Saberikamarposhti

1 Introduction 210

2 Related Works 210

2.1 Framework of AR e-Book 210

2.2 Multimedia Authentication 212

3 Proposed Object-Based Multi-level Authentication Framework for AR: OML-AFAR 213

3.1 Script Generation for an AR e-Book Phase 214

3.2 Authentication Code Generation Phase 214

3.3 Authentication Code Embedding Phase 215

3.4 Verification Phase 216

4 Experimental Results 216

5 Conclusions 219

References 219

A Door Lock System Based on Visual Cryptography 221
Chang-Yu Chao, Jen-Chun Chang, Ching-Cheng Wang,
Chi-Sheng Chen, Hao-Hsuan Chung, Kuan-Han Lu,
and Guan-Ying Huang

1 Introduction 221

2 Related Works 222

2.1 Original Visual Cryptography 223

2.2 Visual Cryptography with Cover 224

3 Our Visual Secret Hiding Algorithm 226

3.1 Algorithm for Visual Cryptographic Door Lock System 227

4 Our Proposed Door Lock System 230

4.1 System Architecture Diagram 230

5 Unlocking Process 230

5.1 The User Scanning the Door Lock Share 231

5.2 The Door Lock Scanning User Share 231

References 232

Automatic APT Attack Reconstruction Supporting Lateral Movement 235
Chun-I Fan, Wei-Chen Wu, Cheng-Han Shie, Hsin-Nan Kuo,
and Bo-Yi Lee

1 Introduction 235

2 Related Works 236

2.1 Network Intrusion Detection Datasets 237

2.2 Datasets for APT Attacks 237

3 The System Architecture 239

3.1 System Design 239

3.2 Environment Overview 239

3.3 Data Generation 240

3.4 Data Collection 241

3.5 Labelling 241

4 Experimental Results 242

4.1 Distribution of Log Quantities 243

4.2 Distribution of TTPs 243

4.3 Comparison and Discussion 246

5 Conclusions 247

References 248

Using Open-Source Intelligence to Archive Criminal Organizations 251
Peter Heusch and Patrick S. Chen

1 Introduction 252

2 Research Model and Design 253

2.1 Graphic Models 253

2.2 Organizational Structure Center 254

2.3 Activities of Criminal Organization 255

2.4 Geographical Center Proliferation 259

2.5	Changes in the Focus of Activities	259
3	Problem Discussion	260
3.1	Transfer of Gang Leadership Centers	260
3.2	Possibility for Suppression of Gangs in the Early Days of Their Birth	261
3.3	Mastering the Communication Situation of Nodes to Monitor the Gang Structure and Center Transfer	261
3.4	Prevention of the Expansion of Bad Gangs	261
4	Conclusion	261
	References	262

FIDO-Based Access Control Mechanism in Named Data

Networking	263

Yeong-Sheng Chen and Wei-Cheng Chien

1	Introduction	263
2	Background and Related Work	264
2.1	Named Data Networking	264
2.2	Name-Based Access Control in NDN	266
2.3	FIDO U2F	268
3	The Proposed Mechanism	268
3.1	System Structure	268
3.2	Access Control List	269
3.3	Registration	269
3.4	KDK Request	271
4	Comparisons with Related Work	272
5	Conclusions	273
	References	273

Artificial Intelligence (II)

Empirical Approach to a Fine-Tuning Using Forgetting in Large Language Models

.....	277

Kaito Taguchi, Yujie Gu, and Kouichi Sakurai

1	Introduction	277
2	Preliminary	278
2.1	Language Model	278
2.2	Perplexity	279
2.3	Training Methods in LLMs	279
3	Memorization and Forgetting in LLMs	279
3.1	Memorization in LLMs	279
3.2	Measuring Memorization	280
3.3	Membership Inference Attack	281
3.4	Forgetting in LLMs	281
4	Our Proposal	282
5	Experiment	282
5.1	Dataset	282

5.2	Fine-Tuning	282
5.3	Setting of Membership Inference Attacks	283
5.4	Hyperparameters for M_1 and M_2	283
5.5	Result	283
5.6	Additional Experiment 1: Reordering of Datasets	285
5.7	Additional Experiment 2: Forgetting Effect from High Memorization	286
5.8	Additional Experiment 3: Adapter Reduction Factor Set to 2	287
6	Conclusion	288
	References	288
Automatic Wound Segmentation with Deep Convolutional Neural Networks		291
Jasmin T. Ngoc Nguyen, Dun-Hao Chang, and Chien-Lung Chan		
1	Introduction	292
2	Related Works	293
3	Methodology	294
3.1	Data Preparation	294
3.2	Model Architecture	297
3.3	Evaluation Metrics	297
4	Experiment and Results	300
5	Discussion and Conclusion	301
	References	302
Prediction of the Prevalence of COVID-19 Using Epidemic Differential Equations and Deep Learning Network		305
Liang Kong, Yanhui Guo, and Chung-Wei Lee		
1	Introduction	306
2	Proposed Model	307
2.1	SIRD Compartmental Epidemiology Process Model	307
2.2	Derivation of Discrete-Time I-Equation	308
2.3	Edge Feature and Latent I_e	310
2.4	Training the Proposed Model	310
3	Experimental Results	311
4	Conclusion	314
	References	315
Russo-Ukrainian War on Facebook: A Temporal and Spatial Study Based on BERTopic Modeling		317
Kai-Yi Tan, Chun-Ming Lai, Yin-Jie Foo, and Yuya Shibuya		
1	Introduction	317
2	Related Work	318
3	Methodology	320
3.1	Data Collecting and Preprocessing	320
3.2	Topic Modeling Analysis	321
3.3	Sentiment Analysis	322

4	Results	323
4.1	RQ1: What Are the Main Topics Underlying Facebook Posts?	323
4.2	RQ2: How Has the Sentiment Distribution on Different Topics Changed over Time?	326
5	Conclusion	328
	References	328

A Performance Study of Selected Machine Learning Techniques for Predicting Heart Diseases 331

Blessing Oluwatobi Olorunfemi, Adewale Opeoluwa Ogunde,
Abidemi Emmanuel Adeniyi, Joseph Bamidele Awotunde,
Agbotiname Lucky Imoize, and Chun-Ta Li

1	Introduction	332
2	Materials and Methods	333
2.1	Data Preprocessing	333
2.2	Model Architecture	334
2.3	Applying Various Machine Learning Algorithms	335
3	Results and Discussion	338
3.1	Performance Evaluation Matrix	338
3.2	Discussion of the Model Performance Comparison	341
4	Conclusion	342
	References	343

Unmanned Aerial Vehicles Detection Using CNN-Transfer Learning 345

John Irungu, Hubert Smith, Jose Cruz, Thabet Kacem,
and Anteneh Girma

1	Introduction	346
2	Related Work	347
3	Methodology	349
3.1	Data Description	349
3.2	Data Augmentation	349
3.3	MobileNet-V2 Transfer Learning Model	350
4	Experimental Results	351
5	Discussion	354
6	Conclusion	355
	References	355

Learning and Defense

Multi-head Temporal Learning for Human Activity Recognition Using Smartphone 359
So Zheng Wei, Pang Ying Han, Khoh Wee How, and Ooi Shih Yin

1 Introduction 360

2 Related Work 360

3 Multi-head Temporal Convolutional Model 362

4 Database 364

5 Experimental Results 364

6 Conclusions 368

References 368

Windows Registry Attack and Defense: Behavior Analysis and Persistence Detection Based on ETW 371
Jian-Xin Chen, Yang-Hsin Chang, Yan-Ling Hwang, Jian-Hong Hunag, Hao-Jyun Wang, and Fu-Hau Hsu

1 Introduction 371

2 Background 373

2.1 Windows Register Editors 373

2.2 Event Trace on Windows 375

3 Related Works 376

4 System Design 377

5 Experimental Results and Analysis 378

6 Conclusion 379

References 379

Device Identification Technology Within the Network Address Translation Devices 381
Jia-Ning Luo and Ying-Yi Yeh

1 Introduction 381

2 System Model 382

3 Device Identification 383

3.1 Identify the Same Device 385

3.2 Proposed Model 385

4 Experimental Design and Results 386

4.1 Identifying Device Types 387

4.2 To Recognize Similar Devices 387

4.3 Experimental Results 388

4.4 Identifying Device Types 388

4.5 To Recognize Similar Devices 390

5 Conclusion 391

References 392

Fuzz Testing and Analysis

Prediction of Processing Parameters Based on Multi-objective Bayesian Optimization Combined with Machine Learning Model for High Frequency Heat Treatment of Metals 395

Chih-Hsueh Lin, Yu-Hsiang Tseng, Kuo-Hsin Hu, and Chia-Wei Ho

1 Introduction 396

2 Machine Learning Models and Hyperparameter Optimization 398

 2.1 Decision Tree [15] 398

 2.2 Random Forest [16] 398

 2.3 XGboost (eXtreme Gradient Boosting) [17] 400

 2.4 Multi-objective Bayesian Optimization for High-Dimensional Search Spaces [20] 401

3 Research Results and Comparisons 402

4 Experimental Results 403

5 Conclusion 405

References 405

Protocol Security Fuzz Testing with Data Augmentation 407

Nain-Fang Tu, Chia-Ling Lu, and Chi-Yuan Chen

1 Introduction 407

2 Related Works 408

 2.1 Bluetooth Low Energy (BLE) 408

 2.2 Generative Adversarial Networks (GANs) 410

3 Evaluation of Bluetooth Low Energy Packets 413

4 SEQ-BAGAN 415

5 Conclusions 416

References 418

Preserving the Higher Coverage Guided of an ICS-Oriented Fuzzy Testing Technique 419

Ying-Chin Chen, Yang-Hong Yi, Chit-Jie Chew, and Jung-San Lee

1 Introduction 419

2 Preliminaries 421

 2.1 The Significant Components of ICS 421

 2.2 The ICS Communication Protocol-Open Platform Communications Unified Architecture (OPC UA) 422

3 The Proposed Framework 423

 3.1 Seed Generation Module 423

 3.2 Mutation Module 424

 3.3 Trace Module 424

4 Experimental Results 424

5 Conclusions 426

References 426

Security Application

Video Access Control Aligned with Health Education Classification 431

Yi-Hui Chen and Cheng-I Hsieh

1	Introduction	431
2	Related Works	432
2.1	Violence Detection	432
2.2	Visual Cryptography	434
3	Methodology	435
3.1	System Design	435
3.2	Violent Area Detection	436
3.3	Image Encryption	437
4	Experimental Evaluation	437
5	Conclusion	440
	References	440

Information and Public Opinion Analysis Platform Based on Social Media 443

Han-Ting Gao, Shang-Yi Yang, Chun-Han Chang, Jung Kuo,
Guan-Yu Chen, Jheng-Jia Huang, and Nai-Wei Lo

1	Introduction	444
1.1	Research Background	444
1.2	Research Motivation	444
2	Theory and Practice	445
2.1	Public Opinion Analysis	445
2.2	Sentiment Analysis	446
3	Research Methods and Procedures	446
3.1	Information System Development	446
3.2	Research Process	447
4	Research Results	447
4.1	Front-End	448
5	Conclusion	451
	References	452

Analysis of the Domain Name of Fraud Investigation Sites with the Lexical Entailment Recognition Methods 453

Yeu-Pong Lai

1	Introduction	454
2	Backgrounds and Related Works	455
3	Proposed Method	457
4	Conclusion	462
	References	462

The Evaluation Approach for Indoor Design Usability Based on Gaussian Mixed Model	465
Hsing-Chung Chen, Yong-Jian Siao, Pei-Yu Hsu, Jhih-Sheng Su, and Yuan-Jung Lee	
1 Introduction	466
2 Background Knowledges and Related Researches	466
3 Experiments and Results	470
4 Conclusions	475
References	476
A Lightweight Deep Leakage Prevention Model with Mixing Multi-gradients	477
Wei-Chun Hsiao and Chih-Hung Wang	
1 Introduction	477
2 Related Work	478
2.1 Convolutional Neural Networks	479
2.2 Deep Leakage Problem	479
3 The Proposed Lightweight Solution	480
4 Experiments	481
4.1 Model Training	482
4.2 Test Accuracy	482
4.3 Deep Leakage Testing	483
5 Conclusions	486
References	487
PP-3FAT Scheme for 5G-IoHT Telemedicine Systems	489
Tzu-Wei Lin	
1 Introduction	489
2 Related Works	490
2.1 Telemedicine Systems	490
2.2 Three-Factor Authentication (3FA)	490
2.3 Chebyshev Chaotic Maps	491
3 Proposed Scheme	491
4 Security Analysis	492
4.1 Preventing Man-In-The-Middle (MITM) Attack	492
4.2 Key Confirmation	492
4.3 Mutual Authentication	493
5 Conclusion	493
References	494
Cryptography, Blockchain & IoT	
Enhanced Lightweight Authenticated Key Agreement Protocol with Privacy Protection for Vehicle Cloud Computing	499
Tian-Fu Lee and Wen-Hsien Wu	
1 Introduction	500
1.1 The System Model of VCC	500

1.2	Related Works	501
1.3	Our Contributions	502
1.4	Organization of Paper	502
2	Preliminaries	502
2.1	Notations	502
2.2	Review of Lee et al.'s Protocol for VCC	503
3	Limitations of the Protocol of Lee et al.	504
3.1	Failure to Impersonation Attacks	504
3.2	Failure to Replay Attacks	505
3.3	Violation of Session Key Security	505
3.4	Failure to the Man-in-Middle Attacks	505
3.5	Maintaince of a Verifier Table	505
4	Enhanced Authenticated Key Agreement Protocol for VCC	506
5	Security and Performance Analyses	507
5.1	Security Analysis of the Enhanced Protocol	507
5.2	Functionality Comparisons	508
5.3	Performance Analysis and Comparisons	509
6	Conclusion	510
	References	510

An Extended Euclidean Algorithm Implementation over Ciphertext Space 511

Po-Wen Chi, Yuan-Pang Lin, Ming-Hung Wang, and Yun-Hsin Chuang

1	Introduction	512
2	Introduction to Fully Homomorphic Encryption	513
2.1	Related Works	513
2.2	Definition	514
3	Our Extended Euclidean Algorithm Implementation Through FHE	515
3.1	Extended Euclidean Algorithm	515
3.2	Our Choice in FHE Schemes	516
3.3	Modulus Operation Implementation	516
3.4	Iteration Issue	517
3.5	Integration	518
4	Performance Evaluation	519
4.1	Experimental Environment	519
4.2	Computational Time Versus Different Data Sizes	519
4.3	Computational Time Versus Different Key Sizes	520
5	Applications: Proxy Re-encryption	520
6	Conclusion and Future Works	522
	References	522

Cryptanaylsis to Ciphertext-Policy Attribute-Based Encryption of Liu et al. and Amirthavalli et al. 525

Yi-Fan Tseng, Wen-Yi Chen, You-Qian Chen, and Jheng-Jia Huang

1	Introduction	525
2	Preliminaries	526

2.1	Notations	526
2.2	Ciphertext-Policy Attribute-Based Encryption	527
2.3	Access Structure	528
3	Review on Pairing-Free Attribute-Based Encryption Schemes	528
3.1	Liu et al.'s Pairing-Free ABE Scheme	529
3.2	Amirthavalli et al.'s Pairing-Free ABE Scheme	530
4	Cryptanalysis to [36, 37]	530
5	Conclusion	531
	References	531

Implementation and Analysis of Using Autoencoder to Suppress Noise Interference on Side-Channel Attacks 535

Kuo-Yu Tsai, Chung-Wei Kuo, Wei-Ming Weng, Guan-Ling Wang, and Chun-Chang Lin

1	Introduction	536
2	Related Works	537
2.1	Advanced Encryption Standard	537
2.2	Correlation Power Analysis	538
2.3	Denoising Autoencoder	538
3	Our Proposed Method	540
3.1	Detailed Descriptions for Experiment	540
3.2	Original Signal Trace Analysis	541
3.3	Signal Trace Analysis with Noise Added	542
4	Conclusions	542
	References	543

Design and Implementation of a Secure AIoT-Based Health Promotion Prescription Systems for Community-Based Healthcare Promotion Services 545

Chien-Lung Hsu, Yu-Ting Chang, Pei-Wei Chen, Chieh-Ni Chen, and Che-Wei Su

1	Introduction	546
2	Related Works	546
2.1	Health Promotion Prescription	546
2.2	AIoT	547
2.3	Signcryption	548
2.4	Attribute-Based Signcryption	548
3	Illustrations	548
3.1	Architecture	548
3.2	System Method	550

- 4 Security Analysis 554
 - 4.1 Security Analysis Using AVISPA Tool 554
 - 4.2 Semantic Security Proof 555
- 5 Conclusion 556
- References 557
- Financial Lending Application Based on Blockchain Technology 559**
Zhe-You Liu, Ting-Yu Lin, Min-Teng Chen, and Pei-Yu Lin
 - 1 Introduction 559
 - 2 The Concept of ChainLink 561
 - 3 The Framework of the Implementation System 563
 - 3.1 The Financial Model 563
 - 3.2 Smart Contract Creating and Testing 566
 - 4 Implementation Results and Analysis 566
 - 4.1 Front-End Interface and React 566
 - 4.2 Smart Contract and Lending Development 567
 - 5 Conclusions 569
 - References 570

About the Editors

George A. Tsihrintzis is a Full Professor in the Department of Informatics in the University of Piraeus, Greece and served as its Head from September 2016 through August 2020. He received the Diploma of Electrical Engineering from the National Technical University of Athens, Greece (with honors) and the M.Sc. and Ph.D. degrees in Electrical Engineering from Northeastern University, Boston, Massachusetts, USA. His current research interests include Pattern Recognition, Machine Learning, Decision Theory, and Statistical Signal Processing and their applications in Multimedia Interactive Services, User Modeling, Knowledge-based Software Systems, Human-Computer Interaction and Information Retrieval. He has authored or co-authored over 350 research publications in these areas, which include five monographs and 26 edited volumes.

He is the co-founder and co-editor of the Springer series on Learning and Analytics in Intelligent Systems, the Editor-in-Chief of Intelligent Decision Technologies (IOS Press) and the International Journal of Computational Intelligence Studies (Inderscience) and a member of the editorial boards of 15 additional journals. He has (co-)authored six monographs and guest (co-)edited 11 special issues of international journals. He has (co-)chaired over 30 international conferences and (co-)edited 26 books.

Shiuh-Jeng WANG (<https://hera.secforensics.org/>) was a visiting scholar of Computer Science Department at Florida State University (FSU), USA in 2002 and 2004. He also was a visiting scholar of Department of Computer and Information Science and Engineering at University of Florida (UF) in 2002, 2004, 2005, 2010, and 2011, respectively. He served as the editor-in-chief of the journal of Communications of the CCISA in Taiwan from 2000 to 2006, and the directors in Chinese Cryptology and Information Security Association from 2000 to 2024. He was, then, with the Vice-President of CCISA in 2012–2015. Dr. WANG academically toured the CyLab with School of Computer Science in Carnegie Mellon University, USA, in 2007 for international project collaboration inspection.

He is also the author of 20 books (in Chinese versions): Information Security, Cryptography and Network Security, State of the Art on Internet Security and Digital

Forensics, Eyes of Privacy—Information Security and Computer Forensics, Information Multimedia Security, Computer Forensics and Digital Evidence, Computer Forensics and Security Systems, Computer and Network Security in Practice, Image Processing with Applications, Digital Forensics in Mobile Service with Evidence Investigations, and some Science Books in Fundamental of Cryptography Learning and Security and Forensic Sciences, all published in 2003, 2004, 2006, 2007, 2009, 2012, 2013, 2015, 2016, 2019, 2020, 2022, and 2023, respectively. Prof. WANG has published over 350 papers in referred Journals/Conference Proceedings/Technique Reports so far. He is a full professor and a member of the IEEE, ACM. He served a lot of academic and reputable journals in the position of guest editors, including the lead guest editor at the issue of Advances in Digital Forensics for Communications and Networking—IEEE J-SAC (IEEE Journal on Selected Areas in Communications), AT <http://www.comsoc.org/jsac>, <http://ieeexplore.ieee.org/document/5963154/> and the issue of Security and Privacy Information Technologies and Applications for Wireless Pervasive Computing Environments -INFORMATION SCIENCES AT <http://www.sciencedirect.com/science/article/pii/S0020025515005332>, <http://www.sciencedirect.com/science/journal/00200255/321>.

Prof. WANG is currently with the President of E-Security Analysis and Management Association (ESAM) AT <https://www.esam.io/> and the Director with ICCL (Information Cryptology and Construction Lab., <https://hera.secforensics.org/>) and Chairman of SecForensics (Intelligence and Security Forensics Lab., <https://www.secforensics.org/>).

Chih-Hung Wang was born in Kaohsiung Taiwan, in 1968. He received the B.S. degree in Information Science from Tunghai University and M.S. degree in Information Engineering from National Chung Cheng University, Taiwan, R.O.C., in 1991 and 1993, respectively. He received the Ph.D. degree in Information Engineering from National Cheng Kung University, Taiwan, R.O.C. in 1998. He is presently a full professor of Department of Computer Science and Information Engineering, National Chiayi University, Taiwan. His research interests include cryptography, network security and intrusion detection with deep learning. He served as the director in Chinese Cryptology and Information Security Association (CCISA) from 2009 to 2023, and the editor-in-chief of the Journal of Communications of the CCISA in Taiwan from 2009 to 2018.

Prof. Wang is currently the Executive Director of CCISA, and the Director of E-Security Analysis and Management Association (ESAM). He has served as the Program Committee Co-chair for 2019 3rd International Conference on Security with Intelligent Computing and Big-data Services (SICBS 2019) and 2021 International Conference on Security and Information Technologies with AI, Internet Computing and Big-data Applications (SITAIBA 2021).

Multimedia

Image Interpolation Utilizing Neighboring Pixels as References in Conjunction with Multilayer Perceptron Neural Networks



Tzu Chuen Lu, Thanh Nhan Vo, and Biswapati Jana

Abstract This study seeks to investigate the utilization of neural networks for the generation of high-quality image interpolation. This introduction endeavors to deliver a comprehensive exploration of the foundational principles underpinning interpolation, coupled with an investigation into the substantial potential presented by neural networks in this field. We introduce an image interpolation approach and substantiate its exceptional performance through empirical evidence. Additionally, we develop methodologies for evaluating experimental interpolation outcomes and explore potential applications in other domains.

Keywords Image interpolation · Image enhancement · Interpolation technique · Interpolation data hiding · MLP interpolation · FNN interpolation

1 Introduction

Image interpolation constitutes a critical task in digital image processing, addressing the generation of high-quality pixel information when altering image size or resolution. It finds extensive applications in digital photography, multimedia, television, film production, medical imaging, and more. Interpolation techniques enable image adaptation to diverse display devices and facilitate the processing of high-resolution images [1–7].

Traditional interpolation methods (such as nearest-neighbor and bilinear interpolation) have been employed for many years. However, with the rapid advancements in deep learning and neural networks, we now possess more potent tools to enhance

T. C. Lu (✉) · T. N. Vo

Department of Information Management, Chaoyang University of Technology, Taichung 41349, Taiwan, R.O.C.

e-mail: tclu@cyut.edu.tw

B. Jana

Department of Computer Science, Vidyasagar University, Midnapore 721102, India

e-mail: biswapati.jana@mail.vidyasagar.ac.in

image interpolation quality. Neural networks can learn feature extraction from extensive datasets and yield high-quality predictions. Hence, the application of neural networks to image interpolation has emerged as an exciting research direction.

Image data hiding is a technique employed to embed confidential information within an image while preserving its visual quality [8–13]. Interpolation techniques have been extensively utilized in image data hiding to augment hiding capacity and enhance stego-image quality.

In image interpolation techniques, unknown pixels are predicted from seed pixels before secret data are embedded into the interpolated image, thereby increasing image steganography capacity while maintaining the capability to restore the original image at the receiver's end. Through the interpolation process, this can be achieved in a reversible manner, permitting both secret data extraction and original image restoration at the receiver's end.

The motivation of this study is to harness the potential of neural networks to improve the quality and efficiency of image interpolation. Traditional methods may introduce blurriness or aliasing artifacts, particularly when dealing with intricate details and textures. Neural networks excel at capturing complex image structures [14]. Moreover, with the improvement in hardware acceleration and deep learning frameworks, implementing neural network-based interpolation methods has become more practical.

This research endeavor seeks to advance neural network-based image interpolation methods for the purpose of achieving superior image processing in terms of quality and efficiency. We will investigate various network architectures, datasets, evaluation metrics, and application scenarios to ensure the practical applicability of our approach. Through this study, we aspire to introduce further innovations to the field of image processing and multimedia applications, ultimately elevating the quality and efficiency of image processing.

The main objectives of this paper are as follows:

1. Enhance the image quality of interpolation: One of the primary objectives is to improve the quality of image interpolation by leveraging the advantages of deep learning and neural networks. We aim to develop a method capable of generating high-definition, detailed, and textured images during the scaling or resizing process.
2. Achieve higher interpolation efficiency: Another goal is to enhance the efficiency of interpolation methods, allowing them to process large images in less time. This can be achieved through optimization of network architecture, hardware acceleration, and parallel processing.
3. Universality across multiple application scenarios: Our research intends to develop a universal image interpolation method applicable to various scenarios, including digital photography, television and film production, medical imaging, and computer vision. We aim for our approach to adapt to diverse requirements in different fields.

The rest of the paper is structured as follows: In Sect. 2, the related works have been presented. Section 3 describes the proposed method. Section 4 presents the experimental results. Finally, Sect. 5 presents the discussion and the conclusion.

2 Related Works

2.1 The Enhanced Neighbor Mean Interpolation Technique (ENMI)

The interpolation technique proves to be effective in enhancing data hiding capacity and security [15]. The image interpolation process selectively modifies pixels within an image, allowing unaltered pixels to serve as reference points for image restoration. These interpolated pixels, generated based on neighboring unaltered pixels known as pivot pixels, provide a canvas for embedding secret data. To illustrate the image interpolation technique, we adopt the ENMI technique proposed by Chang et al. [16].

In the ENMI technique, the input image I with $N \times N$ size, first is scaled-down with the size $N/2 \times N/2$ to generate the original image O . Each block of input image with 2×2 size is discarded by three pixels and remains only one pixel at the top-left position. The remaining pixels are called the pivot pixels, which are used to predict the unknowing pixels. The predicted pixels are the replacements in the input image, and the pivot pixels are the unchanged pixel values of the input image. The interpolated image (P) has the same size with the input image and both are dissimilar in content at the predicted pixel positions. Eq. (1) shows the ENMI calculations.

$$\begin{aligned} P_{(0,1)} &= (P_{(0,0)} + P_{(0,2)})/2, \\ P_{(1,0)} &= (P_{(0,0)} + P_{(2,0)})/2, \\ P_{(1,1)} &= (P_{(0,0)} + P_{(0,2)} + P_{(2,0)} + P_{(2,2)})/4. \end{aligned} \quad (1)$$

An example of the ENMI method is shown in Fig. 1. The interpolated image P is generated, which carries the embedded data, and then sent to the receiver.

Where $P_{(0,0)}$, $P_{(0,2)}$, $P_{(2,0)}$, $P_{(2,2)}$ are the pivot pixels, and the remaining pixels of the 3×3 block are the interpolated pixels.

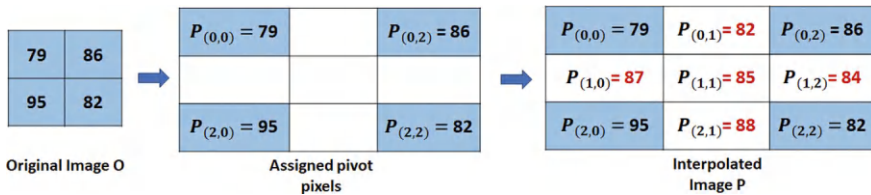


Fig. 1 An example of the ENMI technique

The calculations of the ENMI method are used to find the prediction targets within the feedforward artificial neural network (FNN) architecture. The process of artificial neural network (ANN) prediction required a much time for training in order to generate interpolated pixels. The determination of interpolated pixels is contingent on the pivot pixels. The intension is to produce the predicted image with higher visual quality. The results will provide evidence of the effectiveness of the proposed methodology.

2.2 Multilayer Neural Networks

ANNs have emerged as a widely utilized architecture in various fields. An ANN consists of a number of interconnections that connect the elements in the ANN, called neurons or nodes. Each neuron is a processing element.

One of the most widely used architectures of ANNs is the feedforward neural network (FNN), also called the multilayer perceptron (MLP). The MLP allows information to flow in one direction from input layers to the output player without any recurrent connections between neurons [17]. Each neuron in the network receives input from neurons in the previous layer and sends its output to the next layer, forming a sequential flow of information through the network.

Generally, an MLP consists of three typical classes of layers, including an input layer, one or more hidden layers, and an output layer. Before reaching the output layer, the information travels sequentially through the network starting at the input layer and passing through each hidden layer. The hidden layer consists of the computation neurons, and the output layer is composed of at least one computation neural. Each neuron acts based on the activation function, which is shown below.

$$y_i = f \left(\sum_{j=1}^n \omega_{i,j} x_j + b_i \right) \quad (2)$$

where x_j is the j th input of the i th neuron, $\omega_{i,j}$ is the weight from the j th input to the i th neuron. b_i is called the bias of the i th neuron, y_i is the output of the i th neuron, and $f(\cdot)$ is the activation function. The activation function that is nonlinear function presents the reaction of the i th neuron with the inputs $x_j(t)$, $j = 1, 2, \dots, n$. Three types of activation functions are binary step functions, linear activation functions, and non-linear activation functions.

In the proposed approach (AP), we focus on the MLP for application to predict the unknowing pixels in the image interpolation technique. The proposed technique tends to generate a high-quality interpolation image, which is used as the cover image to carry the secret data in the image steganography (Fig. 2).

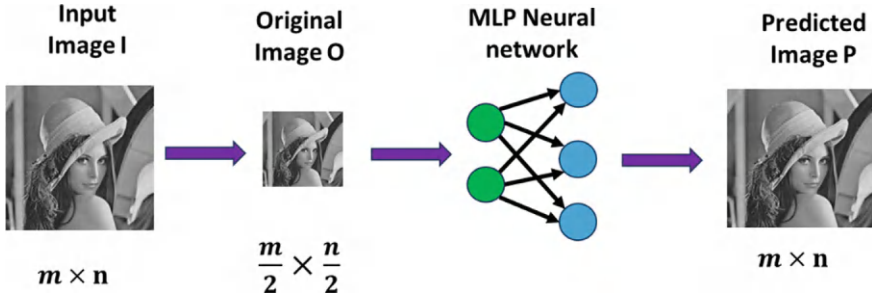


Fig. 2 The proposed approach

3 Proposed Method

The proposed approach exploits MLP-based image interpolation to predict the intermediate pixel values within a 3×3 block. Similar to the ENMI method, the input image, denoted as I , is downsampled a half and producing the original O . The resulting pixels are subsequently distributed evenly across an empty image that matches the dimensions of the input image. The pivot pixels within the assigned image are exploited for the purpose of estimating the values of the unknowing pixels. The process clears away the essentially predicted positions of a 3×3 block, as shown in Fig. 3b. Lastly, the estimating process is based on the combination of ENMI and MLP architectures in which the calculations of the ENMI are utilized for the goals for predictions, and the MLP architecture is simply designed for the interpolation, including an input layer, one hidden layer, and an output layer.

After the process identifies clearly the essentially predicted positions and remains the pivot pixels that are used as the input information of the MLP structure. This stage produces an image that consists of the pivot pixels and a number of the unknown pixels, which we call the assigned image. Then, the assigned image is divided into several overlapping blocks sized 3×3 . Each divided block includes four pivot pixels to handle the prediction of the unknown pixel values. The intermediate pixels in the block are then predicted through feedforward MLP structures with inputs selected by the values of pivot pixels to match each value of the intermediate pixels. The

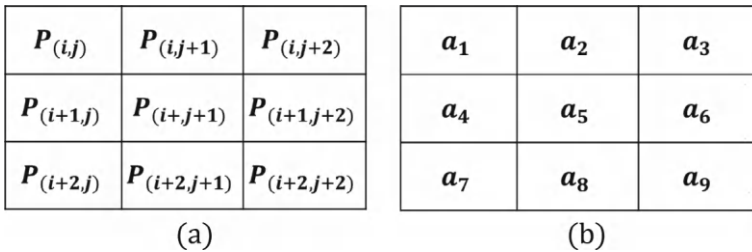


Fig. 3 A typical block sized 3×3 with the position (a) and name (b) of the pixels

targets of feedforward MLP structures are computed following the ENMI method as in Eq. (1). In a block with 3×3 size, the pixels are named and match with their respective positions as shown in Fig. 3.

Where a_1, a_3, a_7, a_9 pixels at positions $P_{(i,j)}, P_{(i,j+2)}, P_{(i+2,j)}, P_{(i+2,j+2)}$ are the pivot pixels of the cover image. These pixels are used to predict the remaining pixel values in the 3×3 block. The values in turn a_2, a_4, a_5, a_6, a_8 predicted using the MLP network structure as follows.

The proposed scheme builds three MLP structures called *Net_1*, *Net_2*, and *Net_3*. *Net_1* is used for the prediction of the center value a_5 of the 3×3 block. For the prediction, *Net_1* is formed by a MLP architecture with four inputs of a_1, a_3, a_7, a_9 and the *target_1* of *Net_1* is calculated by the average of four values a_1, a_3, a_7, a_9 , the output of *Net_1* is a_5 . The target values refer to the desired output of the network with the given inputs. During the training process of an MLP, the network learns to map input data to the corresponding target outputs. This is done by adjusting the weights and biases associated with the connections between neurons in the network. The goal of training is to minimize the error between the predicted outputs and the actual targets. This is usually done using optimization algorithms like gradient descent.

For predicting the value a_2 of the block, a MLP structure called *Net_2* is formed with inputs a_1, a_3 , and the *target_2* is the average of a_1 and a_3 . The output value of *Net_2* is assigned to a_2 . Similarly, establish a *Net_3* network with input a_1, a_7 and *target_3* is the mean of a_1 and a_7 . The output of *Net_3* is a_4 . Two positions a_6, a_8 can resemble the manner that a_2 and a_4 pixel values were predicted. The same process is carried out for the remaining blocks in the cover image. The equation is shown in Eq. (3).

$$\begin{aligned} \text{target\_1} &= (P_{(i,j)} + P_{(i,j+2)} + P_{(i+2,j)} + P_{(i+2,j+2)})/4 \\ \text{target\_2} &= (P_{(i,j)} + P_{(i,j+2)})/2, \\ \text{target\_3} &= (P_{(i,j)} + P_{(i+2,j)})/2 \end{aligned} \quad (3)$$

After interpolation, the predicted image has the acceptable resolution. The interpolated pixels are used to conceal data. The interpolation method enhances data embedding security and capacity. In the proposed method, the feedforward neural network (FNN) structures estimated intermediate pixel values, ensuring safety and producing a high-quality cover image.

The proposed method effectively exploits the features of the neighboring pixels and the MLP neural network for predicting the values of the interpolated pixels. Based on the desired goals, the MLP structures *Net_1*, *Net_2*, and *Net_3* compute moderately reasonable outputs with the given inputs, respectively.

4 Experimental Results

Six common standard images which shown in Fig. 4 with 512×512 size, Airplane, Lake, Lena, Mandrill, Pepper, and Tiffany, are used in the experiments to examine the performance of the suggested interpolation technique. The experimental results are evaluated and compared with six other methods. The image quality measurements employ the peak signal to noise ratio (PSNR) and structural index similarity (SSIM) (Fig. 4).

The measurement between the input image and the predicted image is PSNR. Equations (4) and (5) demonstrate the calculations.

$$MSE = \frac{1}{h \times w} \sum_{i=1}^h \sum_{j=1}^w (P_{i,j} - I_{i,j})^2 \quad (4)$$

$$PSNR = 10 \times \log_{10} \left(\frac{255^2}{MSE} \right) (dB) \quad (5)$$

where, MSE is the mean square error; $P_{i,j}$ is the pixel value of the predicted image, and $I_{i,j}$ is the pixel value of the input image at position (i, j) .

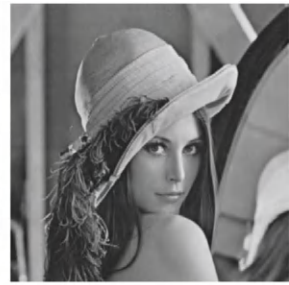
The structural similarity index measure (SSIM) predicts the quality of digital images, and media. The similarity between two images is measured using SSIM. Quality measurement metrics are calculated using brightness, contrast, and structural



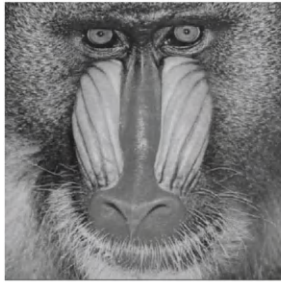
(a) Airplane



(b) Lake



(c) Lena



(d) Mandrill



(e) Pepper



(f) Tiffany

Fig. 4 The testing images

or correlation term as shown in Eq. (6).

$$\text{SSIM}(x, y) = [l(x, y)]^\alpha \times [c(x, y)]^\beta \times [s(x, y)]^\gamma \quad (6)$$

where, l is the luminance, c is the contrast and s is the structure, while α , β and γ are the positive constants [18]. Image brightness, contrast, and structure can be stated separately:

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \quad (7)$$

$$c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \quad (8)$$

$$s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \quad (9)$$

where, μ_x and μ_y are the local means, σ_x and σ_y are the standard deviations and σ_{xy} is the cross-covariance for image x and y sequentially. If $\alpha = \beta = \gamma = 1$, then the SSIM index is calculated simpler than (7)–(9) as using Eq. (10).

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_x\sigma_y + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (10)$$

In our practical experiments, the conventional computations will generate the pixel values based on a simplistic assessment. The efficacy of the offered methodologies in comparison to previous methods can be determined through the analysis of experimental findings. The efficacy of artificial neural networks in predicting outputs based on inputs and network topology enables the selection of appropriate models for application-specific reasons. In our perspective, the desired results do not necessitate precise numbers exactly. Our objective is to evaluate whether the network generates random values within the acceptable boundaries of image resolution. The results shown in Table 1 show that the MLP neural network performs well in this challenge by utilizing a single hidden layer inside the MLP design.

Table 1 illustrates that the MLP model produced the predicted pixels with better high quality in the comparisons when we ran the experiments together based on the experimental research. The compared techniques were run on one computer to ensure objective measurements. The calculations and the programming code follow the descriptions in the published techniques, and we assure that the comparisons are honest, impartial, and objective.

Table 2 shows that the PSNR and SSIM indexes of the MLP method are almost higher than the others in the comparisons with six testing images. The PSNR is the highest, which proves the proposed technique based on the computation of the MLP neural network that generates the interpolated image. The PSNR and SSIM indexes are average values that are simpler for assessing the quality of the interpolation

Table 1 The PSNR values by using different input images

Images (512 × 512)	NMI [19]	INP [20]	ENMI [16]	CRS [21]	MNMI [22]	FWS [18]	Our
Airplane	26.41	25.47	26.64	25.48	22.08	30.51	32.08
Lake	26.29	25.11	26.56	25.24	23.84	28.32	29.21
Lena	28.05	27.16	28.18	27.11	24.81	32.42	33.49
Mandrill	22.63	21.82	22.81	22.06	21.51	23.12	23.16
Pepper	27.29	26.28	27.54	26.39	24.45	29.41	31.76
Tiffany	25.63	25.23	25.70	25.29	21.43	27.15	32.85
Average	26.05	25.17	26.23	25.26	23.02	28.48	30.42

Table 2 The SSIM values by using different input images

Images (512 × 512)	NMI [19]	INP [20]	ENMI [16]	CRS [21]	MNMI [22]	FWS [18]	Our
Airplane	0.9260	0.9014	0.9314	0.8988	0.9001	0.8918	0.9431
Lake	0.8298	0.7986	0.8365	0.8070	0.8171	0.8129	0.8494
Lena	0.8945	0.8663	0.8982	0.8685	0.8782	0.8924	0.9139
Mandrill	0.7103	0.6699	0.7186	0.6730	0.6771	0.7166	0.7247
Pepper	0.8603	0.8350	0.8656	0.8420	0.8515	0.8164	0.8800
Tiffany	0.8770	0.8498	0.8807	0.8543	0.8605	0.7250	0.8942
Average	0.8497	0.8202	0.8552	0.8239	0.8307	0.8092	0.8676

techniques. In our study, to enhance the interpolation method, we need to set the output limits so that the original can be changed or we can track more features to restore the interpolation image with their features more influenced by the relationship of the original image. The feature of the original image needs to be considered in the interpolation or smooth the regions in the interpolated image with modern prediction techniques.

5 Conclusion

This paper presents an image interpolation technique primarily based on the MLP architecture. We utilize the neighboring pixel values within blocks as seeds and employ ENMI to compute potential predictions, which serve as inputs to the neural network FNN for training the predicted values of interpolated pixels. Comparative results with other methods demonstrate the effectiveness of the proposed MLP method in accurately predicting interpolated pixel values, resulting in minimal differences between the interpolated and original images. As a result, the quality of the generated images exhibits higher PSNR and SSIM values compared to other methods.

Acknowledgements This paper received partial funding assistance from the Ministry of Education's Digital Educational Material Development Program—Introduction of Neural Networks and Information Hiding Concepts into Text-Based Programming Language.

References

1. Chen YQ, Sun WJ, Li LY, Chang CC, Wang X (2020) An efficient general data hiding scheme based on image interpolation. *Inform Secur Appl* 54(102584)
2. Li X, Peng F, Yang B (2014) Improved PVO-based reversible data hiding. *Digit Signal Process* 25:255–265
3. Lu TC (2017) An interpolation-based lossless hiding scheme based on message recoding mechanism. *Optik* 130:1377–1396
4. Lu TC (2018) Interpolation-based hiding scheme using the modulus function and re-encoding strategy. *Signal Process* 142:244–259
5. Lee CF, Weng CY, Kao CY (2019) Reversible data hiding using Lagrange interpolation for prediction-error expansion embedding. *Soft Comput* 23(19):9719–9731
6. Lu TC, Huang SR, Huang SW (2020) Reversible hiding method for interpolation images featuring a multilayer center folding strategy. *Soft Comput* 25(7)
7. Mandal PC, Mukherjee I, Chatterji BN (2021) High capacity reversible and secured data hiding in images using interpolation and difference expansion technique. *Multimedia Tools Appl* 80:3623–3644
8. Alattar AM (2004) Reversible watermark using the difference expansion of a generalized integer transform. *IEEE Trans Image Process* 13(8):1147–1156
9. Cheddad A, Condell J, Curran K, McKeivitt P (2010) Digital image steganography: survey and analysis of current methods. *Signal Process* 727–752
10. Kadhim IJ, Premaratne P, Vial PJ, Halloran B (2019) Comprehensive survey of image steganography: techniques evaluations and trends in future research. *Neurocomputing* 335:299–326
11. Meikap S, Jana B (2018) Directional PVO for reversible data hiding scheme with image interpolation. *Multimedia Tools Appl* 77(23):31281–31311
12. Kumar R, Moyal V (2013) Visual image quality assessment technique using FSIM. *Int J Comput Appl Technol Res* 2:250–254. <https://doi.org/10.7753/IJCATR0203.1008>
13. Lu TC, Wu JH, Huang CC (2015) Dual-image based reversible data hiding method using center folding strategy. *Signal Process* 115:195–213
14. Russel I (2000) Hack proofing your network. Syngress
15. Hu J, Li T (2015) Reversible steganography using extended image interpolation technique. *Comput Electr Eng* 46:447–455
16. Chang YT, Huang CT, Lee CF, Wang SJ (2013) Image interpolating based data hiding in conjunction with pixel-shifting of histogram. *Supercomputing* 66:1093–1110
17. Mercép A, Mrcela L, Birov M, Kostanjčar Z (2020) Deep neural networks for behavioral credit rating. *Entropy* 23(1):27. <https://doi.org/10.3390/e23010027>
18. Jana S, Jana B, Lu TC, Vo TN (2022) Reversible data hiding scheme exploiting center folding with fuzzy weight strategy. *J Inform Secur Appl* 69:2126–2214
19. Jung KH, Yoo KY (2009) Data hiding method using image interpolation. *Comput Stand Interfaces* 31(2):465–470
20. Lee CF, Huang YL (2012) An efficient image interpolation increasing payload in reversible data hiding. *Expert Syst Appl* 39(8):6712–6719
21. Tang MW, Hu J, Song W (2014) A high-capacity image steganography using multi-layer embedding. *Optik* 125:3972–3976
22. Malik A, Sikka G, Verma HK (2020) A reversible data hiding scheme for interpolated images based on pixel intensity range. *Multimedia Tools Appl* 79:18005–18031

Reversible Data Hiding in Encrypted Image Based on Block and Huffman Encoding



Cheng-Hsing Yang, Chi-Yao Weng, Bo-Han Huang, and Wei-Qing Huang

Abstract The paper presents a reversible data hiding method with image compression consisting of Huffman coding and block encrypted. In the proposed approach, the image is used to compressed for generating the specified space for hiding the secret messages. When the secret message is extracted from the image, the location of the secret message is obtained through the image information hiding structure and Huffman coding table. After message extracting and image decryption, the image could be recovered to the original image. Experimental results show that our proposed method can achieve image recovering and have better hiding payload than that of related works.

Keywords Reversible data hiding · Image encryption · Huffman code · Data hiding

1 Introduction

With the development of information technology, it is more and more convenient for people to exchange messages. However, in the process of message transmission, there may be the problem of information interception and theft leading to the insecurity

C.-H. Yang · B.-H. Huang · W.-Q. Huang

Department of Computer Science and Artificial Intelligence, National Pingtung University, Pingtung 900, Taiwan

e-mail: chyang@mail.nptu.edu.tw

B.-H. Huang

e-mail: cbe109021@mail.nptu.edu.tw

W.-Q. Huang

e-mail: bbe111015@mail.nptu.edu.tw

C.-Y. Weng (✉)

Department of Computer Science and Information Engineering, National Chiayi University, Chiayi 600, Taiwan

e-mail: cyweng@mail.ncyu.edu.tw

of the user's privacy. In order to protect the security of transmitted messages, the Data Hiding (DH) technology is used to protect the secret messages during the transmission process.

In the modern network transmission of information, images gradually become common and important information, and they are widely utilized as the main carrier of information. Thus, the information hiding technology of images has become very important. Reversible Data Hiding (RDH) is one of the important technologies for image information hiding. RDH technology not only guarantees that the information cannot be stolen during the information transmission process but also ensures that the secret information can be extracted and the original image can be recovered completely.

The principle of reversible information hiding is to establish a secret message transmission channel between the sender and the receiver. The sender protects the security of the secret message through the reversible information hiding technology, and the receiver uses the same reversible information hiding technology to extract the secret message and restore the original image. Even if the message is intercepted during transmission, RDH is guaranteed that a third party cannot steal the secret message in the image without being sure of its information hiding technology.

Existing reversible information hiding techniques can be broadly categorized into two types: Reserved Room Before Encryption (RRBE) and Vacated Room After Encryption (VRAE). The RRBE algorithm [1] preserves the privacy space in the original image. The VRAE algorithm [2] encrypts the image and then hides the secret message. Nowadays, the proposed reversible information hiding techniques include Histogram Shifting (HS) [3, 4], Difference Expansion (DE) [5, 6], and Lossless Compression (LC) [7]. HS approach is the distribution of pixel values in the image that is counted and converted into histogram form. From this histogram form, the peaks and zeros are identified, and the pixel values between the peaks and the zeros are shifted to make the peaks available for information hiding. Difference Expansion [5, 6] used the difference between neighboring pixels by expanding the difference for hiding the secret message. Lossless Compression (LC) [7] used image compression to generate additional space for hiding the secret message.

This paper proposed a reversible data hiding scheme based on image compression, which uses Huffman coding and block cipher to compress the image and generate additional secret information hiding space. The secret message can be extracted and the original image can be recovered by the image information hiding architecture.

This paper is organized as follows: This chapter introduces our proposed method, Chap. 3 presents our experimental results and analysis, and Chap. 4 concludes the paper.

2 The Proposed Approach

2.1 Image Encryption

In our image encryption procedure, some information, such as prediction error map, Huffman code, and extra overhead, are generated. For an original image I of size $W \times H$, is first generated a prediction error map, then, use Huffman code to obtain a shorter encrypted message. Next, filter the block for block encryption. Finally, embed additional information and encrypted message into the blocks. The following steps explain our approach in detail.

2.1.1 Generating Prediction Error Map

First, the original image $I = \{p_{i,j} | 1 \leq i \leq W, 1 \leq j \leq H\}$ of size $W \times H$ is used to generate the prediction error map. The prediction error is carried out by targeting the characteristic of the image that neighboring pixels do not have a high difference in value in order to enhance the hiding capacity after encoding.

The first step uses the first pixel value of the original pixel I as the record point $p_{1,1}' = p_{1,1}$ of the prediction error map $P = \{p'_{i,j} | 1 \leq i \leq W, 1 \leq j \leq H\}$. The record point of the prediction error map can be used to restore the original image. The second step is to use the neighboring pixel value to generate the prediction error map as follows Eq. (1).

$$p'_{i,j} = \begin{cases} p_{i,j} - p_{i,j-1} & \text{if } i = 1, j > 1 \\ p_{i,j} - p_{i-1,j} & \text{if } i > 1, j = 1 \\ p_{i,j} - \bar{p}_{i,j} & \text{otherwise} \end{cases} \quad (1)$$

where $\bar{p}_{i,j}$ is predicted value using the median edge detector (MED) prediction algorithm.

$$\bar{p}_{i,j} = \begin{cases} \min(b, c) & \text{if } a \geq \max(b, c) \\ \max(b, c) & \text{if } a \leq \min(b, c) \\ b + c - a & \text{otherwise} \end{cases} \quad (2)$$

where the pixel relationship of $a = p_{i-1,j-1}$, $b = p_{i,j-1}$, $c = p_{i-1,j}$ is shown in Fig. 1.

For example, we apply an original image of size 4×4 to generate the prediction error map as shown in Fig. 2. The first pixel of the original image is used to as the record point $p'_{1,1} = p_{1,1} = 155$ of the prediction error map. To predict the pixel at the position $i = 1, j > 1$ or $i > 1, j = 1$, the left or top pixel is used for prediction, such as $p_{1,3}$ is used to predicted by $p_{1,2}$, then, the prediction error is $p'_{1,3} = p_{1,3} - p_{1,2} = 154 - 158 = -4$. In the other way, MED algorithm is

Fig. 1 The pixel relation diagram of MED algorithm:
a Position relation diagram;
b Symbol contrast diagram

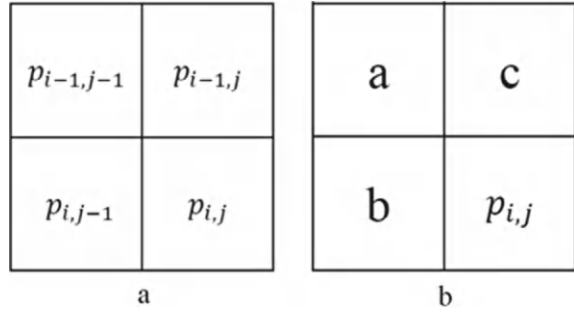


Fig. 2 Generating prediction error map:
a original image;
b Prediction error map

155	158	154	157	155	3	-4	3
153	152	155	156	-2	-4	3	-1
154	158	159	157	1	5	1	-2
155	150	154	159	1	-8	3	5

a
b

applied to predict the pixel at other locations, such as prediction error at $p_{3,4}$ is $p'_{4,3} = p_{4,3} - \bar{p}_{4,3} = 154 - (159 + 150 - 158) = 3$.

2.1.2 Huffman Coding

After generating the prediction error maps, our method uses Huffman codes to encode the prediction error maps for obtaining shorter encrypted information to increase the space of hidden message. Form a set $S = \{s_i | i = 1, 2, \dots, n\}$ of all pixel values in the prediction error map P , and generate a Hoffman coding table $T = \{t_i | i = 1, 2, \dots, n\}$ for the set S , as shown in Fig. 3. We use the prediction error map P generated in the previous subsection to generate the Huffman coding table T . We obtain the set of prediction errors $S = \{1, -1, -2, 3, -4, 5, -8, 155\}$ and the Huffman coding table $T = \{1: '111', -1: '0101', -2: '011', 3: '10', -4: '001', 5: '110', -8: '000', 155: '0100'\}$.

2.1.3 Block Encoding

There are smooth and complex blocks in the original image. In the smooth blocks, our method uses the prediction error for Huffman coding to receive shorter encrypted message. However, in the complex blocks, applying the prediction error for Huffman

Prediction error map P				Huffman Code Table T	
155	3	-4	3	S	T
-2	-4	3	-1	1	'111'
1	5	1	-2	-1	'0101'
1	-8	3	5	-2	'011'
				3	'10'
				-4	'001'
				5	'110'
				-8	'000'
				155	'0100'

Fig. 3 Huffman coding table

coding will result in lengthier code information, thus our method will use block encoding approach to filter the image blocks for obtaining shorter encrypted information instead of Huffman coding.

The original image is first divided into non-overlapping block with size of 4×4 . Apply the Huffman coding table T mentioned in previous subsection to encode and calculate the encode length L_1 . The encode length L_0 is directly using original pixel values without using Huffman coding. The lengths of L_1 and L_0 are judged as in Fig. 4. Our coding method for this block depends on which code length is shorter. Assume that $L_1 \leq L_0$, the block is encoded by Huffman coding. If $L_1 > L_0$, then, the block is encoded by original pixel values. Our method will use 1 bit to record the block whether is using Huffman coding or not as shown in the red part. In this example, the block can get shorter encrypted message by using Huffman coding because of $L_1 \leq L_0 = 47 \leq 129$.

After block encryption to determine the encoding method of each block, we encode the original image I to get the encrypted message $C = \{c_{i,j} | 1 \leq i \leq \frac{W}{4}, 1 \leq j \leq \frac{H}{4}\}$.

2.1.4 Overhead Information Processing

After generating prediction error, Huffman coding, and block encryption, we should store the Huffman coding table T for resorting the encrypted message. The encrypted message C embedded in the original image, and hiding capacity E , as shown in Fig. 5. After collecting all information, we can get the encrypted image I' .

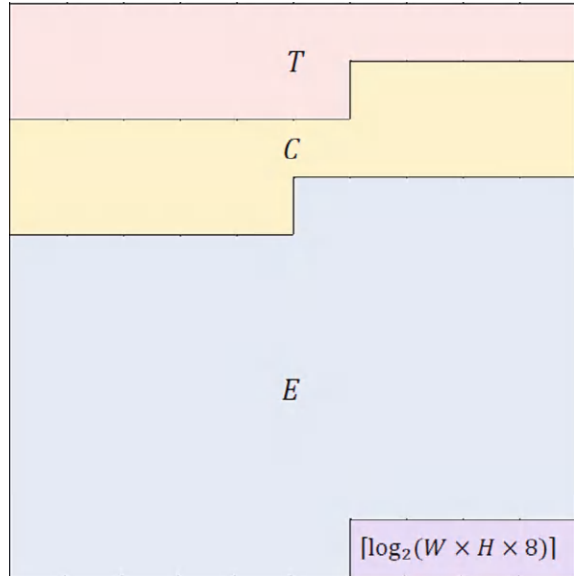
(a) Huffman Coding Table T

We use a 14-bit to present the length of the Hoffman coding table T and store them on the front of image. The length of T is $L_T = 9 + 9 \times n + \sum_{i=1}^n (8 + L_{t_i})$. We use a 9-bit representation of the number n of elements in the prediction error set S . The interval for element S is $[-255, 255]$ and reserve $9 \times n$ bits to present

Original Image I				Prediction Error Map P'			
155	158	154	157	155	3	-4	3
153	152	155	156	-2	-4	3	-1
154	158	159	157	1	5	1	-2
155	150	154	159	1	-8	3	5

Huffman Code Table T

S	T
1	'111'
-1	'0101'
-2	'011'
3	'10'
-4	'001'
5	'110'
-8	'000'
155	'0100'

 $C_1 : \mathbf{1} 0100 10 001 10 011 001 10 0101 111 110 111 011 111 000 10 110$
 $C_0 : \mathbf{0} 10011011 10011110 10011010 10011101$
 $10011001 10011000 10011011 10011100$
 $10011010 10011110 10011111 10011101$
 $10011011 10010110 10011010 10011111$
 $Length(C_1) : L_1 = 47 \text{ bits}$
 $Length(C_0) : L_0 = 129 \text{ bits}$
Fig. 4 Block encryption**Fig. 5** Schematic diagram of image information hiding structure

the values of the elements of S . As storing the Huffman coding elements of T , each element uses 8 bits to store its code length L_{t_i} message. When reading the length message t_i , we know the length of the code L_{t_i} and then read the next L_{t_i} bits to get the code L_{t_i} .

(b) **Encrypted message C of the original image**

After storing the Huffman coding table T , embed the encrypted message C in the original image using the L_C bit.

(c) **Hiding capacity E**

We use the end bit of the image $\lceil \log_2(W \times H \times 8) \rceil$ to store the length of the hiding capacity $L_E = W \times H \times 8 - (14 + L_T + L_C + \lceil \log_2(W \times H \times 8) \rceil)$. In the procedure of information hiding or message extracting, fetch the length of hiding capacity L_E to get location information used to hide the secret message.

2.2 Information Hiding

After encrypting the image, we get the encrypted image I' by reading $\lceil \log_2(W \times H \times 8) \rceil$ bits at the end of the image to obtain the spatial location where the secret message can be hidden. Then, embed the secret message E into the location, we can get the encrypted image M which hides the secret message.

2.3 Message Extracting and Image Recovery

When the receiver receives an encrypted image that hides a secret message M , we extract the secret message E by fetching the end $\lceil \log_2(W \times H \times 8) \rceil$ bits of image to obtain the length of the hidden space L_E .

To recover the original image I , we should extract the Huffman coding table T to decode the encrypted message. The decode manner is to read the front bits of the image to obtain the length L_T of the stored Huffman coding table T , and extract the L_T bits to restore the Huffman coding table T .

Finally, the encrypted message C is extracted to decode the restored prediction error map P . By reading the value of the leading bit, it is known whether the block is encoded using Huffman coding or using the original pixel value. If the pre-bit value is 1, then use Huffman coding table T to decode the block, otherwise, if the pre-bit value is 0, then directly convert to the original pixel value.

After decoding the encrypted message C , we need to restore the part of the prediction error to recover the original image $I = \{p_{i,j} | 1 \leq i \leq W, 1 \leq j \leq H\}$.

If the pre-bit is 1, then the prediction error is restored; if the pre-bit is 0, then nothing to do. The following is the prediction error recovery method.

$$p_{i,j} = \begin{cases} p_{i,j-1} + p'_{i,j} & \text{if } i = 1, j > 1 \\ p_{i-1,j} + p'_{i,j} & \text{if } i > 1, j = 1 \\ \bar{p}_{i,j} + p'_{i,j} & \text{otherwise} \end{cases} \quad (3)$$

where $\bar{p}_{i,j}$ is the prediction value using MED algorithm.

3 Experimental Results

In this section, we will demonstrate our performance and the comparison between ours and related works. Our method applied the prediction error map and Huffman coding to obtain compressed space for information hiding. Figure 6 shows the comparison between the number of prediction errors and the coding length using Lena image. This figure shows when the number of prediction error values is higher, the length of the generated Huffman code is shorter.

Seven images of size 512×512 are used as the tested image in this paper, the tested images are Lena, Airplane, Barbara, Peppers, Boat, Lak, and Baboon, as shown in Fig. 7.

The original image has smooth areas and complex area. The difference between neighboring pixels in the smooth region is small, and the difference between neighboring pixels in the complex region is large, so the encoding length of the complex block is longer. Table 1 shows that the tested image is divided into 4×4 blocks. Each block uses block encryption to select the encoding method of its blocks and number of blocks in each encoding method. From Table 1, the C_1 is greater than C_2 , indicating that our method can compress and create more space for information hiding.

In addition, we apply the ER (Embedding Rate) to measure the collection of each image in terms of bpp (bits per pixel). The collection rate ER (bpp) is calculated as follows

$$ER(bpp) = \frac{L_E}{W \times H} \quad (4)$$

where $L_E = W \times H \times 8 - (14 + L_T + L_C + \lceil \log_2(W \times H \times 8) \rceil)$, and $W \times H$ is the original image size.

Table 2 presents the bpp comparison between the proposed method and related works. From Table 2, it indicates that our method has a larger hiding capacity than related works. harborage space.

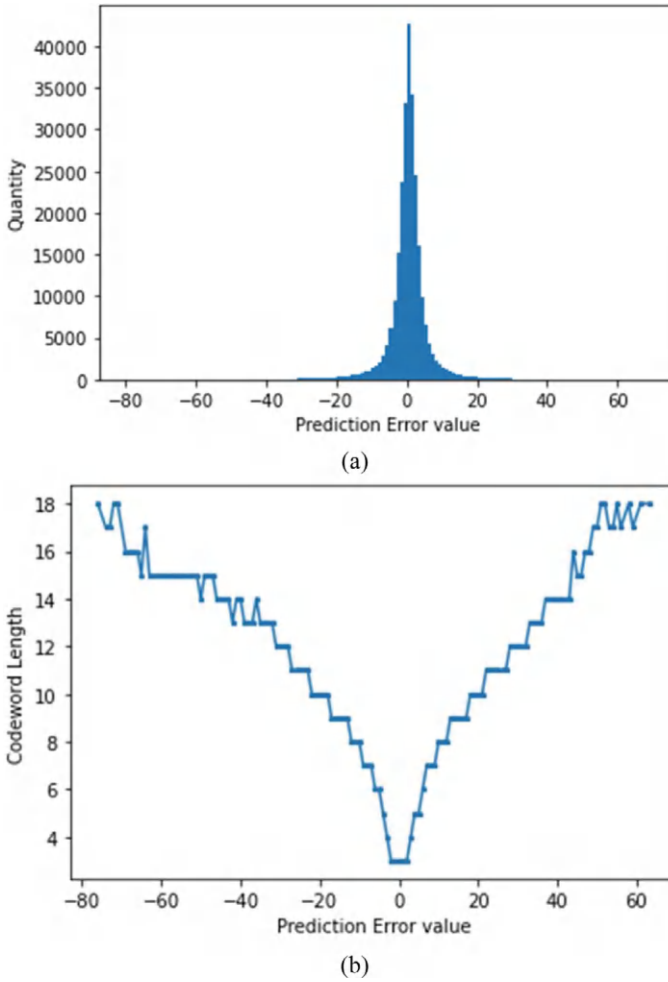


Fig. 6 Lena image comparison between the number of prediction errors and coding length: **a** The number of prediction errors; **b** The coding length of the prediction errors

4 Conclusion

In this paper, we propose a reversible data hiding method using image compression. This article uses prediction error and encryption by Huffman coding, and block encryption is selected according to the coding method of each block to generate extra space to hide the secret message. Following up the analysis of the experimental results, our proposed method applies to maximize the compression of the prediction error by Huffman coding through the feature of small difference of the neighboring pixels, which produces a larger hiding space.

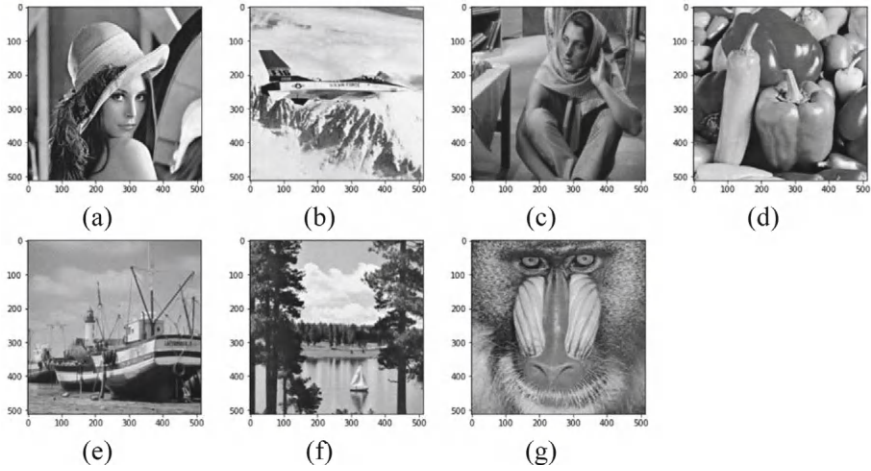


Fig. 7 Seven tested images: **a** Lena; **b** Airplane; **c** Barbara; **d** Peppers; **e** Boat; **f** Lak; **g** Baboon

Table 1 Number of blocks for each coding method

	Lena	Airplane	Barbara	Peppers	Boat	Lak	Baboon
C_1	16,178	16,167	14,830	16,215	16,132	16,627	15,159
C_0	206	217	1,554	169	252	117	465
Total	16,384	16,384	16,384	16,384	16,384	16,384	16,384

C_1 : with Huffman coding

C_0 : without Huffman coding

Table 2 ER (bpp) Comparison between the proposed method and other methods

	Wang [10]	Chen [11]	Chen [12]	Wang [8]	Gao [9]	Ours
Lena	1.89	2.58	3.18	1.65	1.71	3.33
Airplane	2.29	2.91	3.40	2.01	2.04	3.70
Barbara	1.40	2.00	2.44	1.17	1.20	2.61
Peppers	1.82	2.51	2.98	1.51	1.56	2.97
Boat	1.93	2.54	2.98	1.29	1.41	3.41
Lake	1.56	2.18	2.52	1.24	1.32	2.97
Baboon	0.66	1.21	1.53	0.52	0.81	2.04
Average	1.65	2.28	2.72	1.34	1.43	3.00

Acknowledgements This work was partially supported by the National Science and Technology Council of the Republic of China under the Grant Numbers MOST 110-2221-E-153-002-MY2, MOST 111-2221-E153-005, and NSTC 112-2221-E-153-003.

References

1. Ma K, Zhang W, Zhao X et al (2013) Reversible data hiding in encrypted images by reserving room before encryption. *IEEE Trans Inf Forensics Secur* 8(3):553–562
2. Puech W, Chaumont M, Strauss O (2008) A reversible data hiding method for encrypted images. In: *Proceedings of SPIE*, vol 6819, pp 68191E-1–68191E-9
3. Ni Z, Shi Y, Ansari N, Su W (2006) Reversible data hiding. *IEEE Trans Circ Syst Video Technol* 16(3):354–362
4. Tai WL, Yeh CM, Chang CC (2009) Reversible data hiding based on histogram modification of pixel differences. *IEEE Trans Circ Syst Video Technol* 19(6):906–910
5. Tian J (2003) Reversible data embedding using a difference expansion. *IEEE Trans Circuits Syst Video Technol* 13(8):890–896
6. Li J, Li X, Yang B (2013) Reversible data hiding scheme for color image based on prediction-error expansion and cross-channel correlation. *Signal Process* 93:2748–2758
7. Yin Z, Xiang Y, Zhang X (2020) Reversible data hiding in encrypted images based on multi-MSB prediction and Huffman coding. *IEEE Trans Multimedia* 22(4):874–884
8. Wang Y, He W (2021) High capacity reversible data hiding in encrypted image based on adaptive MSB prediction. *IEEE Trans Multimedia* 24:1288–1298
9. Gao K, Horng JH, Chang CC (2022) High-capacity reversible data hiding in encrypted images based on adaptive block encoding. *J Visual Commun Image Represent* 84
10. Wang X, Chang C, Lin C (2021) Reversible data hiding in encrypted images with block-based adaptive MSB encoding. *Inf Sci* 567:375–394
11. Chen F, Yuan Y, He H et al (2021) Multi-MSB compression based reversible data hiding scheme in encrypted images. *IEEE Trans Circ Syst Video Technol* 31(3):905–916
12. Chen F, Yang Y, He H, Yuan Y (2021) Adaptive coding and ordered index extended scrambling based RDH in encrypted images. *IEEE Trans Multimedia*. <https://doi.org/10.1109/TMM.2021.3132168>
13. Yang Y, He H, Chen F, Yuan Y, Mao N (2023) Reversible data hiding in encrypted images based on time-varying Huffman coding table. *IEEE Trans Multimedia* 1–12

Authentication of Image Fidelity in Watermarking-QR-Code Approving Copyright Systems



Hung-Jui Ko, Cheng-Ta Huang, and Shiuh-Jeng WANG

Abstract In this study, we propose a digital watermarking scheme with quick response (QR) codes for authenticating image fidelity. We hide an image hash value in a QR code as a watermark and then embed the watermark using our robust digital watermarking scheme. The proposed scheme takes advantage of QR codes for error correction. Using a QR code as a watermark can decrease the error rate after watermark extraction from an attacked image. To protect the content of images, we perform the perceptual hash algorithm for evaluation. The values of the perceptual hash algorithm can be stored in QR codes. Therefore, perceptual hash values are secured, even when the watermarked image has been damaged. Using perceptual hash values, we can determine the difference between the attacked and original images. The proposed scheme provides an effective method for protecting public and copyrighted digital images.

Keywords Digital watermarking · QR code · Image fidelity · Perceptual hash algorithm

1 Introduction

Digital images have become increasingly popular in recent years. Digital images are readily edited or altered and are distributed globally. It is becoming increasingly difficult to protect copyright and intellectual property. However, copyright protection

H.-J. Ko

Information Management Division, Changhua County Police Department, Changhua, Taiwan

C.-T. Huang

International Bachelor Program in Informatics, Yuan Ze University, Taoyuan, Taiwan

Department of Information Management, Yuan Ze University, Taoyuan, Taiwan

S.-J. WANG (✉)

Department of Information Management, Central Police University, Taoyuan, Taiwan

e-mail: dopwang@gmail.com

is not always an issue for some images, especially when such images are public and can be used for machine learning (ML). However, the fidelity of these images is a major concern. Malicious attackers may perturb an image through adversarial attacks to confuse ML models [1].

Using cryptographic hash algorithms or digital signatures to protect an image is impractical because the image could still be slightly edited or format converted. The digest of such a cryptographic hash algorithm on such an image can be changed by simply converting the image format. Therefore, perceptual hash algorithms, and digital image watermarking are better approaches for protecting copyright or content. Perceptual hash algorithms generate content-based image hashes. Unlike cryptographic hashes, perceptual hashes are robust to significant changes when an image undergoes minor modifications [2].

The use of digital watermarking schemes to protect the ownership of an image is very popular. A digital watermarking scheme should be robust to tampering and imperceptibility. Here, a logo or trademark is always used as a digital watermark; however, it is insufficient to protect the content of the target image. We need a perceptual hash to preserve the essential information of the image. Watermarked images can have identical image hash values with the original images if the digital watermarking scheme used can maintain high imperceptibility. For this reason, an increased capacity to store and retain perceptual hashes is required, particularly during image processing.

To enhance the robustness of digital watermarking methods, an error-correcting code (ECC) can be applied to these methods against attacks [3]. Lefèvre et al. proposed a different type of ECC called rank metric codes. These codes are used to improve robustness against random errors. Anand et al. applied the Hamming code to digital watermarking [4]. Based on the error correction function of the Hamming code, the noise distortion can be reduced. In 2022, Artiles et al. used the BCH code as an ECC to propose a robust image watermarking method [5]. The generated parity bits by the BCH code were embedded in the digital image. When the image was to be authenticated, the extracted parity bits could be used to fix and recover the watermark. Besides an ECC, a quick response (QR) code is also an ideal logo image as a digital watermark. A QR code is a binary image that can store text and has an error correction mechanism. Even if a part of a QR code is missing or damaged, it is still readable. Using a QR code as a watermark can help store more information about images securely.

2 Background and Related Works

2.1 QR Code

A QR code was invented by a Japanese company named Denso Wave in 1994 [6], and a standard for QR code was established by the International Organization for Standardization [7]. Since then, QR codes have seen widespread adoption worldwide due to their ease of use, high capacity, small size, damage resistance, readability in any direction, and so on. A QR code symbol is constructed as a two-dimensional (2D) array of light and dark squares, referred to as modules. The QR code error recovery capability increases at the expense of the message length. An advantage of using a QR code as a watermark is that it is easy to repair. Each module of a QR code is composed of $n \times n$ pixels. We can determine whether 0 or 1 is the majority in a module, and fix all pixels with the majority bit.

2.2 Perceptual Hash Algorithm

Image hash algorithms mainly include aHash, pHash, and dHash. The pHash algorithm is based on discrete cosine transform (DCT) [8]. The pHash algorithm is highly accurate but has an average speed compared with other image hash algorithms such as aHash and pHash [9, 10].

To determine the similarity of two images, we can compute the Hamming distance between their two pHash values. The Hamming distance between two values of equal length is the number of positions at which their corresponding bits are different.

2.3 Digital Watermarking

Digital watermarking is used to protect the ownership of digital media over many decades. Digital media includes audio, image, and video. The main idea of digital watermarking is to embed a logo or trademark in digital media. The embedded digital watermark should be robust to alteration and imperceptibility.

There are two main techniques of digital watermarking: spatial- and frequency domain. Spatial-domain techniques perform alteration on pixels [11]. Frequency-domain techniques modify coefficients after DCT [12] or discrete wavelet transform (DWT) [13].

There is a different technique of digital watermarking: zero watermarking [14]. Zero watermarking creates shares based on the content of an image without embedding any information. One of the shares is registered to a certificated authority, and the other shares can be used to verify the ownership when two shares are combined.

Zero watermarking does not alter the image, but it needs extra space for storing shares.

2.4 Related Works

QR codes have been used in various ways in the field of digital watermarking. In 2009, Feng et al. [15] studied a method of using 2D barcodes as watermarks. Their technique can resist a compression attack. To improve the imperceptibility of a watermark, Lee et al. [16] proposed a digital watermarking scheme based on just noticeable differences. The human visual system may be incapable of detecting the difference between the original images and those watermarked by Lee et al.'s scheme. In 2014, Kang et al. [17] proposed a watermarking approach based on DCT, QR codes, and chaotic theory. In their scheme, a QR code image is encrypted with a chaotic system to enhance the security of the watermark. In 2015, Thulasidharan et al. [18] proposed a scheme for detecting a malicious interference by an attacker based on a unique image registry code. The QR code is embedded in the HH component of the DWT domain of a cover image. Meng et al. [19] proposed a copyright management system based on digital watermarking and blockchain. They generate a QR code to contain image hash and copyright information as a watermark. The QR code, image, and cryptographic hash of the image are stored on a blockchain. Liu and Tang [20] proposed a QR code-watermarking approach based on image segmentation in 2020. Liu et al. divided the original image into sub-blocks and calculated the digital signature of each sub-block. The digital signature is encoded as a QR code to be embedded in the original image. This approach of Liu et al. could detect tampered image areas, but it also decreased the quality of the watermarked image.

3 Proposed Method

The proposed method is based on our previous work [12]. We adopt an inter-block DCT coefficient correlation approach and exploit QR codes as a digital watermarking scheme.

3.1 Preliminary Phase

In the preliminary phase, we calculate the pHash value of the cover image and construct a QR code to store the pHash value as a watermark. The cover image is divided into non-overlapping blocks, and DCT is performed for each block. The operation of the preliminary phase is as follows:

Input: original image I , key p

Output: non-overlapping blocks in the DCT-transformed image I_{DCT} , permuted watermark W_p

Step 1. Calculate the pHash value HV of image I

Step 2. Generate QR code to store HV as watermark W

Step 3. Permute the watermark using the pseudorandom bits named p

Step 4. Input original image I of size $W \times H$ is divided into non-overlapping $N \times N$ pixel blocks, resulting in $R \times C$ blocks

Step 5. Perform DCT for each block. The matrix of DCT blocks is $B_{P,Q}$, where $1 \leq P \leq R$ and $1 \leq Q \leq C$.

3.2 Embedding Phase

To embed the watermark bits in the DCT blocks, we calculate the difference between the DCT coefficients of the adjacent blocks. The difference is denoted as “*Diff*.” Note that there are four types of transitions, namely LR (Left-to-Right), RD (Right-to-Down), DU (Down-to-Up), and RL (Right-to-Left). The four transitions for “*Diff*” and the reference block (RB) include the following:

$$RB(x, y) = \begin{cases} B_{P,Q+1}(x, y), & \text{if transition is LR} \\ B_{P+1,Q}(x, y), & \text{if transition is UD} \\ B_{P-1,Q}(x, y), & \text{if transition is DU} \\ B_{P,Q-1}(x, y), & \text{if transition is RL} \end{cases}$$

$$Diff = B_{P,Q}(x, y) - RB(x, y). \quad (1)$$

Note that the two coefficients may differ significantly; therefore, significant modifications can change the image dramatically. Therefore, we divide the *Diff* range into six regions. The actual *Diff* level is shifted or kept in the region based on the embedding bit. Then, the difference in the coefficients around the boundary can be easily modified to be extracted as 0 rather than 1 and vice versa. We also use a predefined value K to control robustness. The K value is greater than 0 and less than $Threshold/4$. If the *Diff* level is shifted to other regions, *Diff* has to be greater than the K value from the region boundary. The K value will result in more robust watermarking with degraded image quality. Here, the number of coefficient changes depends on the characteristics of DCT blocks. The modification of the coefficient $B_{P,Q}(x, y)$ for embedding a watermark bit is displayed in Fig. 1.

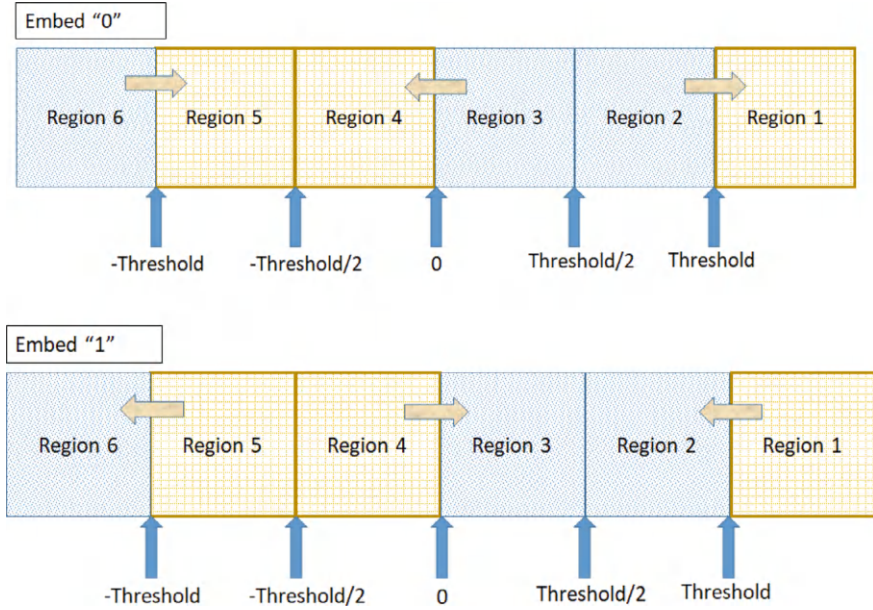


Fig. 1 Modification of coefficient $B_{PQ}(x, y)$ for embedding a watermark bit

3.3 Watermark Extraction and Verification Phase

An authorized user can verify the watermarked image by watermark extraction and by recovering the watermarked image as an original-like image after receiving the watermarked image from a sender. Watermark extraction requires a watermarked image and the initial permutation order of the binary watermark. The operation of the extraction phase is as follows:

Input: watermarked image O_E

Output: the extracted watermark WM_b , pHash value PV_E

Step 1. Divide watermarked image O_E of size $W \times H$ into non-overlapping $R \times C$ blocks

Step 2. DCT is performed for each block, and DCT blocks are denoted as B_{PQ} , where $1 \leq P \leq R$ and $1 \leq Q \leq C$

Step 3. Follow Eq. (1) to compute $Diff$

Step 4. Recover the extracted watermark using a pseudorandom key p

Step 5. Fix the extracted watermark based on the majority bits in each module, as described in Table 1.

Step 6. Read the watermark to obtain the pHash value PV_E .

By calculating the pHash value PV_W of the watermarked image, we can verify the image using the extracted watermark and pHash value PV_E . If the watermark is clear, and the Hamming distance between these two pHash values is close to zero, the watermarked image can be considered genuine.

4 Experimental Results and Discussions

All test images and watermarks are shown in Table 1. Each image size is 512×512 pixels. We take version 1 (21×21 module) and the L error correction level as our watermarks. Each module is 3×3 pixels; thus, the size of a watermark is 63×63 pixels. The capacity version 1 of the QR code with the L (low) error correction level is 25 alphanumeric, which is sufficient to contain the 16 alphanumeric in a hexadecimal system of pHash values.

We evaluate the quality of the watermarked image in terms of the peak signal-to-noise ratio (PSNR). For the extracted watermark, we use bit error rate (BER) as benchmarks. PSNR and BER are defined in [5].

Table 2 shows the watermarked image, PSNR, and pHash values. A watermarked image is imperceptibility when the PSNR is greater than 40 dB. The pHash values of the watermarked images are the same as those of the original images.

Table 1 The original images and watermark

Image 1	Image 2	Image 3	Watermark
			
Lena	pepper	airplane	

Table 2 The watermarked image

Watermarked image	Watermark	PSNR	pHash value
		41.61	99c6562d7533a296
			Original image pHash value
			99c6562d7533a296
Lena			

Table 3 Watermarks extracted after attacks and distance with original image Lenna

Attack	Proposed method		
	Extracted BER (%)	Fixed BER (%)	Hamming distance of pHash and watermarked pHash
Salt and pepper noise	17.73	0.9	0
Gaussian noise	12.37	0	0
Median Filter (3×3)	8.64	0	0
Sharpening	2.26	0	0
JPEG compression (QF = 50)	11.03	0	0
Blurring	10.25	0	0
Cropping (25%)	12.21	1.36	14

Table 4 The comparison of BER with Meng et al. [19]

Attack	Meng et al. [19] (%)	Proposed method (%)
Compression (QF = 50)	21.10	11.03
Sharpening	16.61	2.26
Blurring	20.62	10.25
Salt and pepper noise (0.01)	19.07	17.73
Cropping (25%)	17.74	12.21

The proposed method was tested by noise addition, median filtering, histogram equalization, and sharpening. These results demonstrate that the damaged watermark can be repaired using our method. Even a 12.37% damaged QR code can be repaired perfectly, confirming that a QR code enhances the robustness of digital watermarking. We list the experimental results of the first test image “Lenna” in Table 3. As shown in Table 3, the BERs of the repaired watermarks are almost 0%; furthermore, their pHash values are the same, because these image processing operations are acceptable.

In Table 4, the BER values of our proposed method are higher than those of Liu and Tang [20]. Table 5 compares our method with [5]; the fixed BERs are better than those in these previous works.

5 Conclusions

In this study, we proposed a novel method for protecting public and copyrighted images. We used the perceptual hash algorithm to generate an image hash value and store it in a QR code. Based on our previous work on digital watermarking, robustness and imperceptibility can be maintained and balanced very well. A damaged QR code

Table 5 The comparison of BER with related work

	Proposed method		[5]
Attack	Extracted BER (%)	Fixed BER (%)	BER (%)
Salt and pepper noise (0.01)	17.73	0.9	2.84
Gaussian noise (0.001)	12.37	0	2.01
Median Filter (3×3)	8.64	0	1.87
Sharpening	2.26	0	N/A
JPEG compression (QF = 50)	11.03	0	0
Cropping (25%)	12.21	1.36	0.04

can be repaired, even when the QR code looks vague. A QR code can protect the pHash value from malicious attacks.

The pHash value can also help users ascertain the difference between images. Users can set a threshold to evaluate image fidelity. Compared with the previous methods, the evaluation of image fidelity and the degree of tampering can be performed using our proposed method. Therefore, our proposed method is more effective for the public and copyright protection of digital images than existing methods.

Acknowledgements This research was partially supported by the Ministry of Science and Technology of the Republic of China under the Grants MOST 110-2221-E-015-001-MY3.

References

1. Biggio B, Nelson B, Laskov P (2012) Poisoning attacks against support vector machines. In: Proceedings of the 29th international conference on machine learning, Omnipress, Madison, WI, USA, pp 1467–1474
2. Samanta P, Jain S (2021) Analysis of perceptual hashing algorithms in image manipulation detection. *Proc Comput Sci* 185:203–212
3. Lefèvre P, Carre P, Gaborit P (2019) Application of rank metric codes in digital image watermarking. *Signal Process Image Commun* 74:119–128
4. Anand A, Singh K (2020) An improved DWT-SVD domain watermarking for medical information security. *Comput Commun* 152:72–80
5. Artiles JAP, Chaves DPB, Pimentel C (2022) Robust image watermarking algorithm using chaotic sequences. *J Inform Secur Appl* 68:103219
6. Denso Wave Incorporated, QRcode.com. <http://www.qrcode.com/en/>
7. International Organization for Standardization, ISO/IEC 18004:2015. <https://www.iso.org/standard/62021.html>
8. Ahmed N, Natarajan T, Rao K (1974) Discrete cosine transform. *IEEE Trans Comput C-* 23(1):90–93
9. Agyekum KOBO et al (2019) Digital media copyright and content protection using IPFS and blockchain. In: International conference on image graph, pp 266–277
10. Fei M, Li J, Liu H (2015) Visual tracking based on improved foreground detection and perceptual hashing. *Neurocomputing* 152:413–428

11. Wan W, Zhou K, Zhang K, Zhan Y, Li J (2020) JND-guided perceptually color image watermarking in spatial domain. *IEEE Access* 8:164504–164520
12. Ko HJ, Huang CT, Horng G, Wang SJ (2020) Robust and blind image watermarking in DCT domain using inter-block coefficient correlation. *Inform Sci* 517:128–147
13. Choudhary R, Parmar G (2016) A robust image watermarking technique using 2-level discrete wavelet transform (DWT). In: 2016 2nd international conference on communication control and intelligent systems (CCIS), Mathura, India, pp 120–124
14. Wang B, Jiawei S, Wang W, Zhao P (2022) Image copyright protection based on blockchain and zero-watermark. *IEEE Trans. Netw. Sci. Eng.* 9(4):2188–2199
15. Feng X, Ji X (2009) A blind watermarking method with strong robust based on 2D-barcode. In: 2009 international conference on information technology and computer science Kiev, Ukraine, pp 452–456
16. Lee HC, Dong CR, Lin TM (2013) Digital watermarking based on JND model and QR code features. In: Advanced intelligent systems applications, pp 141–148
17. Kang Q, Li K, Yang J (2014) A digital watermarking approach based on DCT domain combining QR code and chaotic theory. In: 2014 eleventh international conference on wireless and optical communications networks (WOCN), Vijayawada, India, pp 1–7
18. Thulasidharan PP, Nair MS (2015) QR code based blind digital image watermarking with attack detection code. *AEU—Int J Electr Commun* 69(7):1074–1084
19. Meng Z, Morizumi T, Miyata S, Kinoshita H (2018) Design scheme of copyright management system based on digital watermarking and blockchain. In: 2018 IEEE 42nd annual computer software and applications conference (COMPSAC), Tokyo, Japan, pp 359–364
20. Liu X, Tang X (2020) Image authentication using QR code watermarking approach based on image segmentation. In: 2020 IEEE 19th international conference on trust, security and privacy in computing and communications (TrustCom), Guangzhou, China, 2020, pp 1572–1577

A Zero-Watermarking Image Scheme in Normalized Cross-Correlation with Robust Copyright Protections



Hung-Jui Ko, Cheng-Ta Huang, and Shiuh-Jeng WANG

Abstract For sensitive images, e.g., military and medical images, traditional digital image watermarking methods are insufficient to realize effective protection. The original image is distorted due to traditional digital image watermarking methods embedding a watermark into it. We suggest a method for generating a zero-watermark image that is robust and uses discrete wavelet transform and singular value decomposition techniques. The Arnold transform is used in the zero-watermark generation and verification phases to enhance the security of the proposed method. The feature image of the original image is extracted after normalization and then frequency transformed. Various attacks were evaluated to determine the robustness of the proposed method, and the experimental results indicate that it exhibits good robustness against most attacks.

Keywords Digital watermarking · Zero-watermarking · Image normalization · Image protection

1 Introduction

With rapid technological developments, digital media have become increasingly popular. However, digital media, e.g., images, audio, and video, can be tampered with or distributed easily. A digital image can be altered easily and even be “face-off.” Tampering is an increasingly common challenge to copyright and intellectual

H.-J. Ko

Information Management Division, Changhua County Police Department, Changhua, Taiwan

C.-T. Huang

International Bachelor Program in Informatics, Yuan Ze University, Taoyuan, Taiwan

Department of Information Management, Yuan Ze University, Taoyuan, Taiwan

S.-J. WANG (✉)

Department of Information Management, Central Police University, Taoyuan, Taiwan

e-mail: dopwang@gmail.com

property protection. Digital image watermarking has been proposed to realize image protection. Digital watermarking schemes protect the ownership of an image by embedding a watermark. The embedded digital watermark should exhibit robustness and imperceptibility. Imperceptibility means that the original image and watermarked image cannot be distinguished by the human eye. Robustness means that the watermark must be reasonably resilient to various attacks and common signal processing operation. Note that there is always a trade-off between imperceptibility and robustness.

However, traditional digital watermarking schemes must embed the watermark into the original image, and protection schemes for military, remote sensing, or medical images are insufficient. Modification of such sensitive images could lead to incorrect judgments. Thus, zero-watermarking was proposed to address this issue. Zero-watermarking schemes do not modify the original images. Instead, they extract robust features from the original images. The extracted features are combined with a watermark to generate verification information using encryption algorithms. The trade-off between imperceptibility and robustness is resolved by this process, which provides authenticity.

2 Background and Related Work

2.1 Image Normalization

The computation of high numeric values can become increasingly complex when we put the image into transformation. To reduce computational complexity, we can normalize the values to the range 0–1. Pixel values of a grayscale image can range from 0 to 255. Since 255 is the maximum value, dividing by 255 yields a 0–1 representation. Thus, the values will be small, and the computation becomes easier and faster.

Ping Dong et al. [1] proposed an image normalization algorithm that involves translation normalization, rotation normalization, and scaling normalization. After these normalization processes, the invariance of the image is obtained. The invariance can be extracted as robust image features. In this paper, we combine these two types of image normalization processes to improve the robustness.

2.2 Discrete Wavelet Transform

The use of discrete wavelet transform (DWT) is widespread in signal processing tasks due to its frequency domain transform technique. A 2D image is decomposed into frequency channels with constant bandwidth on a logarithmic scale using DWT [2]. The image can be divided into four sub-bands that are labeled low–low (LL),

low–high (LH), high–low (HL), and high–high (HH). Decomposing these sub-bands at the next level can continue until the desired number of levels is reached. In the original image, the LL sub-band is where most of the information is contained. Due to the human visual system’s higher sensitivity to its content, the LL sub-band is considered the highest level of visual fidelity in an image.

2.3 Singular Value Decomposition

Singular value decomposition (SVD) is a technique that transforms diagonal matrices orthogonally. Here, assume that a singular value is decomposed on image A of size $m \times n$, and the SVD of image A is defined as follows:

$$A = USV^T$$

where U represents the left singular matrix, V^T represents the right singular matrix, and S represents the singular matrix value matrix of size $m \times n$. In this process, redundant information is removed after performing SVD on the image. The image’s singular value is quite stable, so it won’t significantly change even if subjected to various image attacks. The zero-watermarking algorithm’s robustness can be enhanced through the use of SVD.

2.4 Arnold Transform

There is a strong correlation between adjacent pixels in an image. Here, we consider a point (x, y) in an $N \times N$ square. When we transform the point (x, y) mapping to another point (x', y') in the square, existing a rule as shown in Eq. (1). The security of an image can be enhanced by Arnold transform by minimizing the correlation between neighboring pixels [3]. After multiple Arnold transform iterations, an original image becomes a chaotic image. After applying a certain number of transform iterations to the chaotic image, it is able to recover the original image. The Arnold transform is periodic, which implies that the original image can be reconstructed after a specified number of transformations:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \mod N. \quad (1)$$

2.5 Related Work

Transform domain and image moment methods are part of zero-watermarking algorithms. Ye proposed an algorithm for zero-watermarking that utilizes SVD and discrete cosine transform (DCT) [4], where the host image is initially divided into blocks that do not overlap. To derive the zero-watermark sequence, he compares the numerical relationship between the direct current coefficients of two adjacent blocks after performing each block using SVD and DCT. However, the attacks used in this algorithm were few. Rani and Raman [5] used DCT and SVD to extract robust features from an image to generate a zero-watermark. In this method, the host image is divided into 8×8 overlapping blocks, and each block undergoes DCT and SVD. Zero-watermarking is obtained by comparing the relationship between the two largest singular values randomly selected using a sequence of four pseudo-random numbers. Images can be protected to some extent due to the robustness and security of this algorithm. The algorithm's computational complexity increases when the host image is divided into too many overlapping blocks.

Although the abovementioned zero-watermarking algorithms can resist conventional image processing operations, they are almost vulnerable to geometrical distortions. Yang et al. [6] proposed fast generic polar complex exponential transform and quaternion generic polar complex exponential transform based on quaternion algebra. Their algorithm computes moments feature for the original image, and their experiment results demonstrated robustness against geometric attacks. Yang et al. [7] adopted normalization processing in their watermark method. Here, when the original image is normalized, the effective region is extracted as a square area in the center of the normalized image. Yang et al. select the effective region to construct zero watermark. The robustness of this method has been demonstrated to be stronger, and it is resistant to malicious image processing and geometric attacks. Wang et al. [8] proposed a resistant geometric attack zero-watermark algorithm based on nonsub-sampled pyramid decomposition (NSPD) and DCT. NSPD and DCT are used to transform the original image into coefficients. Low-frequency coefficients are employed to create a feature image, and an XOR operation is carried out on the watermark and feature images to create a zero-watermark image. They also proposed a rotation detection scheme to recover the watermark. However, their rotation detection scheme must have the original image. This scheme demonstrates good robustness against both geometric attacks and image processing, as demonstrated by their experimental results. In 2022, Wang et al. [9] proposed a DWT-SVD algorithm based on the method proposed by Yang et al. [7]. They selected an effective area from the original image and input it to DWT and SVD to extract the image features. Then, the extracted features and watermark are performed by an XOR operation to generate a zero-watermark image. Their experiment results demonstrated good robustness against various attacks.

3 Proposed Method

Two image normalization processes and the frequency domain are a foundation for the proposed method. Below are the detailed steps of the proposed method's zero-watermark generation and verification phases.

3.1 Zero-Watermark Generation Phase

In the zero-watermark generation phase, we apply image normalization to the original image, and then we perform DWT and SVD on the normalized image. Here, we extract the feature image from each matrix that is after SVD transformed. The zero-watermark image can be obtained by performing an XOR operation on the scrambled watermark and feature image. The zero-watermark generation phase is summarized as follows:

Input: Original image O , K_1 , and K_2 for Arnold transform, watermark W of size $M \times M$

Output: Zero-watermark image Z

Step 1. Input image O is normalized by [1], and then all pixels are divided by 255 as O^N .

Step 2. Perform n -level DWT on O^N , O^N is transformed into LL_n, LH_n, HL_n, HH_n .

Step 3. LL_n is extracted from O^N divided into $M \times M$ sub-blocks based on the size of watermark W .

Step 4. SVD is performed on each sub-block, and the maximum singular value of each sub-block is picked up to form matrix S .

Step 5. Feature image F is generated from the highest bit of each element of matrix S . If the highest bit is 1, the corresponding feature in the same coordinate is 1 (and vice versa).

Step 6. Feature image F is scrambled via Arnold transform with K_1 as F' .

Step 7. Logo image W is scrambled via Arnold transform with K_2 . As W' .

Step 8. An XOR operation is performed on the scrambled feature image F' and scrambled watermark image to generate the zero-watermark image Z .

Figure 1 shows the flow of the zero-watermark generation process.

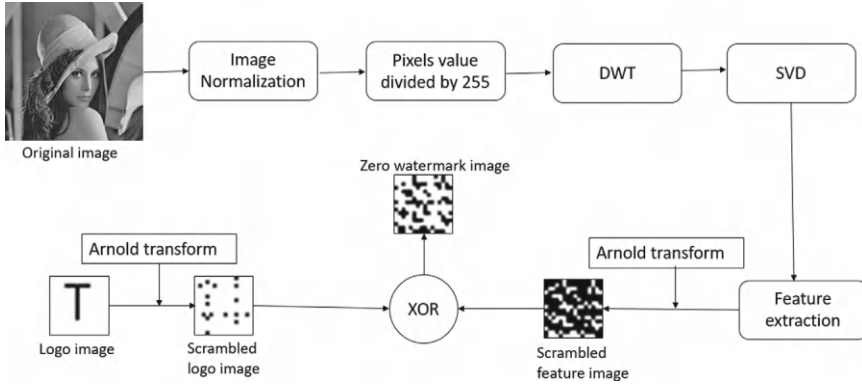


Fig. 1 Zero-watermark generation process

3.2 Zero-Watermark Verification Phase

Here, assume that OT is the target image to be verified. As follows is how the zero-watermark verification phase progresses.

Input: Image OT, K_1 , and K_2 for Arnold transform, the size of watermark $M \times M$, zero-watermark image Z

Output: Recovered watermark WR of size $M \times M$

Step 1. Input image OT is normalized by [1], and then all pixels are divided by 255 as OT^N .

Step 2. Perform n -level DWT on OT^N , and OT^N is transformed into LL_n , LH_n , HL_n , HH_n .

Step 3. LL_n is extracted from OT^N and divided into $M \times M$ sub-blocks based on the size of the watermark.

Step 4. SVD is performed on each sub-block, then we pick the maximum singular value of each sub-block to form matrix S' .

Step 5. Feature image FT is generated from the highest bit of each element of matrix S' . If the highest bit is 1, the corresponding feature in the same coordinate is 1 (and vice versa).

Step 6. Feature image FT is scrambled via Arnold transform with K_1 as FT' .

Step 7. An XOR operation is performed on zero-watermark image Z and FT' to generate the scrambled watermark W^S .

Step 8. The watermark image WR is recovered by performing Arnold transform on W^S with K_2 .

The zero-watermark verification process is illustrated in Fig. 2.

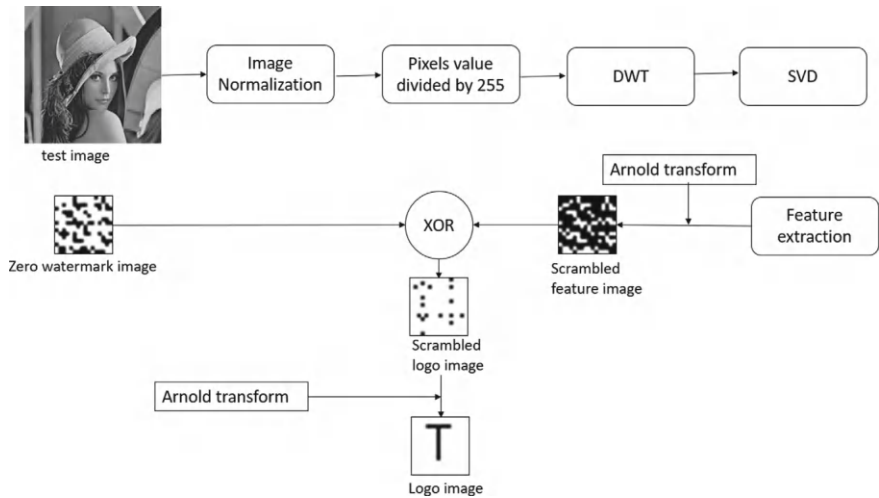


Fig. 2 Zero-watermark verification process

4 Experimental Results and Discussions

To ensure that the proposed method is robust. The test images and watermark are shown in Fig. 3. Here, each image is 512×512 pixels, and the size of watermark is 16×16 pixels.

Note that the proposed method does not modify any of the content in the original image. Thus, the quality of the original image is not reduced by the proposed method.

Man	Lenna	Goldhill	Baboon	water-mark
Barbara	Boat	Bridge	Pepper	

Fig. 3 Experimental test images and watermark

The extracted watermark is evaluated using normalized cross-correlation (NC) and bit error rate (BER) as benchmarks. NC and BER are detailed as follows.

Normalized cross-correlation: NC is used to evaluate the quality of watermark. NC is defined as follows:

$$NC = \frac{\sum_i \sum_j W(i, j) \times W_x(i, j)}{\sum_i \sum_j [W(i, j)]^2}, \quad (2)$$

where W is the original watermark bit, W_x is the extracted watermark bit, and (i, j) is the position of the watermark. Greater NC values indicate good quality for an extracted watermark.

Bit error rate: BER is used to calculate the number of error bits between a watermark and the extracted watermark. The method is robust against attacks if the BER value is 0. A lower BER value is better in comparison to different methods. BER is defined as follows:

$$BER(\%) = \frac{1}{n} \left[\sum_{j=1}^n B(j) \oplus B_x(j) \right] \times 100, \quad (3)$$

where n is the total number of embedded watermark bits, $B(j)$ is the original watermark bit, and $B_x(j)$ is the bit of the extracted watermark.

The proposed method was tested under various image processing conditions, i.e., noise addition, Wiener filtering, median filtering, JPEG compression, and scaling. The experimental results obtained with the proposed method and compared to previous methods [8–10] are shown in Tables 1 and 2. It can be observed that the proposed method outperformed the compared methods. Note that [8] requires the original image to rotate back; however, the proposed method does not require the original image.

Table 1 Comparison in terms of average NC with previous works

Attack	[9]	[8]	Proposed method
Gaussian noise (0.02)	0.9297	0.9793	0.9956
Salt and pepper noise (0.02)	0.9807	0.9895	0.9911
Wiener filtering (4×4)	0.9823	0.9939	0.9965
Median filter (4×4)	0.9684	0.9835	0.9914
JPEG compress ($Q = 10$)	0.9750	0.9863	0.9922
Rotation 30°	0.9844	0.9204	0.9961
Scale 0.9	0.9953	N/A	0.9964
Vertical mirror	0.9974	N/A	1
Rotate 30° and scale 0.9	0.9759	N/A	0.9833

Table 2 Comparison in terms of average BER (%) with previous works

Attack	[10]	Proposed method
Gaussian noise (0.003)	3.05	0.62
Salt and pepper noise (0.03)	1.05	0.58
Median filter (3×3)	0.72	0.48
JPEG compress ($Q = 40$)	0.87	0.15

5 Conclusion

Our paper proposes a reliable zero-watermarking technique to safeguard sensitive images. The proposed method utilizes two image normalizations techniques, i.e., DWT and SVD, to enhance robustness. In addition, Arnold transform is introduced in the zero-watermark generation and verification phases to improve security. As a result, the proposed method can protect the entire image against various attacks. Our experimental results demonstrate that the proposed method has good robustness against noise addition, Wiener filtering, median filtering, scaling, and JPEG compression. Comparing the proposed method to existing methods, we concluded that it is more effective in protecting digital images for public and copyright purposes.

Acknowledgements This research was partially supported by the Ministry of Science and Technology of the Republic of China under the Grants MOST 110-2221-E-015-001-MY3.

References

1. Ping Dong JGB, Galatsanos NP, Yang Y, Davoine F (2005) Digital watermarking robust to geometric distortions. In: IEEE transactions on image processing, vol 14, no 12, pp 2140–2150, December 2005. <https://doi.org/10.1109/TIP.2005.857263>
2. Mallat SG (1989) A theory for multiresolution signal decomposition: the wavelet representation. IEEE Trans Pattern Anal Mach Intell 11(7):674–693. <https://doi.org/10.1109/34.192463>
3. Guan Z-H, Huang F, Guan W (2005) Chaos-based image encryption algorithm. Phys Lett A 346(1–3):153–157. ISSN 0375–9601, <https://doi.org/10.1016/j.physleta.2005.08.006>
4. Ye T (2011) A robust zero-watermark algorithm based on singular value decomposition and discrete cosine transform. Parallel Distrib Comput Netw 137:1–8
5. Rani A, Raman B (2016) An image copyright protection scheme by encrypting secret data with the host image. Multimedia Tools Appl. 75(2):1027–1042
6. Yang H-Y, Qi S-R, Niu P-P, Wang X-Y (2020) Color image zero-watermarking based on fast quaternion generic polar complex exponential transform. Signal Process: Image Commun 82:115747. ISSN 0923-5965, <https://doi.org/10.1016/j.image.2019.115747>
7. Yang K, Wang W, Yuan Z, Zhao W (2018) Strong robust zero watermarking algorithm based on NSCT transform and image normalization. In: Proceedings of IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), pp 236–240
8. Wang R, Shaocheng H, Zhang P, Yue M, Cheng Z, Zhang Y (2020) A novel zero-watermarking scheme based on variable parameter chaotic mapping in NSPD-DCT domain. IEEE Access 8:182391–182411

9. Wang B, Jiawei S, Wang W, Zhao P (2022) Image copyright protection based on blockchain and zero-watermark. In: IEEE transactions on network science and engineering, vol 9, no 4, pp 2188–2199, 1 July–August 2022. <https://doi.org/10.1109/TNSE.2022.3157867>
10. Daoui A, Karmouni H, Sayyouri M, Qjidaa H, Maaroufi M, Alami B (2021) New robust method for image copyright protection using histogram features and Sine Cosine Algorithm. Expert Syst Appl 177

Network Security (I)

A Novel Approach for Detecting and Analyzing Cyber-Attacks in Cyber-Physical Systems



Zhi-Ren Zheng and Bo-Chao Cheng

Abstract With the rapid development of industry, we are currently experiencing the fourth industrial revolution. Many products, equipment, and sensors are connected to the internet. Although this gives us greater access to data, it opens up opportunities for hackers to access the system, and the cause for system anomaly becomes extremely complicated. However, computer systems have grown in complexity to the point where manual detection of anomalies in the system has become impracticable, and machine-led analysis must be relied upon. To more accurately discriminate the cause of the abnormality, we propose a new method based on detecting log anomalies, which is to judge whether the system has been attacked by cyber-attacks through the sequence of occurrences of cyber data and physical phenomena. In this paper, we use Autoencoder (AE) and Bidirectional Long Short Term Memory (Bi-LSTM) to discriminate sources of disturbance. Our proposed solution, “Result explainable Neural Network” (RxNN), outputs an anomaly score for each log event. This score expresses the contextual relevance of cyber data and physical phenomena. The experimental results show that the proposed solution is more accurate and effective in anomalies in the log than other neural network models.

Keywords Recurrent Neural Network (RNN) · Cyber-Physical system · Cyber-attack detection · Log analysis

Z.-R. Zheng (✉) · B.-C. Cheng
Department of Communications Engineering, National Chung Cheng University, Chiayi, Taiwan
e-mail: g10430075@ccu.edu.tw

B.-C. Cheng
e-mail: bcheng@ccu.edu.tw

1 Introduction

Computer systems are gradually integrated into modern industries and applications to help people obtain relevant data more conveniently for system analysis [1]. As these systems become more and more complex, the amount of data and computational complexity are also increasing, which makes these systems more vulnerable to attacks and damage, and the reasons for abnormalities become relatively complicated. Monitoring systems will become mission-critical today when operational technology (OT) and information technology (IT) are merged, and help to build systems that people can rely on by being able to accurately and rapidly identify anomalies and determine the source of interference [2].

Cyber-Physical System (CPS) is a control system that integrates Information Technology (IT) and Operational Technology (OT). When IT and OT converge (for example in industrial applications such as manufacturing, transportation, and health-care), the challenges posed by CPS become more complex due to the diversity of these technologies. To begin with, distinct technology stacks must solve compatibility issues and provide unobstructed communication while ensuring system security in the complex technological environment created by the merging of two different technologies. Next, it is critical for a CPS to effectively monitor, detect, and respond to threats, but monitoring OT networks without impacting their performance can be tricky. Finally, different environments face different risks, and IT and OT systems often have different priorities and risk tolerances. IT environments are more accustomed to dealing with cyber threats, while OT is focused on operational security. This gap to create a common understanding of system security goals can be challenging. Addressing these challenges requires a holistic, correlated approach involving both IT and OT. Clear security policies, careful risk assessments, and prioritization of security measures must be developed to help build a sound CPS environment.

The integration of IT and OT creates a more convenient system environment, but it also introduces a slew of important security threats that must be addressed, such as network attacks that disrupt physical end-user operations, system internal threats, and cross-domain attacks. How to perform real-time system security monitoring and detect network attacks has garnered the most attention. When the CPS environment suffers from cyber-attacks, it often affects the operation and continuity of critical infrastructure. Industrial equipment and components can be physically harmed by cyber-attacks that manipulate systems and sensors, which can lead to costly repairs and part replacements, lower operational efficiency, and downtime. If key infrastructure such as power plants, hydropower, and natural gas lose basic services due to cyber-attacks, it will cause a significant damage and even endanger national operations.

In order to effectively detect network malicious activities in the CPS system, an intrusion detection system (IDS) plays a key role in monitoring network traffic and identifying potential abnormal activities. It can provide real-time monitoring of threats to system security and perform anomaly detection to help the system identify unknown threats or abnormal activities. When an IDS identifies a potential threat or

anomaly, it would immediately issue an alert to notify the operator on the nature and severity of the event detected by the system. The alert triggered by the IDS in the CPS environment gives full awareness to the network and physical security threats that may affect the system's objective insights and reminders. Events are recorded by the detection system in the system logs, which include valuable data for post-analysis and inquiry. However, the integration of IT and OT has made the content of these logs extremely complex, encompassing data from two distinct environments, which greatly increases the difficulty of analyzing the source of anomalies. To this end, we provide a fast and accurate analysis system, which we call "Result explainable Neural Network" (RxNN), to help operators quickly analyze the source of interference and give an appropriate response.

The rest of the paper is organized as follows: Sect. 2 discusses the existing log-based anomaly detection methods and challenges. Section 3 describes the system architecture proposed in this research. Section 4 explores the experimental method and analyzes the evaluation results. Section 5 summarizes and discusses the future development and application of this paper.

2 Related Work

Recently, some deep learning detection methods have also been applied to address log analysis. Despite significant advances in log-based anomaly detection, it is still challenging to create a real-time and accurate anomaly detection system on the device, which requires high computational power to increase anomaly detection accuracy. Therefore, Wang et al. [3] proposed a novel lightweight log anomaly detection method to solve the dilemma of limited computing resources. They use Temporal Convolutional Networks (TCNs) for the supervised classification of temporal data and use multi-kernel point-wise convolution to improve residual blocks to achieve a lightweight model, which proves the feasibility of deep learning in this field. Although TCNs is a neural network specially developed for analyzing temporal data, the model has difficulty computing data features from different domains and is not ideal for handling long log sequences, so it would be difficult to operate effectively in CPS systems.

Generative Adversarial Networks (GANs) learn features through adversarial training, which is very suitable for anomaly detection. Although GANs is a fairly mature method in the field of anomaly detection, the biggest disadvantage of this method is that it takes too much time to find suitable features, and the existing methods need to set hyperparameters in the detection stage, which can negatively affect the accuracy of the detection. In order to improve the speed of GANs to judge log abnormal events and reduce the number of manual parameter adjustments, Qi et al. [4] proposed an adaptive log anomaly detection method, using Bidirectional Generative Adversarial Network (BiGAN) to calculate reconstruction loss function and discriminative loss function improve detection speed and accuracy, use ensemble learning to reduce manual adjustment of parameter settings and achieve an adaptive

learning detection system. GANs have been used to detect log anomalies, however they are still unable to handle the features of time-related data.

When the scale and complexity of the system continue to increase, it is out of reach to manually check the abnormalities in the logs. In order to cope with the increasing data volume and computational complexity, Qian et al. [5] proposed an automatic anomaly detection method based on the Variational Autoencoder (VAE). They use two distinct feature extraction techniques, which take into account the amount of log executions in addition to the sequence in which logs are executed, to determine if the system's data is abnormal. Although VAE can automatically extract log characteristics and assess their abnormality, it cannot process time-related data features as well as GANs, and it is considerably more challenging for VAE to acquire data features in many application domains.

In order to enable the system to extract time-related data features in the log, Bursic et al. [6] proposed an anomaly detection method with two autoencoders. The first AE is used to train the fixed dimension of the log, followed by the undecoded data and the appropriate attributes of the time stamp as input to the second AE. The second AE employs a Long Short Term Memory (LSTM) network, which is intended to evaluate time-related data and detect anomalies. The detection system can generate an abnormality score, which can be used to assess the system's anomaly. LSTM networks overcome the difficulty of extracting time-correlated data characteristics, but they have several drawbacks that must be addressed. It has been demonstrated that LSTM is inefficient when dealing with extended long sequences. Further, assessing log data in a CPS is problematic due to a unidirectional neural network's incapacity to derive data characteristics from multiple domains.

Table 1 compares the relevant literature with the characteristics of this study. The method proposed in this paper (RxNN) is based on log anomaly detection. Log anomaly detection is a network security technology that is usually used to identify abnormal or potentially malicious activities in log files generated by the system. Logs are a key component of CPS, and the ability to detect anomalies in logs has become a major indicator of measurement methods. All deep learning methods in the relevant literature can detect anomalies in logs. It is important to pay attention to the performance metric of log sequence length in CPS. When the model does not have enough ability to handle long sequences, it leads to performance degradation. Data from two distinct domains in the IT and OT environments are typically included in the logs produced by the CPS system, and whether this data can be handled is the major focus of observation of these approaches. It is quite difficult for deep learning methods to explain classification results. In contrast to techniques used in related research, we employ neural networks that can capture contextual cues to produce classification results that can be explained. As shown in the table, the outstanding feature of RxNN compared with other deep learning methods is that it can perfectly adapt to the CPS environment.

Table 1 Comparison of RxNN and related literature

	Log anomaly detection	Long sequence efficiency	Different domains data	Result explainability
Wang et al. [3]	✓			
Qi et al. [4]	✓	✓	✓	
Qian et al. [5]	✓	✓		
Bursic et al. [6]	✓			
RxNN	✓	✓	✓	✓

3 Proposed Method

RxNN uses a neural network technology to handle log anomaly detection and analysis. The system architecture is divided into two layers (as shown in Fig. 1). As a result, RxNN can determine if an abnormal event is triggered by a network attack and assist the operator with choosing the appropriate response and follow-up security incident handling steps.

The first layer serves as the main discriminator for identifying anomalous occurrences and events. The Autoencoder approach is anticipated to be able to discern regular occurrences from aberrant ones. In addition to accomplishing anomaly detection, autoecncoder may also lower the amount of data to ease the computational load on the second-layer resolver. The second layer, employing Bidirectional Long Short Term Memory (Bi-LSTM), trains physical phenomena and network-related characteristics from the same event but in distinct domains separately after receiving

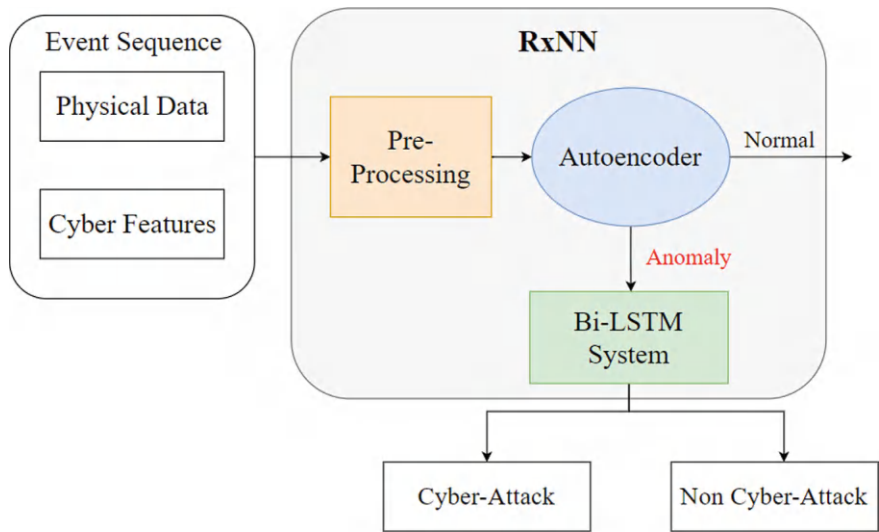


Fig. 1 RxNN system architecture

occurrences deemed as anomalous from the Autoencoder. As a result, RxNN can determine if an abnormal event is triggered by a network attack and assist the operator with choosing the appropriate response and follow-up security incident handling steps. The second layer of discriminators is also responsible for how well RxNN can explain the classification outcomes.

3.1 Pre-processing

To clean up missing or null values in the data set, we first use the normalization approach to scale every physical phenomenon and network characteristic data in proportion and convert the data to a value between zero and one during the pre-processing step. Next, we intend to separate the two different characteristics of each event at this point so that Bi-LSTM may accept them separately because each event in the CPS dataset (for example, the smart grid dataset) has both physical phenomena and network characteristics. To identify the origin of the anomaly, features from two separate domains are acquired and then its contextual properties are applied. Finally, we utilize label encoding technology to transfer the labels supplied by the dataset to the associated values, and one-hot encoding technology to transform the encoded data into a two-bit vector representation.

3.2 Autoencoder

The Autoencoder serves as the foundation for the discriminator in the first stage, whose primary function is to classify incoming events into either the normal or abnormal category. Essential features are extracted and learned by AE from the input data, and when new data is received, it is compared to the model's learned features. If the new data deviates too much from the taught properties, it is considered abnormal. In order to determine if an outlier is abnormal, AE often employs the loss function to measure its distance. Here we use the mean absolute error (MAE) to calculate, the calculation method is finding the average absolute difference between predicted values ($\hat{y}_i$) and actual values (y_i), thus evaluating the extent of data variation, as shown in Eq. (1). In order for the Autoencoder to achieve the effect of anomaly detection, we use the normal events provided by the data set to train the AE model and use the value calculated by MAE to compare with the threshold value. The model event would classify it as abnormal if it were greater than the threshold; otherwise, it would be considered a normal occurrence.

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{y}_i - y_i| \quad (1)$$

3.3 Bi-LSTM System

Bi-LSTM is a recurrent neural network that is primarily used for time-dependent data processing, natural language processing (NLP), and other related applications. The ability of Bi-LSTM to extract context-specific characteristics and allow the system to determine the most likely outcome after learning is its most potent feature. We discovered, however, that if such a potent feature of Bi-LSTM can be used, not only can it keep the properties of general recurrent neural networks that can process time-related data, but it can also have the capacity to process past and future data properties.

Because the events in the log are kept in chronological order, the properties of processing time data may be utilized for precise analysis and categorization. It may be used to collect context characteristics to deal with physical phenomena and network characteristics in the CPS dataset, as well as to determine which field of data is the source of the irregularity. The system proposed in this paper is Bi-LSTM based on the attention mechanism, which is a many-to-one architecture, that is divided into several components, as shown in Fig. 2. The Bi-LSTM layer is specifically used to handle event time and context data, which can assist RxNN in determining the cause of abnormalities. The attention mechanism can automatically seek and pick the most relevant characteristics, which can address the weaknesses of general LSTM, which cannot handle lengthy sequences well. The final estimate is calculated using the categorical cross-entropy (CE) loss function. As shown in Eq. (2), CE is a method to judge the quality of model prediction results.

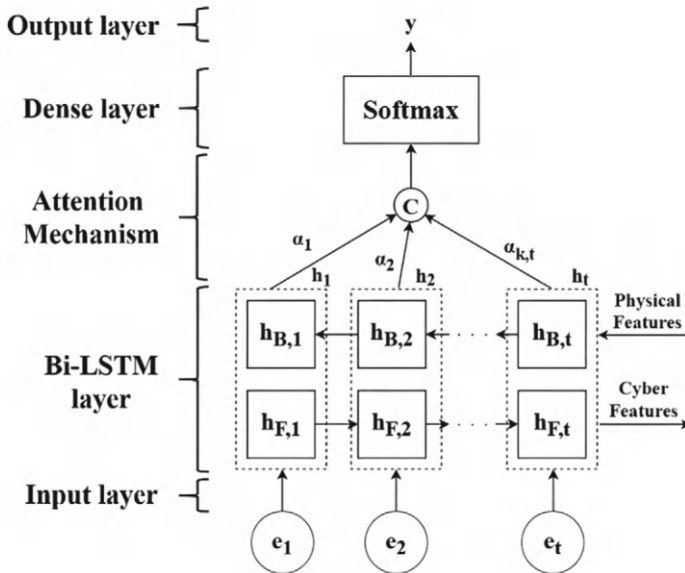


Fig. 2 Bi-LSTM system architecture

$$CE = - \sum_{i=1}^n y_i \cdot \log(p(y_i)) \quad (2)$$

A. Bi-LSTM Layer

Combining the forward and reverse directions of LSTM, this Bi-LSTM layer is used to investigate the sources of anomalies induced by abnormal occurrences as determined by the first layer of anomaly resolvers. Although unidirectional LSTM has a faster computation speed than Bi-LSTM, its fundamental disadvantage is that it does not account for the order of data before and after, leading to a loss in accuracy. The two-way LSTM improves the limitations of one-way, so it can greatly increase the accuracy of classification. In this layer, we solely utilize the Autoencoder's anomalous events to train the Bi-LSTM model, and we employ its two hidden layers in opposite directions to analyze distinct sorts of data characteristics to discover the source of the anomaly. As shown in Fig. 2, we train the forward and backward hidden layers using the split physical phenomena and network characteristics. Because the data in the data set are all time-dependent, the system decides that the event is an anomaly induced by the network field when the physical phenomena and network characteristics coincide. If just the physical feature goes through the system, it is considered a natural failure of the system. Our approach achieves the effect of identifying network threats and explaining the model's classification findings.

B. Attention Mechanism

Bahdanau et al. [7] introduced the attention mechanism, which is primarily employed for feature selection, increasing the most significant elements of the input data while lowering less important aspects. Although LSTM may retain the features of the input data, if the data set is too large, LSTM will forget the prior data after a while. It may have significant elements that influence the outcome these data may contain key aspects that influence the outcomes, which is one of the LSTM network's major flaws. The attention mechanism significantly overcomes this problem and guarantees that critical network features are not lost due to time constraints, resulting in more accurate findings. An unnormalized score is first calculated using the weights assigned to each hidden layer of the Bi-LSTM, and this score is then normalized using the normalized exponential function (Softmax) to obtain the probability value ($\alpha_{k,t}$). This probability value is then calculated using the weight (h_t) assigned to the Bi-LSTM when it was first calculated, as shown in Eq. (3). Finally, The Rectified Linear Unit (ReLU) produces a score (c), which utilizes the Softmax activation function to compute the likelihood of each category and takes the most comparable number to provide the final result.

$$c = \sum_{t=1}^t \alpha_{k,t} h_t \quad (3)$$

4 Evaluation

In this section, we first introduce the content of the data sources used in this experiment and the experimental environment. Next, in order to evaluate the effectiveness and pros and cons of the algorithm proposed in this article, we adjust some experimental variables to observe changes in accuracy use multiple metrics to evaluate the quality of the model, and compare RxNN with other methods.

4.1 Dataset: Smart Grids System

We use database information acquired by the smart grid system platform [8] to assess our method. The Smart Grid Cyber-attack dataset was created in 2014 in collaboration with Mississippi State University and Oak Ridge National Laboratory to boost cybersecurity research. The dataset utilized is a scaled-down power plant and fully operating power generating system. The dataset had 78,377 occurrences in total. These occurrences are divided into 15 data sets that comprise measurements of normal events, physical interference, and cyber-attack behaviors. The ratio of these events in the dataset is 1:3:1:2:9:2, and the detailed description of event types is shown in Table 2.

Sensors in the power system are used to collect physical attributes, and data is captured at a rate of 120 occurrences per second. The system’s voltage, current, phase, amplitude, and other pertinent statistics are among the types of data presented. Snort is used to gather network characteristics. A test bed for smart grid systems has Snort deployed as an intrusion detection system. Snort rules must be used to compare all data, and syslog must be used to log the results. This data was initially collected in the form of logs, and the unstructured data was subsequently converted into structured data, which is also the source of data used in this work.

Table 2 Dataset event scenario description [8]

Event type	Description	Ratio
Normal operation	Normal	1
Short-circuit fault	A short in a power line	3
Line maintenance	One or more breakers are opened via the remote relay trip command	1
Remote tripping command injection	This is an attack that sends a command to a relay	2
Relay setting change	The attacker changes the setting to disable the relay function	9
Data Injection	Blind operators and cause a blackout by changing values of parameters such as current, voltage, sequence, etc	2

4.2 Performance Metrics

In this section, we use the sequence length to observe changes in the accuracy of the experimental model, and apply several performance metrics to examine the performance of the experimental model to verify the effectiveness and feasibility of the proposed method. Common evaluation metrics are accuracy, precision, recall, and f1-score to evaluate the machine learning model (as shown in Eqs. (4–7)). As mentioned earlier, our proposed approach is explainable since it reveals if there is a cyber-attack alert producing system device abnormality, indicating that the OT anomaly is caused by a cyber-attack. On the other side, there is no cyber-attack notice in a specific time frame, and a device anomaly happens, indicating a natural system failure.

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \quad (4)$$

$$Precision = TP/(TP + FP) \quad (5)$$

$$Recall = TP/(TP + FN) \quad (6)$$

$$F1 - Score = (2 * Precision * Recall)/(Precision + Recall) \quad (7)$$

The evaluation findings for each model are shown in Table 3. In contrast to the generic LSTM, which performs poorly in the smart grid system without the benefit of processing contextual cues and assistance from the attention mechanism, our suggested technique, RxNN, achieves very accurate results. Bi-LSTM considerably boosts the one-way network model's performance by overcoming its one-direction restriction and processing contextual data, although it is still somewhat insufficient for handling lengthy sequences. In addition to offering very precise analysis findings, RxNN can also explain the reasons for the classification outcomes to support operators in identifying the source of interference and putting in place the necessary countermeasures to speed up system repair and improvement.

Table 3 The results of each model evaluation

	Accuracy	Precision	Recall	F1-Score
LSTM	0.76	0.77	0.76	0.76
Bi-LSTM	0.85	0.85	0.84	0.84
RxNN	0.96	0.95	0.96	0.96

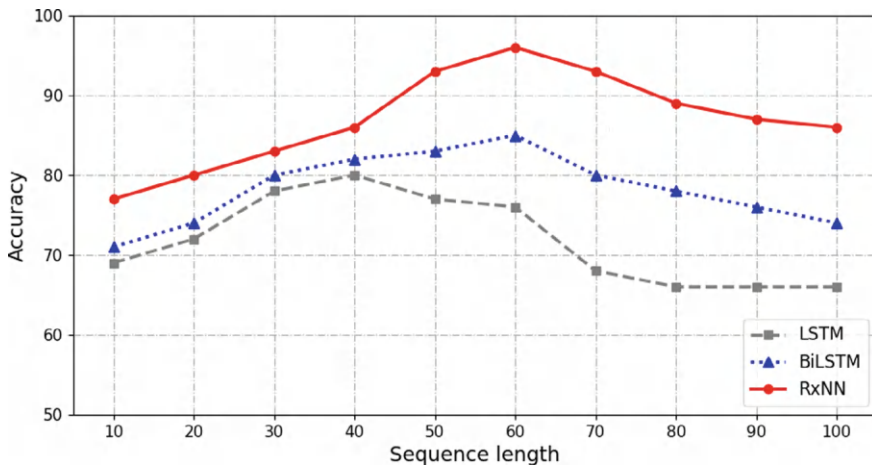


Fig. 3 The impact of sequence length changes on accuracy

4.3 Impact of Sequence Length

The sequence length in this section refers to how many events from the dataset the LSTM network can keep track of at once. To check the model's correctness, we alter the length of the sequence. As seen in Fig. 3, the impact is even worse when the sequence length is longer since LSTM and Bi-LSTM are unable to maintain a certain degree of accuracy. On the other hand, RxNN can only execute at a specific level, and its accuracy increases with increasing sequence length. It is clear that our proposed solution can retain high accuracy while overcoming the limitation of LSTM's inability to handle lengthy sequences.

5 Conclusion

In this study, we present a novel RxNN architecture for log-based anomaly detection that is ideal for detecting network attacks in smart grid systems. The testing dataset correctly depicts the complexity of the actual power generation system and might be used to illustrate the link between physical phenomena and cyber-attacks, which replicates the true status of the power system. The experimental results on the testing dataset show that our proposed technique is feasible, and RxNN can assess anomalies more reliably and effectively than other classic deep learning systems. RxNN has the advantage of potentially replacing human log inspection, allowing for stable and speedy anomaly identification in systems that often create logs as sources of interference and the deployment of suitable countermeasures.

References

1. Kagermann H, Lukas WD, Wahlster W (2011) Industrie 4.0: Mit dem Internet der Dinge auf dem Weg zur 4. industriellen Revolution. VDI-Nachrichten
2. Humayed A, Lin J, Li F, Luo B (2017) Cyber-Physical systems security-a survey. *IEEE Internet Things J*, 4(6)
3. Wang Z, Tian J, Fang H, Chen L, Qin J (2022) LightLog: A lightweight temporal convolutional network for log anomaly detection on the edge. *Comput Netw* 203:108616
4. Qi J, Luan Z, Huang S, Wang Y, Fung C, Yang H, Qian D (2022) Adanomaly: adaptive anomaly detection for system logs with adversarial learning. In *NOMS 2022–2022 IEEE/IFIP Netw Oper Manag Symposium IEEE*
5. Qian Y, Ying S, Wang B (2020) Anomaly detection in distributed systems via variational autoencoders. In: *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, pp 2822–2829
6. Bursic S, Cuculo V, Alessandro D’Amelio (2019) Anomaly detection from log files using unsupervised deep learning. In: *International Symposium on Formal Methods*. Springer, pp 200–207
7. Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. *arXiv:1409*
8. Adhikari U, Pan S, Morris T, Borges R, Beaver J (2014) Industrial Control System (ICS) cyber attack datasets. Dataset 1: Power System Datasets

White-Box Penetration Testing for Hash Collision Attack on Web Applications



Chao-Lung Chou and Hung-Wei Chiang

Abstract Web applications have become essential tools for communication, commerce, and information dissemination in an era where the internet is pivotal in our daily lives. However, as the digital landscape continues to evolve, so do the threats that target web applications. Web application security ensures the integrity, availability, and confidentiality of online services. Hash collisions, though often considered a theoretical concern, can threaten web application security under specific circumstances. A hash collision occurs when two different inputs produce the same hash value when hashed. This paper utilizes a white-box penetration testing approach specifically targeting hash collision vulnerabilities in web applications. Using PHP as our example language, we create an experimental website to demonstrate privilege escalation by exploiting hash collision vulnerabilities.

Keywords Web application security · Hash collision · White-box · Penetration testing

1 Introduction

In the Internet age, most companies have embraced web-based systems to facilitate paperless and real-time office communication. These systems enable rapid information retrieval, enhancing corporate productivity and business capabilities. Web systems frequently handle sensitive data, including personal information, financial records, and computer credentials. Safeguarding this data against unauthorized access is paramount in preventing data breaches and identity theft [1].

C.-L. Chou (✉)

Department of Information Engineering and Computer Science, Feng Chia University, Taichung, Taiwan
e-mail: clchou@fcu.edu.tw

H.-W. Chiang

Department of Computer Science and Information Engineering, Chung Cheng Institute of Technology, National Defense University, Taichung, Taiwan

Web application security encompasses a set of practices, technologies, and strategies designed to safeguard web applications from various threats and vulnerabilities. These threats range from common issues like SQL injection and cross-site scripting (XSS) to more advanced attacks like application layer DDoS attacks and API exploitation. The primary objective of web application security is to protect both the application itself and the data it processes, ensuring the privacy and trust of users. Key components of Web application security include:

- **Firewalls and Intrusion Detection Systems:** These technologies help detect and prevent unauthorized access, malicious traffic, and potential attacks.
- **Authentication and Authorization:** Robust authentication mechanisms ensure that only authorized users can access the application, while authorization controls dictate what actions those users can perform.
- **Data Encryption:** The use of encryption protocols like HTTPS ensures that data transmitted between the client and the server remains confidential and secure.
- **Input Validation:** Properly validating and sanitizing user inputs can prevent common vulnerabilities such as SQL injection and XSS attacks.
- **Regular Updates and Patch Management:** Keeping web application components, frameworks, and libraries up-to-date is crucial to addressing known vulnerabilities.
- **Security Testing:** Employing techniques like penetration testing and vulnerability scanning helps identify and remediate potential security weaknesses.
- **Incident Response and Monitoring:** Having a plan in place to respond to security incidents and continuously monitoring the application for anomalies is essential.

A hash function is a mathematical algorithm that takes an input (or “message”) and returns a fixed-size string of characters, which is typically a hexadecimal number. The output, known as the hash value or hash code, is unique for each unique input. Hash functions are deterministic, meaning the same input will always produce the same hash value. While hash functions are designed to generate unique hash values for different inputs, collisions can occur due to the finite number of possible hash values and the infinite number of potential inputs. As a result, hash collisions are theoretically possible, even with strong hash functions. Implications of hash collisions include:

- **Data Integrity:** Hash collisions can pose a threat to data integrity. If two different pieces of data produce the same hash value, it may lead to data corruption or unauthorized changes not being detected.
- **Password Security:** Hash collisions can undermine password security. Attackers can create different passwords that produce the same hash value, making it easier to crack password hashes.
- **Cryptographic Vulnerabilities:** Hash collisions can lead to vulnerabilities in cryptographic applications. For example, in digital signatures, a collision could enable attackers to forge a signature.

Hash collisions, while generally considered a theoretical problem, threaten web application security in certain circumstances. A hash collision occurs when two

different inputs produce the same hash value when processed through a hash function. Although this phenomenon is uncommon, it is critical to understand how it affects web application security. The hash collision can be classified as cryptographic failure in category A02 in OWASP TOP 10:2021 [2], which is related to cryptography and often leads to sensitive data exposure or system compromise.

One of the most significant areas where hash collisions can be exploited in web applications is password storage. Web applications typically store user passwords as hashed values to protect them from unauthorized access. In a hash collision, an attacker could craft a different password that produces the same hash value as a legitimate user's password. This would allow the attacker to gain unauthorized access to an account without knowing the actual password. This type of attack is known as a "collision attack" and highlights the importance of using strong and collision-resistant hash functions for password hashing.

Black-box testing and White-box testing are two distinct approaches to software testing [3]. Black-box testing focuses on evaluating the functionality and behavior of a system or application without delving into its internal code, structure, or logic [4]. In essence, testers examine the software as a "black box," meaning they do not know its inner workings. White-box testing, also known as Clear-box testing or structural testing, focuses on examining a software application's internal structure, logic, and code [5]. Unlike Black-box testing, where testers evaluate the system solely based on its external behavior, White-box testing involves a detailed inspection of the internal workings of the software. The choice between Black-Box and White-Box Testing for website systems generally depends on project goals and requirements. Black-box testing is ideal for assessing user experience and security, while White-box testing offers insights into code quality and internal functionality. Often, a combination of both methods is employed to leverage their strengths and mitigate their weaknesses, achieving a comprehensive testing approach that ensures the reliability and security of website systems.

This paper adopts a white-box penetration testing methodology, explicitly focusing on identifying and exploiting hash collision vulnerabilities within web applications. Our approach systematically examines the web application's internal structure, source code, and data processing mechanisms. By delving deep into the application's codebase, we aim to uncover potential weaknesses that might be exploited through hash collisions. We have chosen PHP as our illustrative programming language, creating a controlled experimental website to meticulously showcase the possible consequences of hash collision vulnerabilities in the context of privilege escalation. Through this experimentation, we shed light on the risks associated with hash collisions, particularly regarding unauthorized privilege escalation. Our research highlights the theoretical aspects of hash collision vulnerabilities and emphasizes the practical implications for web application security. The experimental results raise awareness among developers, security professionals, and system administrators about the importance of robust security measures to mitigate the risks posed by hash collisions.

The rest of this paper is organized as follows: Sect. 2 reviews the common white-box penetration testing methods. Section 3 presents the proposed method. Section 4 demonstrates the experimental results and Sect. 5 gives the conclusion.

2 Literature Review

P. Tonella and F. Ricca proposed a two-layer model for white-box testing on web applications [6]. This model is designed to enhance the effectiveness and coverage of testing efforts by considering both the static structure of the web application, including HTML and CSS, and the dynamic behavior of server-side scripts. The authors emphasize the importance of comprehensive testing strategies to identify and rectify web application defects, vulnerabilities, and potential security issues.

G. W. Sasmito presented applying the Basis Path Testing technique for white-box testing in a Demography Administration website context [5]. The study aims to enhance the testing process by systematically analyzing the control flow of the web application's source code, identifying critical paths, and executing test cases accordingly. The objective is to ensure the reliability and robustness of the website, contributing to its effective functioning in the field of demography administration.

J. Bau et al. provide a comprehensive overview of the contemporary landscape of automated black-box testing for web application vulnerabilities [7]. The study explores advanced techniques and tools employed in this domain, addressing the challenges and limitations while highlighting the critical role of automated testing in identifying and mitigating security vulnerabilities in web applications.

A. C. Barus et al. developed a prototype for a white-box testing tool using JAVA [8]. The study aims to create an advanced tool for assessing software applications' internal structure and logic, enhancing the efficiency of white-box testing processes. The article delves into the design and implementation of the tool, highlighting its potential contributions to software quality assurance and the identification of defects within software systems.

J. Li et al. introduced a collision attack framework explicitly designed for the RIPEMD-128 cryptographic hash function [9]. This framework delves into the vulnerabilities and potential weaknesses inherent in RIPEMD-128, offering valuable insights into the practical ramifications of collision attacks within the scope of this hashing algorithm.

W. Yu et al. present an innovative collision attack technique directed at the MD5 cryptographic hash function [10]. This research reveals previously undiscovered vulnerabilities and weaknesses inherent in MD5, offering a new perspective on collision attacks within the framework of this hashing algorithm.

Z. Li et al. explore and demonstrate a collision attack aimed explicitly at the NaSHA-384 and NaSHA-512 cryptographic hash functions [11]. This research effectively reveals the presence of vulnerabilities within NaSHA-384/512, shedding light on the practical implications and potential security risks associated with these hashing algorithms.

S. Olga et al. address contemporary hash collision cyberattacks and present strategies for their detection and mitigation [12]. The study explores the evolving landscape of hash collision attacks, emphasizing the importance of staying current with detection and neutralization methods to protect against these threats. This research contributes to cybersecurity by providing insights into effectively countering modern hash collision cyberattacks.

Based on the above literature, both manual and automatic techniques are used to implement penetration testing and discover vulnerabilities; researchers also use different penetration methods to identify defects and vulnerabilities in Web systems to improve security. Hash collision attacks also derive other attack methods due to different encryption methods.

3 The Proposed Method

Penetration testing, often referred to as ethical hacking, is a systematic and methodical approach to assessing the security of a computer system, network, or application. The process typically involves the following steps:

1. Planning and Scope Definition:
2. Information Gathering:
3. Threat Modeling:
4. Vulnerability Analysis:
5. Exploitation:
6. Reporting.

These steps outline the general process that attackers may follow when exploiting hash collision vulnerabilities in web systems. It's important for organizations to be aware of these risks and take measures to mitigate them by using strong and collision-resistant hash functions and implementing proper input validation and security controls.

The flowchart of the proposed White-box penetration testing for hash collision attack is shown in Fig. 1. The proposed method involves the following steps:

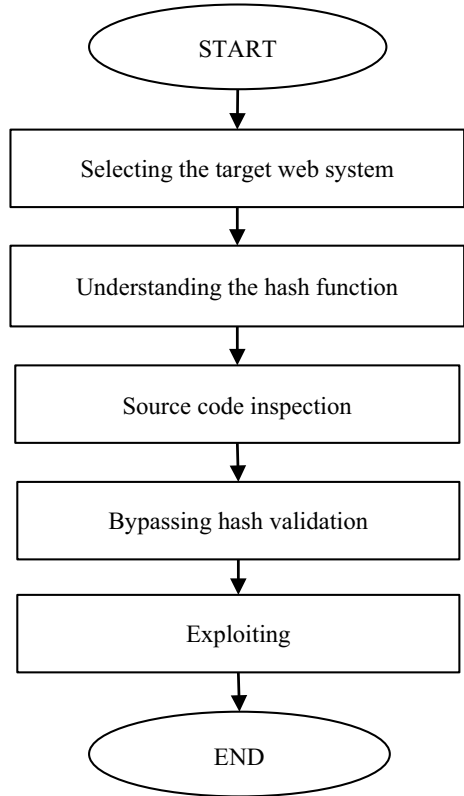
Step 1. Selecting the target web system: The first step is to choose a target web system, which could be any website or application that utilizes hash functions, especially for storing sensitive data.

Step 2. Understanding the hash function: The hash function employed within the target system must be known in advance, including its algorithm and characteristics.

Step 3. Source code inspection: Conduct source code inspection to find the token that generates the hash function.

Step 4. Bypassing hash validation: The proposed method is to directly replace the hash function token so that when the system performs cryptographic authentication, the hash function value will automatically be deemed to have completed qualified verification.

Fig. 1 Flowchart of the proposed method



Step 5. Exploiting: Write scripts to exploit vulnerabilities to gain entry into systems.

Figure 2 shows an example in a sample PHP source code with the ‘\$token = random_int ‘ in which ‘random_int’ is mainly used to generate cryptographically secure random hash numbers in PHP. This value is large enough that ordinary brute-force attacks are unlikely to succeed.

When PHP processes hash strings, it will use ‘!’ = ‘ or ‘ = = ‘ to compare hash values. It interprets every hash value starting with “0E” as 0, so if two different passwords are hashed, their hash values begin with “0E”, then PHP will think they are the same; both are 0. Figure 3 shows a reset link function on this web page. After checking the source code, it is found that eight characters are truncated and retained. And it was found that “ = = \$sig” exists. The ‘ = = ‘ parameter is a type comparison in the PHP language. Since the source code displays “ = = \$sig,” that can be bypassed by controlling “\$sig” to develop an attack code that identifies parts starting with “0e” as scientific symbols.

Figure 4 shows an attack script example. Appending eight characters ‘0e000000’ in the program code, and controlling the part of “\$sig” starting with “0e” to be recognized as a scientific symbol for the attack.


```

src > app > Auth > Auth.php
46         return User::where('email', $email)->first();
47     }
48
49     public function generateResetLink($user){
50         $token = random_int(1000000000000000, 999999999999999);
51         $user->resetToken = $token;
52         $user->save();
53
54         $time = time();
55
56         $rate = RateLimit::where('userId', '=', $user->id)->first();
57         if(!empty($rate)){
58             $rate->touch();
59         }else{
60             $rate = new RateLimit();
61             $rate->userId = $user->id;

```

Fig. 2 Random hash number generator function (random_int) in PHP

```

Auth.php X
src > app > Auth > Auth.php
72
73     public function checkResetLink($userId, $time, $sig){
74         $user = User::find($userId);
75         if(empty($user->resetToken)){
76             return false;
77         }else{
78             return $this->generateSig($user->id, $user->resetToken, $time, $user->password) == $sig;
79         }
80     }
81
82     public function generateSig($id, $resetToken, $time, $password){
83         return substr(hash("sha256", $id . "|" . $resetToken . "|" . $time . "|" . $password), 0, 8);
84     }
85

```

Fig. 3 Reset link function in PHP

```

7     target = "http://192.168.148.132"
8     iSession=requests.session()
9     toTestToken = target + "/reset/1/{}/0e000000"
10    negativeFlag = "Reset arguments do not match user"
11    threadSum = 10
12    finishedFlag = False
13
14    class workerThread (threading.Thread):
15    def testToken(self,timeToken):
16        global finishedFlag
17        r = iSession.get(toTestToken.format(timeToken))
18        if(not negativeFlag in r.text):
19            with open("res.txt","w") as f:

```

Fig. 4 Appending eight characters '0e000000' in the script

4 Experimental Results

The experiment was conducted using a notebook with Linux 5.10.0-kali3-amd64 operating system. The software includes PHP 7.4.14 and MySQL 5.7.29-0ubuntu0.18.04.1. The test web homepage is shown in Fig. 5.

The experiment first used OWASP ZAP [13] and Nikto [14] to conduct black-box testing on the test website. OWASP ZAP (OWASP Zed Attack Proxy) is a tool provided and developed by the OWASP organization. It is a penetration testing tool for web applications. It uses proxies to simulate attacks on web systems and analyze their security vulnerabilities. It is not limited to use on a single system and set up as a man-in-the-middle. Any interaction between client and server will be analyzed and scanned by ZAP. Nikto is an open-source network security scanning tool developed using Perl language. It is not limited to use on a single system. It is a built-in tool in Kali Linux. It can automatically test web structures, frameworks, application configurations, and vulnerabilities and detect websites. Whether out-of-date software versions are used can help penetration testers quickly identify potential security issues. Table 1 shows the same results scanned by both tools. After passing Black-box testing, it was found that neither OWASP ZAP nor Nikto found the hash collision vulnerability but found some configuration settings vulnerabilities.

In this test website, the administrator account is ‘Testman@Testlab.com’ (displayed on left below of Fig. 5). First, embed the administrator account into the attack script. Using this attack script, it is possible to successfully obtain the password reset link, as shown in Fig. 6. By using this link, the password can be reset. With the successfully reset password, login is achievable, as depicted in Fig. 7. Once administrator privileges are obtained, subsequent attacks can be launched to gain server access and potentially propagate laterally.

The following findings were made during the experiment:

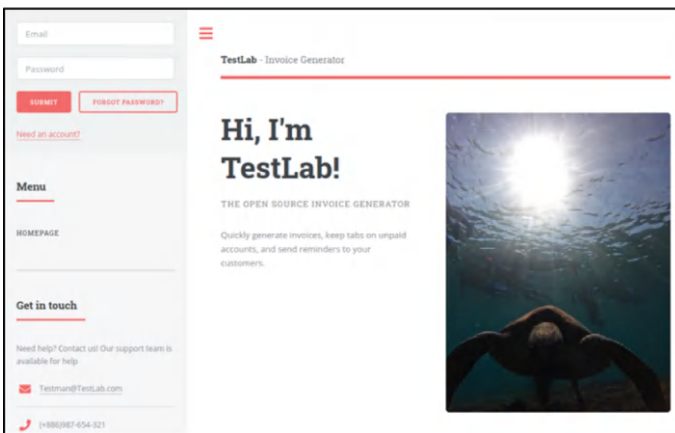


Fig. 5 The test website homepage

Table 1 Scanning results of test website with ZAP and Nikto

No	Vulnerability type
1	.htaccess Information Leak
2	Missing Anti-clickjacking Header
3	The X-Content-Type-Options header is not set
4	Cookie PHPSESSID created without the http only flag
5	Retrieved x-powered-by header

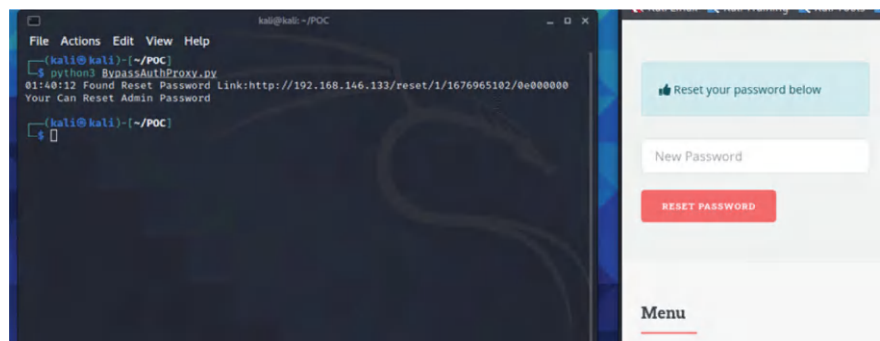


Fig. 6 Successfully to obtain reset password link

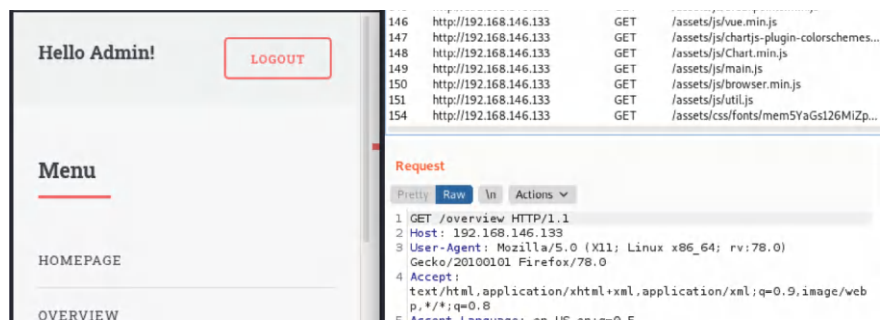


Fig. 7 The Admin password was successfully exploited on the test website homepage

1. The attack may fail if the target system administrator account is unavailable.
2. The default number of login attempts for the website is 100; this limit may need to be shortened and adjusted according to actual needs to prevent possible attacks.

In summary, in the process of web system development, it is essential to ensure password security and input verification. The best practice is to use white-box testing to review and improve these aspects to improve the safety and maintainability of your source code.

5 Conclusions

We introduce a white-box penetration testing method to evaluate hash collision attacks on a sample website. Our experimental results clearly illustrate that traditional black-box testing methods fail to assess hash collision attacks effectively. Therefore, adopting white-box testing approaches becomes imperative to enhance the overall security of the website system. To safeguard web application systems from Hash Collision vulnerabilities, several preventive and mitigation measures can be implemented:

1. Utilize Stronger Hash Functions: Opt for robust hash functions like SHA-256 or SHA-3 instead of weaker alternatives such as MD5 or SHA-1.
2. Implement Salted Hashes: Enhance security by incorporating salted hashes, where a unique random value (salt) is added to each input before hashing.
3. Enforce Stringent Input Validation: Prevent malicious inputs or crafted payloads from reaching the hashing process through comprehensive input validation. Sanitize user inputs to eliminate potentially harmful characters.
4. Apply Proper Output Encoding: Before rendering hashed values in web pages or APIs, ensure proper output encoding to prevent potential script injection attacks.
5. Conduct Routine Security Assessments: Proactively identify and address Hash Collision vulnerabilities by performing regular security assessments, including vulnerability scanning and penetration testing.

Acknowledgements This work was supported by the National Science and Technology Council (NSTC), Taiwan, R.O.C., under Grant 112-2221-E-606-009-MY2.

References

1. Sharma A, Singh A, Sharma N, Kaushik I, Bhushan B (2019) Security countermeasures in web based application. In: 2nd IEEE International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT). pp 1236–1241, Kannur, India
2. OWASP Top Ten Web Application Security Risks, <https://owasp.org/Top10/#welcome-to-the-owasp-top-10-2021>
3. Wylie PL, Crawley K (2021) What is a pentester?. the pentester blueprint: starting a career as an ethical hacker, Wiley
4. Palma SM, I., Martins E. (2015) A black-box approach to detect vulnerabilities in web services using penetration testing. IEEE Lat Am Trans 13(3):707–712
5. Sasmitho GW (2020) White box testing with basis path technique in the demography administration website. In: 2020 IEEE International Conference on Sustainable Engineering and Creative Computing (ICSECC), pp 86–92, Indonesia
6. Tonella P, Ricca F (2004) A 2-layer Model for the White-box Testing of Web Applications. In: Sixth IEEE International Workshop on Web Site Evolution, pp 11–19, Chicago, IL, USA
7. Bau J, Bursztein E, Gupta D, Mitchell J (2010) State of the art: automated black-box web application vulnerability testing. In: 2010 IEEE Symposium on Security and Privacy, pp. 332–345, Oakland, CA, USA

8. Barus AC, Hutasoit DIP, Siringoringo JH, Siahaan YA (2015) White box testing tool prototype development. In: 2015 International Conference on Electrical Engineering and Informatics (ICEEI), pp 417–422, Denpasar, Indonesia
9. Li J, Zeng G, Yang Y (2020) Collision attack framework on RIPEMD-128. In: 2nd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM), pp. 321–324, Manchester, United Kingdom
10. Yu W, Jianhua C, Debiao H (2009) A new collision attack on MD5. In: 2009 International Conference on Networks Security, Wireless Communications and Trusted Computing, pp. 767–770, Wuhan, China
11. Li Z, Jiang H, Li C (2010) Collision attack on NaSHA-384/512. In: 2010 International Conference on Networking and Information Technology, pp 243–246, Manila
12. Olga S, Larissa C, Nikita L, Pavel R, Vladislav C, Boris A, Andrey L (2021) Modern hash collision cyberattacks and methods of their detection and neutralization. J Phys: Conf Ser 2131:022099
13. OWASP Zed Attack Proxy, <https://www.zaproxy.org/>
14. Nikto, <https://cirt.net/Nikto2>

Towards Zero Trust for Financial Sectors: A Proposed Framework on Trust Evaluation



Yu-Chih Wei and Tak-Wai Yu

Abstract The rapid technological advancements in the financial sector necessitate innovative strategies to safeguard sensitive data against emerging threats. In this context, the Zero Trust security model, which mandates stringent verification for every access request on a per-session basis, emerges as a viable approach, offering a significant enhancement over traditional network perimeter-based security models that predominantly rely on network location for granting access. This paper introduces a trust evaluation model that seamlessly integrates with the Zero Trust framework, targeting to bolster the financial sector's defenses. Our model demonstrates a potential pathway for evolving trust management within zero trust architectures, and thereby fostering a more secure and reliable operational environment for financial institutions.

Keywords Zero Trust · Financial sectors · Trust evaluation

1 Introduction

1.1 State of Cybersecurity in Financial Sectors

In recent years, the financial sector has undergone a substantial digital transformation, escalating the complexity of cybersecurity challenges. This sector's reliance on increasingly sophisticated digital technologies has exposed emerging vulnerabilities and heightened the risk of cyber threats, necessitating a revolutionary approach to security and trust management. Traditional security mechanisms, which once stood as

Y.-C. Wei · T.-W. Yu (✉)

National Taipei University of Technology, (10608) 1, Sec. 3, Zhong-Xiao (Chung-Hsiao) E. Rd., Da'an Dist., Taipei City 106, Taiwan (R.O.C.)

e-mail: t11lab8407@ntut.edu.tw

Y.-C. Wei

e-mail: vickrey@mail.ntut.edu.tw

the guardians of data and resources, are revealing their limitations, making it evident that a paradigm shift is vital to maintaining financial stability and safeguarding critical information in the digital age.

At present, the financial sector is navigating a landscape fraught with considerable cybersecurity breaches that bring about severe financial consequences. A notable instance highlighting the vulnerabilities in the sector is the 2016 Bitfinex cryptocurrency exchange breach. During this event, hackers managed to steal nearly USD 72 million worth of Bitcoin [1], demonstrating the pressing vulnerabilities present within cryptocurrency exchanges.

Furthermore, the complex sphere of cyber threats has expanded to include politically motivated attacks led by hacktivist groups. As evidenced in 2022, these groups utilized resources from the dark web to orchestrate significant attacks on various Russian establishments, including the Moscow Stock Exchange and Sberbank [1]. These assaults illustrate the potent potential of hacktivist activities to instigate financial instability and other large-scale disruptions.

These events underline the critical need for a dynamic and adaptive security strategy in the financial sector, capable of evolving with the rapid advancements in technology and the ever-shifting threat landscape. The strategy should anticipate and counter not only financial crimes but also politically driven cyber-attacks with significant geopolitical repercussions.

1.2 Zero Trust Architecture: A Paradigm Shift

To address these pressing issues, the Zero Trust Architecture (ZTA) emerges as a promising solution, offering a radical departure from the ‘trust-but-verify’ approach that has historically characterized cybersecurity efforts. This new paradigm shifts towards a ‘never-trust-always-verify’ strategy, where trust evaluations and identity authentications are continuous and pervasive [2], ensuring a more robust defense against both external and internal threats.

1.3 Research Objective

Despite the evolving discourse on cybersecurity, there remains a research gap in the establishment and evaluation of Zero Trust frameworks specifically targeting the financial sector. This study aims to address this gap by proposing an augmented Zero Trust model tailored to the financial sector, with a particular focus on trust evaluation implementation. The objective is to encourage collaborative engagement in both anticipating and mitigating potential future threats, thereby fostering a more fortified and resilient financial ecosystem.

2 Related Work

2.1 *Cyber Threats in Financial Sectors*

Security threats targeting financial sectors have escalated at an alarming rate in recent years, becoming more sophisticated and destructive, primarily due to the pandemic-induced reliance on remote services. These threats are multifaceted, and evolving at the same time where institutes are struggling to keep up their defense level.

Theft of Financial Data and Customer Privacy Information. Financial sectors must grapple with large-scale monitoring and theft of sensitive data. This category encompasses the breach of confidential financial data and customer privacy information. These breaches often involve the exploitation of vulnerabilities in digital systems, which compromise sensitive financial data, potentially disrupting financial operations. One common avenue through which these vulnerabilities are exploited is phishing attacks. Cybercriminals use phishing emails to deceive employees or customers into providing login credentials or accessing malicious links. Malware, including ransomware attacks, falls under this category, which can severely impact financial data by infiltrating accounting systems to steal or encrypt critical financial records [3], thereby posing significant risks to the accuracy, availability, and integrity of accounting information.

Malicious Data Tampering of Financial Records. An emerging concern in the financial sector is the malicious tampering of financial records. This involves the alteration or manipulation of financial data to create inaccuracies or fraudulent transactions, jeopardizing the integrity and accuracy of financial records. Attack vectors such as broken authentication and cross-site scripting enable unauthorized external access to financial systems [3], allowing attackers to manipulate financial data or carry out fraudulent transactions. This leads to unauthorized access, financial fraud, or the compromise of sensitive financial information. Additionally, the risk of malicious data tampering is not limited to external threats; insider threats pose a significant risk as well. Employees or contractors with authorized access to financial systems could exploit their privileges to manipulate data for personal gain or other malicious purposes. The insider threat amplifies the risk, as these individuals already have knowledge of the internal systems and may more effectively bypass security measures, thereby further endangering the integrity and accuracy of financial records.

2.2 *Cybersecurity in Financial Sectors*

Integrated Control, Intelligent Defense, and Dynamic Monitoring. Financial industries mandate an integrated control system, fortified with smart, adaptive defense mechanisms capable of adjusting to a broad spectrum of cybersecurity threats and an ever-changing digital world. Key to this is the system's ability to flexibly adapt to diverse interaction protocols and application contexts within the financial arena.

Furthermore, ongoing monitoring and possible reauthentication or reauthorization are embedded within a policy framework targeting a balance between security, availability, user-friendliness, and cost-effectiveness [2]. Anchoring this approach is the inclusion of specialized tools such as Identity, Credential, and Access Management (ICAM) as well as advanced security analytics, enabling a tailored, real-time defense across a variety of financial operations and services.

Scalable Identification and Authentication Management Mechanisms. To secure the multitude of daily transactions in the financial sectors, the imperative is for a unified yet scalable identification and authentication management framework. This system is architected to meet a broad array of security needs, extending from individual users to emerging Internet of Things (IoT) devices that are increasingly integral to contemporary financial infrastructures [2]. The system is backed up by policy guidelines that establish minimum criteria for resource accessibility. These policies integrate key components such as MFA, geographically based network limitations, and asset configuration. Developed collaboratively by data custodians and business process administrator, this architectural approach is critical for ensuring user identity traceability while minimizing unauthorized access and data tampering.

2.3 Zero Trust Architecture: The Concept and Core Components

In the face of an expanding enterprise landscape dispersed across multiple platforms and clouds, traditional peripheral access controls fall short, marking the dawn of a pivotal shift where trust is never assumed, whether from internal or external entities. Zero Trust Architecture dismantles the hierarchical approach of past decades, bringing a resilient, segmented, and parallelized structure controlled centrally, meeting the complex demands of modern threat landscapes and compliance needs [2].

Policy Decision Point (PDP). PDP is responsible for dictating access grants through calculated trust scores and algorithmic evaluations, it also acts as a facilitator of decision executions, fostering seamless communication between subjects and resources.

Policy Enforcement Point (PEP). PEP serves as the guardian of the trusted zone, ensuring secure connections based on the directives of the PA.

These elements are intricately supported by a network of Policy Information Points (PIPs), offering critical data across domains including endpoint security and data protection, aiding the Policy Decision Point (PDP) in making well-informed access verdicts. A high-level demonstration of relations between ZTA logical components is shown in Fig. 1 [3], where we take the reference structure from the NIST Special Publication 1800–35 series.

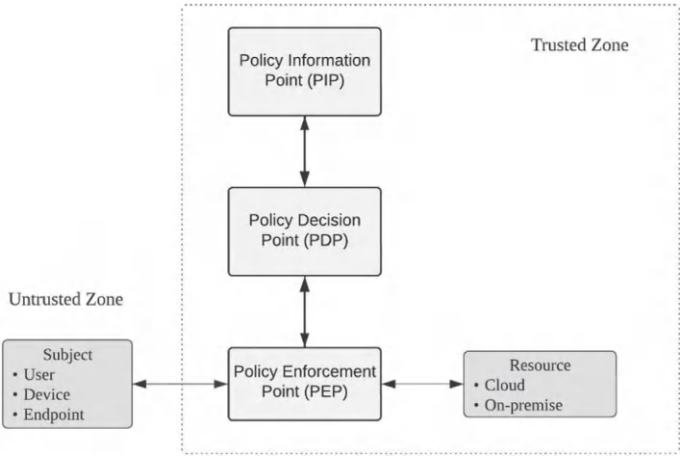


Fig. 1 A High-Level demonstration of zero trust architecture

3 Proposed Framework

The proposed Zero Trust based evaluation framework serves as a centralized component in realizing continuous trust evaluation capabilities within the financial sector. The part interacts with an access control engine, offering a continual feed of critical data that influences the formulation of access control policies, grounded on the comprehensive analysis conducted within this security framework.

The model incorporates input parameters including user identity attributes, credential security, and system behavior analysis for nuanced trust evaluations. It harmonizes the security credentials of objects with the trust ratings of subjects, executing continuous assessments and adapting swiftly to the dynamic landscape of the financial sector.

3.1 Pillars of Trust Evaluation Framework

In this section, we discuss the four foundational pillars of the proposed zero trust framework, which combined provide essential and abundant information to the Environment Monitor, to achieve the goal of robust trust evaluation. The descriptive flow of trust evaluation in an environment monitor is shown in Fig. 2.

Subject. The “Subject” pillar represents entities initiating network and resource access requests. Subjects encompass identity-verified financial professionals, employees, customers, networked financial systems, and online applications connected to personal identity or device verification. The determination of trust levels should be facilitated through real-time analysis of diverse parameters.

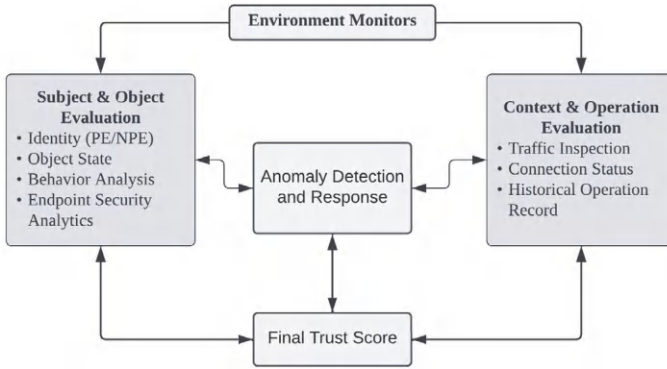


Fig. 2 Four pillars of trust evaluation framework

Object. The “Object” pillar demarcates the resources that are focal points of access requests within a financial information system. These resources encapsulate critical financial data, intelligent service functions, and service interfaces. It is imperative to impose restrictions on these objects to safeguard system security and data privacy, allowing access exclusively to pertinent and verified subjects.

Context. The “Context” pillar scrutinizes the situational security prevailing during network and resource access requests and processes. The context encompasses the physical, computing, and network landscape where financial transactions and data access occur. Augmenting the security attributes of various contexts strengthens the security infrastructure, facilitating the evaluation of potential risks associated with different access points and network domains.

Operation. The “Operation” pillar integrates real-time security analysis based on historical and current operational patterns concerning network activity and resource access. Continuous monitoring and analysis of dynamic operations involving subjects, objects, and contexts are critical to fostering a reliable and secure ecosystem, proficient in adjusting to changing conditions that could potentially affect the security and privacy aspects of financial services and data exchanges.

3.2 *Environment Monitor Candidate Components*

In this section, we discuss tools that play a vital role in realizing the goal in dynamic, continuous monitoring and access control process. Together these components build up the environment monitor responsible for robust and consistent trust evaluation.

Identity, Credential, and Access Management (ICAM). ICAM coordinates the establishment and oversight of both enterprise user profiles and device accounts. It assists identity databases, role specifics, and access characteristics that act as the cornerstone in making informed access decisions within an organization. This systematic process ensures that appropriate entities receive the suitable permissions

to access required resources precisely. It also manages the authorization dynamics by undertaking diligent user and device verifications, utilizing information about identity, roles, and access rights to accurately judge the validity of various access requests [3], and functioning as a gatekeeper that sanctions or prohibits access based on a comprehensive evaluation.

Security Incident and Event Management (SIEM). SIEM plays a role in shielding the enterprise environment by aggregating and synthesizing security information and event data from various sources. It employs analytical techniques to correlate the collected data, aiding in the identification of abnormalities in the system, thereby pinpointing possible threats and vulnerabilities [3].

Threat-Intelligence. Cyber Threat-Intelligence is responsible for leveraging data on current or forthcoming vulnerabilities, attacks, and other dangers impacting business operations and assets to guide decisions on how to protect against and address these threats [3].

3.3 State of ML with Environment Monitor Components

During the identification phase, Convolutional Neural Networks (CNNs) can be utilized to fine-tune attribute recognition and block spoofing attempts. In the authentication step, biometric solutions, backed up by neural networks and multi-factor authentication systems, are emerging as potent replacements for traditional passwords, offering both security and user convenience. During the authorization step, reinforcement learning could be utilized to mitigate vulnerabilities and streamline ICAM processes [4].

Recent strides in the cybersecurity domain are characterized by the fusion of ML with Security Information and Event Management (SIEM) systems, aiming to enhance the efficacy of cyber-attack detection and prevention. Research, including that by Anumol et al. [5], indicates that leveraging ML algorithms with SIEM systems can proficiently analyze network traffic data, identifying anomalous patterns indicative of a cyber threat, thereby improving real-time monitoring and alerting mechanisms. The use of open-source tools in developing these systems offers a cost-effective solution, enabling enterprises to strengthen their cybersecurity infrastructure without substantial investments.

A noteworthy proposal in leveraging AI and ML in the threat-intelligence domain involves a systematic approach to extracting actionable intelligence from various sources, such as STIX or TAXII, alongside unstructured sources like forums and blogs owned by suspicious hackers [6]. An applicable model has been proposed that suggests utilizing the Naïve Bayes classifier, a reliable algorithm for text classification in this context. The approach fosters an independent feature analysis, ensuring that each feature contributes to the prediction independently, enhancing the reliability of threat detection.

4 Security Realization with Zero Trust

The tailored model proposes the deployment of access management technologies for identity and privilege regulation. The approach is complemented by the creation of a foundational security database that continuously tracks and ensures the operational stability of both human and non-human entities connected to the system.

Embracing a security reinforcement approach in financial institutions would entail meticulous access vetting procedures, including the implementation of granulated access controls, specialized isolation protocols, and proactive authentication and authorization mechanisms before initiating any network connections. Special attention is focused on securing data transmission channels and safeguarding the integrity of open-source code repositories to mitigate risks of unauthorized access and exploitation.

Integrating a zero trust framework into financial systems necessitates the utilization of sophisticated technologies to enable secure data collaboration. By setting up identity-based logical perimeters and enacting adaptive security policies, the system is designed to dynamically respond to emerging threats. This reduces the vulnerability associated with human errors while streamlining security workflows.

4.1 Flow Overview

Figure 3 shows a brief overview of how Zero Trust Architecture operates for resource protection. The focus is put after access is granted, the core of ZT architecture is that it doesn't rest but proactively manages the session, continuously monitoring and reevaluating every request made by the user.

The overall architecture is a multi-layered process where the PEP relays information to the PDP which acts as the central command, receiving ongoing reports, performing analysis against the enterprise policies, and determining whether to continue, adjust, or end a session, especially when abnormal behavior is detected.

This process is a live conversation that constantly asks, if the subject is still compliant, The PEP issues periodic challenges to users, verifying their credentials and the security of their devices, much like spot checks in a secure facility. Should anything change such as a device falls short of hygiene standards, the PDP sends the command of modifying or revoking access to protect the network. The continuous cycle of authentication and verification, reflecting a commitment to security that never rests [3].

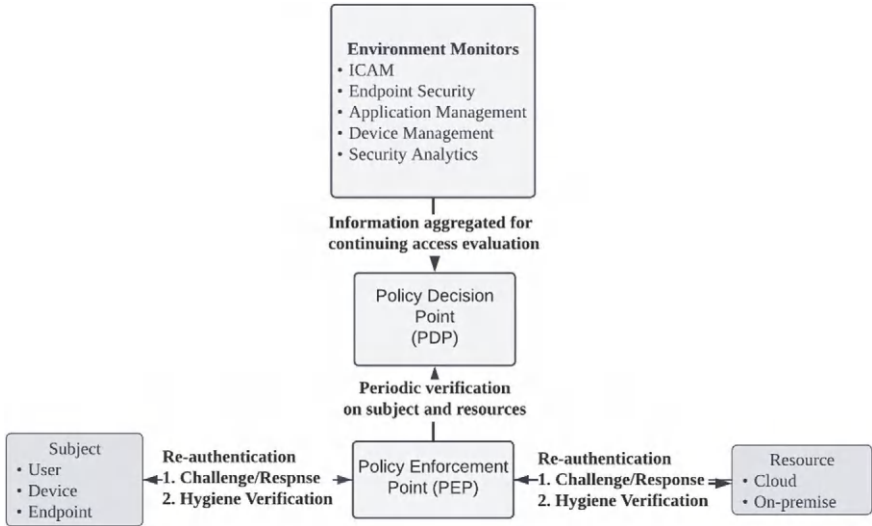


Fig. 3 Framework flow demonstration

5 Framework Evaluation

The proposed evaluation of the proposed Zero Trust framework for the financial sector necessitates a rigorous and informed review process, considering the challenges of acquiring real-world data from this industry due to its restrictions and the sensitive nature of the information. To bridge this gap and ensure the practicality of the framework, an expert review is integral during the development of the framework [7].

5.1 Participant Selection

The first step of the process is focusing on the selection criteria for participants. Choosing cybersecurity professionals with direct experience in financial sectors and familiarity with Zero Trust architectures is essential to proceed with subsequent procedures. A diverse mix of roles including IT security analysts, CISOs, network administrators, compliance officers, and cybersecurity policy makers should be included to ensure a broad range of perspectives.

The selection process involves meticulous background research on potential candidates, including reviewing their participation in cybersecurity conferences and scrutinizing their profiles if available via internet, complemented by the consideration of referrals. This due diligence is crucial in forming a panel with the depth and breadth of knowledge required for a substantive evaluation.

5.2 Interview Preparation

Before commencing the interview, it is crucial to develop an interview guide. The guideline highlights the need for questions that target specific aspects of Zero Trust relevant to the financial sector, including technical implementation, regulatory compliance, and user experience. The guide should be designed to allow follow-up questions for more in-depth exploration. Settings that ensure privacy and confidentiality given the sensitive nature of security practices should also be taken into consideration.

5.3 Semi-structured Interviews

Conducting a semi-structured interview provides the experts with the flexibility to delve into the framework and discuss additional ideas and insights that could be pertinent. This phase is not only about validating the existing components of the framework but also about uncovering any critical elements that may be absent. Each interview is carefully recorded and transcribed, and the transcripts are validated with the experts to maintain the integrity of the information gathered [7].

The feedback from these initial discussions is then analyzed, and the framework is expected to be refined accordingly. Thematic analysis is employed to categorize the feedback into discernible themes, which are then used to enhance the framework. Inputs at this stage are derived from expert opinions and assessments on Zero Trust elements and their applicability in the financial sector. The resulting output is a preliminary assessment framework that identifies key areas and potential gaps.

5.4 Interview Design

The research report conducted by RAND Corporate [8] provides an overview of semi-structured interviews methodologies. The report provides guidance on question development in semi-structured interviews, emphasizing the importance of question design and categorizes questions into different types, including descriptive, structural, and contrast questions.

Examples of questions are depicted in Table 1.

Table 1 Question type and sample for semi-structured interview [8]

Question type	Sample question	Rationale
Descriptive	■ What is a specific instance that you implemented or interacted with a Zero Trust framework in your organization?	Open-ended type that allows respondents to provide rich, in-depth information
Contrast	■ How would you contrast the effectiveness of Zero Trust frameworks with traditional security models in preventing data breaches in the financial sector?	Highlight distinctions, evaluating pros and cons, or understanding varying viewpoints on a topic
Structural	■ How do regulatory standards that a Zero Trust model align within the financial sector fit into each other?	Explore how different elements are related to each other, helping the researcher categorize and understand the structure of experiences, processes, or concepts

6 Limitation and Future Work

It is crucial to recognize potential challenges, including data privacy issues and the complexities involved in effectively implementing various machine learning algorithms, as well as concerns raised on obtaining real industry data to conduct testing and evaluation for financial sectors due to restriction and confidentiality. A possible future refinement of this model might involve the incorporation of additional tools into the trust evaluation engine for a more descriptive and comprehensive model, also the inclusion of other logical components that collectively form an all-inclusive Zero Trust Architecture, upon feedback from the evaluation through interviews conducted.

7 Conclusion

The Zero Trust model opens path for supplementing financial sector security, merging continuous monitoring and adaptability for a proactive approach in the cybersecurity field. The model incorporates trust evaluation as one of its core, targeting to build a simulated metric which promises to be a vital tool in enhancing the efficacy of the security system. With a sustained focus on improvement and adaptation, the model aims to remain abreast of cybersecurity advancements, combining technological innovation and expertise to mitigate evolving cyber threats.

References

1. Olivér Gulyás GK (2023) Impact of cyber-attacks on the financial institution. *ScienceDirect* 219:84–90
2. Rose S, et al. (2020) NIST special publication 800–207: zero trust architecture
3. NIST special publication 1800–35B: implementing a zero trust architecture third preliminary draft, N.C.C.o. Excellence, Editor
4. Sara Aboukadri AO, Abdellatif Mezrioui (2022) Major role of artificial intelligence, machine learning, and deep learning in identity and access management field: challenges and state of the art. In: *Proceedings of the 8th International Conference on Advanced Intelligent Systems and Informatics*. Springer
5. Muhammad AR, PS, AAW (2023) Integrated Security Information and Event Management (SIEM) with Intrusion Detection System (IDS) for live analysis based on machine learning. In: *4th International Conference on Industry 4.0 and Smart Manufacturing* pp 1406–1415
6. Salil Kanhere VTP, Shamik Sural (2020) An overview of cyber threat intelligence platform and role of artificial intelligence and machine learning. In: *16th International Conference, ICISS* M.S. Gaur, Editor. Springer
7. Yeoh W, et al. (2023) Zero trust cybersecurity: Critical success factors and A maturity assessment framework. *Comput & Secur*, 133
8. Margaret C Harrell MAB (2009) Data collection methods. semi-structured interviews and focus groups, R.N.D.R. Institute, Editor

FiT-DPKI: Decentralized Public Key Infrastructure with Flexibility and Transparency for IoT Networks



Ruei-Hau Hsu, Jia-Yin Guo, Bing-Cheng Ke, and Chia-Ho Tan

Abstract The security of Internet of Things (IoT) to achieve identity identification and communication secrecy is the most critical issue to be confronted. To achieve the two security features, authentication of public keys by digital certificate techniques establishes the root of trust. To overcome the deficiencies of Certificate Authority-based Public key Infrastructure (CA-based PKI) on generating certificates and certifying, such as single point failure, non-transparency, and dependency, this paper proposes a decentralized PKI based on blockchain, referred to as FiT-DPKI. FiT-DPKI minimizes the dependency of central components and allows the devices in IoT network to authorize the issuance and revocation of certificates, together to avoid single point failure. For the setting of dynamic security level, it employs threshold signatures so that sufficient number of nodes are required to approve generation of new certificates. For transparency, it combines blockchain so that the nodes can jointly maintain the credential system for IoT networks.

Keywords Public-key infrastructure · Internet of Things · Certificate authority · Three-party authentication · Threshold signatures · Blockchain

1 Introduction

With the development of 5G networks and artificial intelligence, the Internet of Things (IoT) is at the core, and the growth rate of IoT devices is constantly increasing. However, due to hardware limitations, most IoT devices lack sufficient computing resources, rendering IoT devices a hotbed for attackers. The IoT device will exchange private data in various applications. Whether the data is stolen or tampered with by an attacker, or an attacker impersonates and gains control of the device without identity verification or proper authorization, it causes serious harm. To solve these problems,

R.-H. Hsu (✉) · J.-Y. Guo · B.-C. Ke · C.-H. Tan

Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan

e-mail: rhhs@mail.cse.nsysu.edu.tw

IoT applications must generate confidentiality for data transmission. To encrypt data to the correct entity side, the IoT devices must be able to authenticate each other and confirm the communicating entity is correct. Relying on the certificate authority(CA), public key infrastructure (PKI) is one commonly used methods. However, there are some security issues with traditional PKI. In a traditional PKI, the certificate of each device will be created, and it can be imagined as the national identity card issued by the country. We can use the identity card to indicate our identity. For instance, when I trade with Alice and she is unable to provide her identity card, I doubt the identity of the other party and interrupt the transaction. Now the certificate can guarantee her identity and corresponding public key, perform secure authentication, and use public-key cryptography to achieve Confidentiality, Integrity, Authenticity, and Non-repudiation. The current common PKI method is based on CA (X.509 standard) [6].

In this standard, PKI depends on a powerful and fully trusted certificate authority to provide sufficient trust for users by issuing certificates as proofs of trust. Trust in the CA implies trust in the certificates it issues. However, the fact that central infrastructure is an essential to maintain services for users in public networks results in several types of vulnerabilities such as single point of failure [15], non-transparency, and disconnection with CA.

The reasons the above vulnerabilities are critical to security: (1) Availability to IoT network is a critical issue; (2) CA hijacking may occur more frequently than before; (3) Delay of services in IoT networks implies failed services. The traditional CA-based PKI is unsuitable for current IoT network environment based on the disadvantages mentioned. This research aims to address these disadvantages by designing an applicable PKI architecture for IoT networks. We propose a PKI architecture, mainly used in small or medium sized IoT fields, such as schools and hospitals. The IoT network is composed of IoT devices that perform the work of PKI. We use blockchain and threshold signature technologies to modify the traditional centralized PKI into a decentralized PKI to address the single point of failure problem. This scheme enables a certificate to be generated by several nodes rather than one node. With the transparency of blockchain, the nodes can review the chain and jointly maintain the certificate system. With the non-tampering of blockchain, the security of the certificate system increases. With the consensus of blockchain, each node stores the certificate information and maintains its consistency. Therefore, it is not necessary to rely upon an external network connection during authentication.

2 Preliminaries

2.1 Bilinear Mapping

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be a cyclic additive group and a cyclic multiplicative group of prime order p , respectively. A bilinear mapping $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ satisfies the following properties [3] in which P is a generator of $\mathbb{G}_1$. **Bilinearity** satisfies

$e(aP, bQ) = e(P, Q)^{ab}$, $e(P_1 + P_2, Q) = e(P_1, Q) e(P_2, Q)$, and $e(P, Q_1 + Q_2) = e(P, Q_1) e(P, Q_2)$, $\forall P, P_1, P_2, Q, Q_1, Q_2 \in \mathbb{G}_1$ and $a, b \in \mathbb{Z}_p$, **Non-Degeneracy** satisfies that the bilinear map function does not map all pairs in $\mathbb{G}_1 \times \mathbb{G}_1$ to the identity of $\mathbb{G}_2$ since $\mathbb{G}_1, \mathbb{G}_2$ are groups of the same prime order, it implies that if P is a generator of $\mathbb{G}_1$ then $e(P, P)$ is a generator of $\mathbb{G}_2$, and **Computability** satisfies that there exists an efficient algorithm to compute $e(P, Q)$, $\forall P, Q \in \mathbb{G}_1$.

2.2 Gennaro et al.'s Threshold Signature Scheme

Gennaro et al. proposed a threshold signature that can dynamically add members [5]. Later Wang et al. improved it by enhancing efficiency and reducing space complexity [14]. The proposed decentralized PKI adopts this threshold signature scheme as building block in the design of the security protocols. The signature scheme consists of five polynomial-time computable algorithms, i.e., DKGen, SFGGen, SReCon, Verify, and PtAdd, where the proposed scheme introduces the details of the algorithms.

3 Related Work

The Google Certificate Transparency project [2] was proposed to improve the current PKI architecture. This solution requires many independent servers to maintain public, append-only logs for users to check if there are wrong certificates. Although it has improved transparency, it is still a centralized architecture. The CA is the only entity to submit a certificate and update the logs. Namecoin [10] was designed as a decentralized DNS registry. Like Bitcoin, Namecoin is also a cryptocurrency and uses NC payment to register a .bit domain name, which means it will not be censored by the government. Although this is a decentralized PKI application, this application is used for domain name registration, not for IoT environments. Namecoin still has the problem of domain name snatching. Consider that if the system is applied to an IoT network, any device can be easily added as someone else, and even if it is revoked due to malicious behavior, the attacker can easily change the identity of an IoT device and rejoin it into the IoT network.

Yakubov et al. [17] proposed a blockchain-based PKI management framework. Their research introduced which entities should be required for blockchain-based and the functions that should have. However, it did not propose a complete protocol and interactions between entities. Kfoury and Khoury [11] proposed a generic distributed PKI platform based on the Ethereum Blockchain. The Blockchain technology was adopted to replace CAs for supporting PKI that stores the public keys of all devices and introduced a PSK (Pre-Shared Keys) based security protocol for PSK distribution to ensure the security of communication channels. HÃfÃlglund et al. [7] developed an integrated and automated lightweight certificate enrollment protocol for the resource-constrained IoT devices through format conversion between traditional

X.509 digital certificates and a lightweight certificate profile. In the scheme proposed by Belattaf et al. [1], a reliable distributed algorithm was designed, the public key certificate management resources were optimized. The network was organized into various disjoint communities. In each community, the public key certificate management policy was enforced by its own policy, and the delegate object was responsible for sending external authentication requests.

Moreover, Won et al. [16] proposed a decentralized PKI for IoT. In this scheme, the device can only be added through the DM and does not recognize the DM. Therefore, a reliable protocol is required to help IoT devices authenticate the identity of the DM. However, it only allows DM to register IoT devices. A similar study of Certcoin [4] and the scheme [8] has the same disadvantages. In these schemes, it just needs to issue transactions on blockchain for verifying the correctness of issued certificate. Due to lack of authenticity, IoT devices may be added to an attacker's blockchain system, or a malicious device may be added to the IoT network. Kakei et al. [9] proposed a distributed authentication infrastructure that distributes CAs among multiple service providers and performs the standardized cross-certification process to establish the trust relationship. It was built based on smart contracts, which evaluates the trustability of CAs and issues certificates. It can protect the system against CA takeover and spoofing by outsider adversaries. However, it assumes there are no dishonest parties and collusion attacks in the system.

4 System Overview

4.1 System Model

As shown in Fig. 1, the system model contains five roles, (1) new joining device (NJ-Dev) is an IoT device to join the IoT network. (2) authentication device (Auth-Dev) is responsible for introducing unauthenticated external IoT devices. It can be any type of computing device of high computing power and is fully trusted by the IoT network; (3) AAA server (AAA-S) operates outside the IoT network to assist the mutual authentication between IoT devices and Auth-Dev. Between AAA-S and Auth-Devs, they have established trust by sharing common secret keys for the use of authentication; (4) IoT network is composed of Auth-Dev and certain authenticated devices with legal certificates issued by the IoT network. The devices in the IoT network can decide whether to compute signature fragments to endorse the joining of a new device according to computing power and registration policy; (5) Blockchain is responsible for storing certificates and signature fragments for the decentralized setting of IoT networks regarding trust and provenance of transactions in IoT networks.

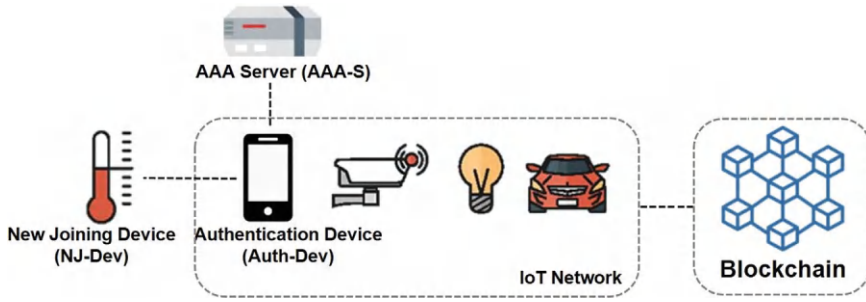


Fig. 1 The system model of the proposed scheme

4.2 Threat Model

This section introduces the security threats by the potential attackers to the current CA-based PKI (X.509 standard) [6]. Several related works analyze potential threats in various aspects. Those security threats are taken into account for the security design of the proposed FiT-DPKI as follows: (1) single point of failure is considered for the traditional PKI as it is a centralized architecture, where CA might be hacked or infected and result in unauthorized distribution of certificates; (2) non-transparency is taken into account for monitoring the entire procedure of issuing certificates regardless of CA's status even though the CA being hacked or exposed; (3) minimizing dependency on CA ensures the issuing of certificates can be completed even if CA is unavailable during the period for the robustness of decentralized PKI design.

There are two problems in the current proposed blockchain-based PKI [5, 9, 20]. Firstly, the lack-of authentication issue. The current blockchain-based PKI system is not authenticated during registration. We need an authentication protocol to ensure that the NJ-Dev is correct. The NJ-Dev should also ensure that it is communicating with the correct entity, but not the attacker. Additionally, the certificate-generation issue. For the current proposed PKI based on blockchain, the method for registering a new member requires the member to provide the public key and use the corresponding private key to generate a signature to prove the ownership of the public key.

4.3 Security Requirements

For the above traditional CA-based PKI and blockchain-based PKI, our research addresses these problems and proposes the corresponding solution. Our research focuses on small or medium-sized IoT networks, such as factories or hospitals. These organizations do not necessarily connect with external CAs for authentication. Instead, they establish an IoT network on the internal network, and the IoT network entities jointly maintain this PKI architecture. Addressing authentication,

our goal is to enable mutual authentication between NJ-Dev, Auth-Dev, and the IoT network. Ensuring unforgeability, we aim to safeguard against unauthorized forging of certificates issued within the IoT network, whether from legitimate IoT devices or external threats. Integrity maintenance involves securing certificate information and transaction records from unauthorized alterations. In a decentralized PKI, our focus is on ensuring consistency among storage nodes, ensuring most nodes retain identical transaction records for certificate management. To prevent single points of failure, our decentralized PKI design strives for operational resilience, enabling PKI tasks to proceed even in the event of IoT device failures. Lastly, our emphasis on auditability involves rigorous inspection of each certificate issuance to verify their legitimacy within IoT networks.

5 Proposed Scheme

The notations of the proposed scheme are as follows, ID is device identification, r and $\tilde{r}$ are random values, K is symmetric key shared with AAA-S, CK is symmetric key for a NJ-Dev and Auth-Devs, $H_K()$ is the function of message authentication code with K , $H()$ is One-way hash function, MAC and MAC' are message authentication code, (PK, SK) is public and private key pair generated by every device itself, SK' is the key used by each device to compute the signature fragment, $Enc_K()$ is the encryption function with symmetric key K , $Dec_K()$ is the decryption function with symmetric key K , $sig(SK, m_1, m_2...)$ is the signing function on m_i with SK , $SFGen()$ is the function to generate the signature fragments, $SReCon()$ is the function to assemble signature fragments, PK_{TSS} is public key for threshold signature, VP_{TSS} is verification parameter for threshold signature, $Verify()$ is the function to verify digital signatures, $Verify_{TSS}()$ is the function to verify threshold signatures.

5.1 Service Setup

The five steps of setup are shown below: (1) The administrator sets up the threshold value (t) according to the security level in an IoT network; (2) When there are enough devices, the administrator executes $DKGen$ with the devices to compute the secret keys, public key PK_{TSS} , and verification parameter VP_{TSS} for the threshold signatures of each device and sets them on the devices; (3) Each device generates its public and private key pair (PK, SK) ; (4) The administrator establishes the blockchain, computes the certificate of each device, and uploads them to the blockchain and all the devices; (5) The administrator uploads PK_{TSS} and VP_{TSS} to the blockchain.

Once the service setup is completed, the security keys shared among the AAA server, Authentication device, newly joined device, and existing IoT devices, along with their issued keys, can be outlined as follows. Stored within the devices are several security keys, including a public-private key pair (PK, SK) , a symmetric key

shared by the AAA server (K), a secret key for threshold signature (SK'), a public key for threshold signature (PK_{TSS}), verification parameters for threshold signature (VP_{TSS}), and certificates ($Cert$). Additionally, within the devices operating on the blockchain, there are certificates for all IoT network devices, a public key for threshold signature (PK_{TSS}), and verification parameters for threshold signature (VP_{TSS}).

5.2 Registration

Three Party Authentication The first step of registering a NJ-Dev requires three-party authentication as followed: (1) A new joining device NJ-Dev_{*i*} sends ID_i and randomly selected r_i to Auth-Dev_{*j*}; (2) Auth-Dev_{*j*} sends $\{ID_i, ID_j, r_i\}$ to AAA-S; (3) AAA-S generates a random values $\tilde{r}$, and computes $MAC_i = H_{K_i}(ID_j \| r_i)$ and $MAC_{ij} = H_{K_A}(r_i \| \tilde{r} \| MAC_i)$, then sends $\tilde{r}$, MAC_i and MAC_{ij} to Auth-Dev_{*j*}; (4) Upon received from AAA-S, Auth-Dev_{*j*} computes MAC_{ij} with K_A and compare with the MAC_{ij} sent by AAA-S. If the verification is failed, it aborts the procedure. Otherwise, it sends $\tilde{r}$, ID_j and MAC_i to NJ-Dev_{*i*}; (5) Upon received from Auth-Dev_{*j*}, NJ-Dev_{*i*} computes MAC_i with K_i and compare it with MAC_i sent by Auth-Dev_{*j*}. If the verification is failed, it aborts the procedure. Otherwise, it computes $MAC'_i = H_{K_i}(\tilde{r})$ and sends MAC'_i to Auth-Dev_{*j*}; (6) Auth-Dev_{*j*} computes $MAC'_j = H_{K_A}(\tilde{r} \| MAC'_i)$ and sends MAC'_i and MAC'_j to AAA-S; (7) AAA-S computes MAC'_i and MAC'_j , and then compares if $MAC'_i = MAC'_j$ from the Auth-Dev_{*j*}. If so, it computes $CK = H_{K_i}(\tilde{r} \| r_i)$ and performs symmetric encryption with keys H_{K_A} and K_i to generate c_j and c_i , where $c_j = Enc_{K_A}(CK \| MAC'_j)$ and $c_i = Enc_{K_i}(CK \| MAC'_i)$; (8) Auth-Dev_{*j*} decrypts c_j to obtain CK and uses CK to perform symmetric encryption $c = Enc_{CK}(PK_j)$, and then sends c_i and c to NJ-Dev_{*i*}; (9) NJ-Dev_{*i*} uses K_i to decrypt c_i to obtain CK and decrypts c with CK to obtain PK_j of the Auth-Dev_{*j*}.

The Detailed Registration After the three-party authentication, NJ-Dev_{*i*} and Auth-Dev_{*j*} confirm that each other is real, correct, and legal. The detailed interactions between the three entities are as followed: (1) NJ-Dev_{*i*} uses its own private key to sign σ_i . The content of this signature is its own ID_i , its own public key PK_i , Auth-Dev_{*j*}'s ID_j and Auth-Dev_{*j*}'s public key PK_j . The symmetric key CK is used to encrypt the public key PK_i and the signature σ_i to the Auth-Dev_{*j*} as $C_i = Enc_{CK}(\sigma_i \| PK_i)$; (2) The Auth-Dev_{*j*} decrypts C_i , obtains the public key PK_i of NJ-Dev_{*i*} and generates $m = \{ID_i \| PK_i \| ID_j \| PK_j \| \sigma_i\}$ and signature $\sigma_j = sig(SK_j, m)$, and finally computes $Q_{register} = \{m \| \sigma_j\}$. In general, $Q_{register}$ contains the information for NJ-Dev_{*i*} to register and device j helped it to perform registration authentication. This information can help to trace back to the $Q_{register}$ and find out who brought the device into the IoT network if the device become illegal due to the violation of systems rules; (3) After receiving $Q_{register}$, a device with reasonable computing power in the IoT network will perform a four-step verification. It verifies if ID_i has been registered and the authenticity of Auth-Dev_{*j*} identity.

The device then executes $Verify(PK_j, m, \sigma_j)$ and $Verify(PK_i, m', \sigma_i)$ with the public key recorded in $Q_{register}$ to verify the corresponding signature. If all verifications are passed, it computes $\tilde{\sigma}'_k = SFGen(H(Q_{register}), SK_k, PK_{TSS}, VP_{TSS})$ and $SF'_k = \{ID_i \| ID_k \| \tilde{\sigma}'_k \| sig(SF_k, \tilde{\sigma}'_k)\}$, and uploads SF'_k to the blockchain. As shown in Algorithm 1, the smart contract will be executed by the blockchain. (4) $Auth-Dev_j$ queries the blockchain to collect SF'_k regularly, and check if it is generated for ID_i . $Auth-Dev$ verifies ID_k identity and extract ID_k 's public key PK_k from $Cert_k$, and uses PK_k for executing $Verify(PK_k, \tilde{\sigma}'_k, sig(SK_k, \tilde{\sigma}'_k))$. When collecting σ' over t , $Auth-Dev_j$ generates the threshold signature σ_i and verifies signature $Verify_{TSS}(PK_{TSS}, H(Q_{register}), \tilde{\sigma}_i)$. After computing a certification, it computes $Cert_i = \{Q_{register} \| \tilde{\sigma}_i\}$, and send $Cert_i$ with random value r to $NJ-Dev_i$; (5) If $NJ-Dev_i$ received $Cert_i$, the signature $\sigma_{accept} = (SK_i, r)$ is returned to $Auth-Dev_j$; (6) $Auth-Dev_j$ verifies the signature σ_{accept} . If it passes the verification, which means $NJ-Dev_i$ has received the certificate correctly, $Auth-Dev_j$ uploads this certificate to the blockchain. As shown in Algorithm 2, the smart contract will be executed by the blockchain.

Algorithm 1 is for generating the signature fragment of the smart contract: It takes as input a signature fragment SF'_j and outputs “accept” or “reject” on the transaction. The detailed steps of this algorithm are shown as followed: (1) Query ID_k public key PK_k ; (2) $Verify(PK_k, \tilde{\sigma}_k, sig(SK_k, \tilde{\sigma}'_k))$. If valid, then accept the transaction and build SF'_j signature fragment structure. If not, then reject the transaction.

Algorithm 2 is for registration of the smart contract: It takes as input (“register”, $Cert_i$) and outputs “accept” or “reject” on the transaction. The detailed steps of this algorithm are shown as followed: (1) Query ID_i that whether device i has been registered. If it already exists, then reject the transaction. If not, then go to step2; (2) Query ID_j that whether device j is an $Auth-Dev$. If it is, then go to step3. If not, then reject the transaction; (3) Query PK_{TSS} ; (4) $Verify_{TSS}(PK_{TSS}, H(Q_j), \tilde{\sigma}_j)$; If valid, then accept the transaction and build device i . If not, then reject the transaction.

Algorithm 3 is for query of the smart contract: It takes as input (“Query”, $Cert_i$) and outputs $NJ-Dev_i$'s structure or “not exist”. The detailed steps of this algorithm are shown as followed: Query ID_i that whether device i exists. If it exists, then return $NJ-Dev_i$'s structure. If not, then return “not exist”.

5.3 Query Whether ID Is Registered

As shown in Algorithm 3, a transaction (“Query”, $Cert_i$) is received for query device i . The device on the blockchain executes function $Query(ID_i)$ according to the smart contract. The detailed steps are shown as followed: (1) The device i sends a transaction (“Query”, ID_j) to the blockchain; (2) The ID_j has been registered if the ID_j certificate can be found on the blockchain. Otherwise, it is not registered.

5.4 Identity Verification

The detailed steps are shown as followed:

- (1) NJ- Dev_i submits a transaction (“ $Query$ ”, ID_j) to the blockchain.
- (2) Blockchain queries $Cert_j$ and sends it back to NJ- Dev_i .
- (3) NJ- Dev_i submits a transaction (“ $Query$ ”, PK_{TSS}) to the blockchain for verifying the threshold signature.
- (4) If PK_{TSS} is found on the blockchain, it is returned to NJ- Dev_i .
- (5) If NJ- Dev_i gets PK_{TSS} , it submits a transaction $Verify_{TSS}(PK_{TSS}, H(Q_j), \tilde{\sigma}_j)$ for verifying the ID_j ’s certificate.

6 Security Analysis

The security analysis of the proposed FiT-DPKI demonstrates the security features, authentication, unforgeability, integrity, consistency, prevention of single points of failures, and auditability. Due to the page limitation, this work provide the complete security analysis in the appendix.

7 Evaluation

The simulating environment contains two Raspberry Pi and a personal computer. The personal computer is responsible for simulating the Auth-Dev and control two Raspberry Pis with putty. The Raspberry Pis are for simulating the NJ-Dev, and the AAA-S and IoT network, respectively. The parameters are including data size, the number of members and threshold. The following is the computer specification: Windows 10 operating system, Intel Core i5-8500 CPU 3.00GHz, 16GB RAM.

7.1 Performance of Threshold Signature

- We ignore the parameters of the number of members and data size since it will not affect the performance based on equation of DKGen. As shown in Fig. 2a, the yellow line represents the SK_i that generates the user for threshold signature, the orange line represents the environmental parameters of the dealer generating the threshold signature, and the blue line represents the total time of DKGen.
- As shown in Fig. 2b, we can observe the change of data size that affects the computation cost of SFGGen.
- As shown in Fig. 2c, the number of signature fragments is fixed at 5, which changes the data size. As shown in Fig. 2d, the fixed data size is 4096 bits, which changes

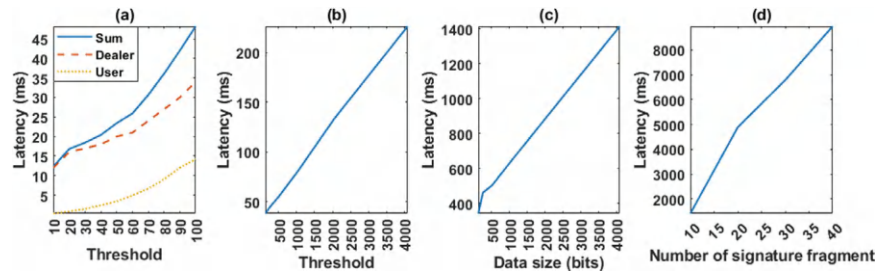


Fig. 2 **a** Computation costs of DKGen **b** Computation costs of SFGGen **c** Computation costs of SReCon on data size **d** Computation costs of SReCon on number of signature fragment

the number of signature fragments. We can see that the impact of data size and the number of signature fragments on latency is linear growth. Moreover, it can also be known that the impact of signature fragments on latency is more significant than the data size.

7.2 Performance of Three Party Authentication

Table 1 shows the simulation results of register phase, and the device specifications are shown in Table 2. We simulate the environment of 33 IoT devices. The NJ-Dev and Auth-Dev are individually simulated using the Raspberry Pi and personal computer, and the remaining 31 devices are simulated using another Raspberry Pi. We maximized the burden of the Auth-Dev, which must combine all 32 signatures. In the real environment, it only needs to exceed the threshold value to combine a full signature, and a more robust device will be used as the Auth-Dev. The size of our data for threshold signature is about 400 bits.

Table 1 Simulation results of register phase

	New joining device	Authentication device	IoT network
Product	Raspberry Pi 3 Model B+	Personal computer	Raspberry Pi 3 model B+
Number of nodes	1	1	31
Latency	1702 ms	567 ms	340 ms
Data size	426 bits		

Table 2 Simulation results of three-party authentication

	New joining device	Authentication device	AAA server
Product	Raspberry Pi 3 Model B+	Personal Compute	Raspberry Pi 3 Model B+
OS	Ubuntu 18.04.4 LTS	Windows 7	Ubuntu 18.04.4 LTS
CPU	ARM Cortex-A53(1.4GHz)	Intel Core i5-4460 (3.20GHz)	ARM Cortex-A53 (1.4GHz)
Memory	1 GB	8 GB	1 GB
Latency	785 ms	382 ms	826 ms
Algorithm	Signature algorithm: RSA signature-1024 bits (java.security) Symmetric encryption and decryption: AES (javax.crypto) HMAC: HmacSHA256 (javax.crypto)		

7.3 Comparison of Security Properties

Comparison of security properties, as shown in Table 3. The scheme [8] is based on the scheme of [4] , so they have similar security properties. For the schemes [4, 8, 13, 16], there is no authentication mechanism for the registered device in the IoT network to ensure that the device communicated with during registration is the correct one. Regarding auditability, each device has the ability to engage the generation of the certificate, and the certificate format can track which devices agreed on generating this certificate. For the schemes [1, 7, 9], certificates are not public in order to assure the owner privacy, which may result in other devices unable to detect fake certificates in the system. Moreover, the single point of failure might happen in the schemes [1, 7]. The former is because the authentication requests in each community which have to be sent to external are all needed to delegate to a single trust object; The latter is because the traditional centralized CA-based PKI is adopted to authenticate the IoT devices.

Table 3 Comparison of security properties

	Authent-ication	Unforg-eability	Integ-rity	Consis-tency	Reduce Single point of Failure	Audita-bility
Proposed FiT-DPKI	✓	✓	✓	✓	✓	✓
Fromknecht et al. [4]	×	✓	✓	✓	✓	×
Matsumoto and Reischuk [12]	✓	×	✓	✓	✓	✓
Hoogland [8]	×	✓	✓	✓	✓	×
Won et al. [16]	×	✓	✓	✓	✓	×
Singla and Bertino [13]	×	✓	✓	✓	✓	×
Kfoury and Khoury [11]	✓	×	✓	✓	✓	✓
Kakei et al. [9]	✓	×	✓	✓	✓	✓
Höglund et al. [7]	✓	×	✓	✓	×	×
Belattaf et al. [1]	✓	×	×	✓	×	×

8 Conclusion

This study proposes a new decentralized public-key infrastructure scheme, i.e., FiT-DPKI, based on blockchain, threshold signatures, and three-party authentication, for IoT networks. The proposed FiT-DPKI overcomes single points of failure, non-transparency, and connection dropping to CAs. FiT-DPKI enables the authorization of a newly joining device for issuing a certificate belonging to a specified IoT networks by multiple devices in the network. Moreover, the number of devices and their weight to authorize the issuance of certificates are adjustable thanks to the adoption of threshold signatures. By introducing an authentication device, which is considered normally an IoT gateway and more powerful than the other IoT devices, the authentication of a new device for the registration and issuance of a certificate is more secure due to the use of AAA-based three-party authentication needs not to be required persistently.

To demonstrate the feasibility, this work implements the prototype system and evaluate the performance. Regarding the security, this paper also analyzes formally the security of the proposed scheme. Overall, the proposed FiT-DPKI provides a feasible solution to build up a decentralized PKI to maintain the security and robustness although the devices are lightweight and vulnerable to cyberattacks in IoT networks.

Acknowledgements This work was partially supported by the National Science and Technology Council of Taiwan under Grant 109-2221-E-110-041-MY3, Grant 109-2923-E-011-006-MY3, and Grant 110-2218-E-110-007-MBK, and in part by the Information Security Research Center, National Sun Yat-sen University, Taiwan.

References

1. Belattaf S, Mohammedi M, Omar M, Aoudjit R (2021) Reliable and adaptive distributed public-key management infrastructure for the internet of things. *Wireless Pers Commun* 120(1):113–137
2. Ben Laurie AL, Kasper E (2013) Certificate transparency. rfc6962, internet engineering task force (ietf). <http://tools.ietf.org/html/rfc6962>
3. Boneh D, Franklin M (2001) Identity based encryption from the weil pairing. *Cryptology ePrint Archive*, Report 2001/090
4. Fromknecht C, Velicanu D, Yakubov S (2014) A decentralized public key infrastructure with identity retention. *IACR Cryptol. ePrint Arch.* 2014:803
5. Gennaro R, Halevi S, Krawczyk H, Rabin T (2008) Threshold RSA for dynamic and Ad-Hoc Groups. In: Smart N (ed) *Advances in cryptology - EUROCRYPT 2008*. Springer, Berlin, pp 88–107
6. Gerck E (2000) Overview of certification systems: X.509, PKIX, CA, PGP and SKIP. www.thebell.net/papers/certover.pdf
7. Höglund J, Lindemer S, Furuheid M, Raza S (2020) Pki4iot: towards public key infrastructure for the internet of things. *Comput Secur* 89:101658
8. Hoogland M (2018) A distributed public key infrastructure for the iot
9. Kakei S, Shiraishi Y, Mohri M, Nakamura T, Hashimoto M, Saito S (2020) Cross-certification towards distributed authentication infrastructure: a case of hyperledger fabric. *IEEE Access* 8:135742–135757

10. Kalodner HA, Carlsten M, Ellenbogen P, Bonneau J, Narayanan A (2015) An empirical study of namecoin and lessons for decentralized namespace design. In: WEIS. Citeseer
11. Kfoury E, Khoury D (2018) Distributed public key infrastructure and psk exchange based on blockchain technology. In: 2018 IEEE international conference on internet of things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCoM) and IEEE smart data (SmartData). IEEE (2018), pp 1116–1120
12. Matsumoto S, Reischuk RM (2016) Ikp: Turning a pki around with blockchains. IACR Cryptol. ePrint Arch. 2016:1018
13. Singla A, Bertino E (2018) Blockchain-based PKI solutions for IoT. In: 2018 IEEE 4th international conference on collaboration and internet computing (CIC), pp 9–15
14. Wang Y, Wong DS, Wu Q, Chow SSM, Qin B, Liu J (2014) Practical distributed signatures in the standard model. In: Benaloh J (ed) Topics in cryptology - CT-RSA 2014. Springer International Publishing, Cham, pp 307–326
15. Wolff J (2016) How the 2011 hack of DigiNotar changed the internet's infrastructure. <https://slate.com/technology/2016/12/how-the-2011-hack-of-diginotar-changed-the-internets-infrastructure.html>
16. Won J, Singla A, Bertino E, Bollella G (2018) Decentralized public key infrastructure for internet-of-things. In: MILCOM 2018 - 2018 IEEE military communications conference (MILCOM), pp. 907–913. <https://doi.org/10.1109/MILCOM.2018.8599710>
17. Yakubov A, Shbair W, Wallbom A, Sanda D, State R (2018) A blockchain-based PKI management framework. <https://doi.org/10.1109/NOMS.2018.8406325>

Revolutionizing Healthcare: Case Studies of AI Algorithms Transforming the Field of Medicine



Aneesh Sharma, Rashee Sharma, and Cheng-Hsing Yang

Abstract In recent years, artificial intelligence (AI) technologies have revolutionized the field of medicine, providing innovative solutions to complex healthcare challenges. This research paper delves into the intersection of AI and medical case studies, exploring the transformative impact of machine learning algorithms, natural language processing, and computer vision techniques on diagnosing, predicting, and treating various medical conditions. Through an exhaustive analysis of diverse medical cases, this study showcases the efficacy of AI-driven tools in enhancing the accuracy and efficiency of clinical decision-making processes. Drawing on a wide array of real-world medical scenarios, this research paper not only demonstrates the potential of AI to revolutionize healthcare but also emphasizes the critical need for multidisciplinary collaboration between AI researchers, healthcare professionals, and policymakers. By harnessing the power of AI in analyzing medical case studies, this study paves the way for a future where precision medicine and personalized healthcare become accessible and efficient, ultimately improving patient outcomes and reshaping the landscape of modern medicine.

Keywords Artificial intelligence · Medical case studies · Precision medicine · Machine learning algorithms · Healthcare innovation · Clinical decision-making · Disease diagnosis · Predictive analytics · Personalized treatment

A. Sharma (✉)
University of California, Berkeley, USA
e-mail: mail@asharma.info

R. Sharma
Santokba Durlabhji Memorial Hospital, Jaipur, India

C.-H. Yang
National Pingtung University, Pingtung, Taiwan

1 Introduction

The landscape of healthcare is undergoing a profound transformation, catalyzed by the integration of Artificial Intelligence (AI) algorithms into medical practice. As technology continues to advance at an unprecedented pace, the intersection of AI and healthcare has emerged as a focal point of innovation, promising revolutionary changes in disease diagnosis, treatment planning, and patient care. The paper, titled “Revolutionizing Healthcare: Case Studies of AI Algorithms Transforming the Field of Medicine,” delves into this transformative paradigm shift, exploring a series of compelling case studies that vividly illustrate the tangible impact of AI in reshaping the healthcare industry.

With the relentless growth of medical data and the complexities of modern diseases, the need for intelligent systems that can swiftly process vast amounts of information and extract meaningful insights has never been more critical. AI algorithms, particularly those utilizing deep learning techniques, have exhibited remarkable capabilities in handling complex medical datasets. By learning patterns, discerning correlations, and predicting outcomes, these algorithms are not just tools; they are becoming indispensable partners for healthcare professionals, enhancing their decision-making processes and, ultimately, improving patient outcomes.

This paper embarks on a journey through select case studies, each representing a milestone in the application of AI in diverse medical domains. From oncology to radiology, from genomics to personalized medicine, these cases not only demonstrate the efficacy of AI algorithms but also showcase their adaptability across a spectrum of medical specialties. Through an in-depth exploration of these real-world examples, we aim to shed light on the ways in which AI is reshaping medical practices, unraveling new avenues for research, and fundamentally redefining the patient experience.

In the following sections, we delve into these case studies, examining the methodologies employed, the challenges encountered, and the transformative outcomes achieved. By doing so, we illuminate the potential of AI algorithms to revolutionize healthcare, offering a glimpse into a future where technology and human expertise harmoniously converge to provide unparalleled healthcare solutions.

2 Overview

The rapid integration of Artificial Intelligence (AI) into the field of medicine has ushered in a new era of healthcare innovation, transforming the way diseases are diagnosed, treatments are planned, and patient outcomes are optimized. This section provides an overview of the revolutionary impact of AI algorithms in healthcare, supported by a selection of sample papers that exemplify the diverse applications and transformative potential of this technology.

2.1 AI Algorithms for Detecting Eye Diseases

In recent years, AI-powered systems, such as DeepMind, have demonstrated exceptional proficiency in detecting eye diseases [1–3] like diabetic retinopathy. By analyzing retinal images, these algorithms exhibit levels of accuracy comparable to skilled ophthalmologists, enabling early diagnosis and intervention, thus preventing vision loss. This paper showcases how AI is revolutionizing ophthalmic care, ensuring timely interventions and improving the quality of life for patients with diabetic eye conditions.

The outcomes of employing DeepMind for detecting eye diseases have been nothing short of transformative. Studies have demonstrated that the AI algorithms can accurately identify subtle changes in retinal structures that might escape the human eye. DeepMind's diagnostic capabilities rival, and in some cases surpass, those of experienced ophthalmologists. The technology facilitates timely intervention, enabling healthcare professionals to initiate appropriate treatments and preventative measures, thereby significantly reducing the risk of vision loss in patients with conditions such as diabetic retinopathy.

The utilization of DeepMind for detecting eye diseases represents a watershed moment in the realm of ophthalmic diagnostics. Its ability to augment the expertise of healthcare professionals and enhance the accuracy of early diagnoses not only transforms the lives of individuals at risk of vision loss but also sets a precedent for the application of AI in other medical disciplines. As the technology continues to evolve, the partnership between human ingenuity and artificial intelligence holds the promise of reshaping the landscape of healthcare, making it more precise, accessible, and patient-centric than ever before.

2.2 Genomic Sequencing and AI in Cancer Research

Genomic sequencing, coupled with AI algorithms, has revolutionized cancer research and treatment strategies [4, 5]. By analyzing vast genomic datasets, AI can identify genetic mutations associated with various cancers, paving the way for personalized therapies. This paper delves into how AI-driven genomics accelerates the development of targeted cancer treatments, offering new hope for patients and significantly improving overall survival rates.

Cancer, a complex and diverse group of diseases, continues to challenge the medical community with its intricacies. In the quest for more effective treatments, the marriage of Genomic Sequencing and Artificial Intelligence (AI) has emerged as a transformative force in cancer research. Genomic sequencing, the process of analyzing an individual's DNA, and AI, with its advanced algorithms and machine learning capabilities, combine forces to decipher the genetic basis of cancer. This paper delves into the synergy of Genomic Sequencing and AI in cancer research, exploring

how these technologies are revolutionizing our understanding of cancer biology and paving the way for highly personalized and targeted therapeutic interventions.

Genomic Sequencing Genomic sequencing, particularly next-generation sequencing (NGS), has revolutionized our ability to decode the genetic alterations underlying cancer. By unraveling the unique genetic mutations, copy number variations, and alterations in gene expression patterns, scientists gain insights into the molecular mechanisms driving cancer progression. This information is invaluable, as it allows researchers to identify potential therapeutic targets and predict a patient's response to specific treatments. Genomic sequencing also aids in classifying cancers into subtypes, enabling tailored therapies that address the specific genetic characteristics of each patient's tumor.

AI in Genomic Analysis The sheer volume and complexity of genomic data necessitate advanced computational techniques for analysis. AI algorithms, especially deep learning models, excel at detecting subtle patterns within <https://www.overleaf.com/project/651ea7958a3a3fad379ad537vast> datasets. In cancer research, AI plays a pivotal role in identifying meaningful associations between genetic alterations and clinical outcomes. These algorithms can predict disease trajectories, identify novel biomarkers, and even suggest personalized treatment strategies based on an individual patient's genomic profile. By continually learning from new data, AI ensures that cancer researchers stay at the forefront of knowledge, enabling more informed decisions in the quest for effective therapies.

Precision Medicine and Targeted Therapies The integration of Genomic Sequencing and AI has propelled the concept of precision medicine, where treatments are tailored to the genetic makeup of each patient's cancer. Targeted therapies, designed to specifically address the molecular vulnerabilities identified through genomic analysis, offer a more nuanced and effective approach than traditional chemotherapy. By targeting the specific genetic alterations driving tumor growth, these therapies minimize damage to healthy cells, enhance treatment efficacy, and reduce side effects, significantly improving the quality of life for cancer patients.

The fusion of Genomic Sequencing and AI in cancer research marks a paradigm shift in our approach to understanding and treating cancer. By deciphering the intricate genetic landscapes of tumors and harnessing the computational power of AI, researchers are ushering in an era of highly personalized cancer therapies. This transformative synergy not only expands our knowledge of cancer biology but also translates into tangible benefits for patients, offering hope for more effective treatments, improved outcomes, and ultimately, a brighter future in the fight against cancer.

2.3 Machine Learning in Drug Discovery

Drug discovery, a complex and time-consuming process, lies at the heart of medical innovation, offering the promise of novel treatments and potential cures for a multitude of diseases. In recent years, the integration of Machine Learning (ML) techniques has revolutionized drug discovery [6–10], transforming the way researchers

identify potential drug candidates, predict their interactions with biological targets, and optimize their efficacy. This paper explores the pivotal role of machine learning in drug discovery, showcasing how advanced computational methods are reshaping the pharmaceutical landscape and expediting the development of innovative and targeted therapies.

Traditional Challenges in Drug Discovery Historically, drug discovery has been plagued by challenges such as the vast chemical space to explore, high costs, and lengthy timelines. Identifying molecules with the desired therapeutic properties among billions of possibilities is a daunting task. Additionally, understanding the complex interactions between these molecules and biological targets is intricate and time-intensive. Machine learning, however, offers a powerful solution by enabling the analysis of enormous datasets and the extraction of valuable insights from this wealth of information.

ML Algorithms in Drug Discovery Machine learning algorithms, ranging from classical methods like Random Forests and Support Vector Machines to more advanced techniques like deep learning neural networks, have been harnessed to analyze diverse biological and chemical data. These algorithms can predict the biological activity of compounds, model the three-dimensional structure of proteins, and optimize the chemical properties of molecules to enhance their drug-like characteristics. By learning from vast datasets of known drug compounds and their interactions, ML algorithms identify patterns and correlations, aiding in the discovery of potential drug candidates and accelerating the lead optimization process.

Accelerating Drug Development One of the significant contributions of machine learning in drug discovery is its ability to significantly reduce the time and resources required for the development of new drugs. By predicting the potential of molecules early in the process, researchers can focus their efforts on the most promising candidates, streamlining preclinical and clinical trials. ML algorithms also play a vital role in repurposing existing drugs for new indications, uncovering novel therapeutic uses for established pharmaceuticals, and expediting their path to market.

Personalized Medicine and Tailored Treatments Beyond traditional drug discovery, machine learning facilitates the realization of personalized medicine. By analyzing genetic and molecular data from individual patients, ML algorithms identify biomarkers, predict disease susceptibility, and guide the selection of optimal therapies. This tailored approach ensures that treatments are not only effective but also tailored to the specific genetic makeup and biological characteristics of each patient, maximizing therapeutic outcomes and minimizing adverse effects.

The integration of machine learning in drug discovery represents a transformative leap forward in the pharmaceutical industry. By leveraging advanced computational techniques, researchers can navigate the complexities of biological systems and chemical interactions with unprecedented efficiency and accuracy. This synergy of human ingenuity and artificial intelligence not only accelerates the drug discovery process but also holds the potential to revolutionize medicine, ushering in an era where treatments are not only innovative and targeted but also personalized to the unique biological profiles of individual patients. As machine learning continues to evolve, the future of drug discovery shines brighter, promising a world where the most

challenging diseases are met with the most sophisticated and effective therapeutic solutions.

2.4 Robotic Surgery and AI-Assisted Procedures

AI-enabled robotic surgery systems [11–14] have redefined the landscape of minimally invasive procedures. By integrating machine learning algorithms, these systems enhance the precision and dexterity of surgeons, leading to faster recovery times and reduced complications. This paper investigates the impact of AI-assisted robotic surgeries across various specialties, illustrating how these technologies are reshaping surgical practices and ensuring better patient outcomes. In the realm of modern medicine, the convergence of Robotics and Artificial Intelligence (AI) has led to a paradigm shift in surgical practices. Robotic Surgery, coupled with AI-assisted procedures, represents the pinnacle of technological advancement in the field. These innovations have revolutionized the way surgeries are performed, enhancing the precision, dexterity, and efficiency of surgeons. This paper explores the transformative impact of Robotic Surgery and AI-assisted procedures, shedding light on how these technologies are redefining surgical standards, improving patient outcomes, and paving the way for a future where complex procedures are safer, more accessible, and highly personalized.

The Evolution of Robotic Surgery Robotic Surgery, with its origins in the early 2000s, has rapidly evolved from a novel concept to a widely adopted surgical tool. Utilizing robotic arms controlled by surgeons via a console, these systems offer unparalleled precision and stability during delicate procedures. In recent years, the integration of Artificial Intelligence has further augmented these robotic platforms, enabling them to analyze vast amounts of patient data, predict surgical outcomes, and assist surgeons in real-time decision-making. This collaboration between human skill and machine intelligence has elevated the capabilities of surgical teams, allowing them to perform intricate procedures with enhanced accuracy.

AI-Assisted Decision Making Artificial Intelligence algorithms, particularly machine learning models, analyze preoperative and intraoperative data to provide surgeons with valuable insights. By processing patient information, such as medical history, imaging scans, and real-time physiological data, AI assists in surgical planning and execution. These algorithms can identify optimal incision points, assess tissue characteristics, and even predict potential complications during surgery. With AI-driven decision support, surgeons can make more informed choices, ensuring a higher degree of safety and efficacy throughout the surgical process.

Enhancing Surgical Precision Robotic Surgery, in conjunction with AI, enhances surgical precision to microscopic levels. The robotic arms, guided by AI algorithms, can perform intricate maneuvers with unmatched steadiness, reducing the margin of error significantly. This level of precision is particularly valuable in delicate surgeries, such as neurosurgery and microsurgery, where even the slightest miscalculation can have profound consequences. The ability to execute precise movements, aided by AI's

real-time feedback, ensures that surgeries are not only successful but also minimally invasive, leading to faster recovery times and reduced post-operative complications.

Personalized and Accessible Healthcare The integration of AI and robotic systems is democratizing healthcare by enabling personalized treatment plans tailored to individual patient profiles. AI analyzes vast datasets, considering genetic factors, lifestyle choices, and medical history, to recommend the most suitable surgical approach and post-operative care. Moreover, these technologies are transcending geographical barriers, allowing expert surgeons to remotely assist in complex procedures, bringing specialized care to underserved regions and ensuring that patients worldwide have access to the best surgical expertise.

Robotic Surgery and AI-assisted procedures stand at the forefront of a medical revolution. By combining the precision of robotics with the analytical power of AI, these technologies are redefining surgical excellence. Surgeons are empowered to perform procedures with unprecedented accuracy, ensuring optimal patient outcomes. As research and development continue to advance, the future holds the promise of even more sophisticated and innovative applications, making surgeries safer, more personalized, and accessible to all, ultimately reshaping the landscape of healthcare and improving lives across the globe.

3 Conclusion

The integration of Artificial Intelligence (AI) across diverse fields, from healthcare to drug discovery and surgical practices, marks a monumental leap in human innovation. Through the synergy of human expertise and computational intelligence, these technological advancements are revolutionizing industries, enhancing outcomes, and paving the way for a future where the boundaries of what is possible in science and medicine continue to expand.

In the realm of healthcare, AI-driven diagnostics are transforming how diseases are identified and managed. Whether it's detecting eye diseases, deciphering genetic mutations in cancer research, or predicting skin conditions with remarkable accuracy, AI is augmenting the capabilities of medical professionals. The precision and efficiency brought by AI not only lead to earlier diagnoses but also open avenues for personalized treatments, ensuring that patients receive tailored care based on their unique genetic profiles and health histories.

In the domain of drug discovery, the marriage of Machine Learning and genomics is reshaping the pharmaceutical landscape. AI's ability to analyze vast datasets accelerates the identification of potential drug candidates, reducing the time and resources traditionally associated with drug development. Moreover, these technologies enable the concept of precision medicine, where treatments are not merely generalized but meticulously designed to match the specific genetic characteristics of individual patients. This personalized approach holds the promise of minimizing adverse effects and maximizing therapeutic outcomes.

In surgical practices, the fusion of Robotics and AI is revolutionizing the operating room. Robotic Surgery, guided by AI algorithms, ensures unparalleled precision and stability during complex procedures. Surgeons are equipped with real-time data analysis, enabling them to make informed decisions and execute intricate maneuvers with unmatched accuracy. This advancement not only improves surgical outcomes but also allows for minimally invasive techniques, reducing recovery times and enhancing overall patient satisfaction.

As these technologies continue to evolve, ethical considerations and responsible implementation remain crucial. Ensuring patient privacy, safeguarding against biases in AI algorithms, and promoting equitable access to these innovations are paramount. Moreover, ongoing research and collaboration between professionals in medicine, computer science, and engineering are essential to unlocking the full potential of AI.

In conclusion, the integration of AI into healthcare, drug discovery, and surgical practices heralds a future where the boundaries of medical knowledge are constantly pushed. These advancements not only improve the quality and efficiency of healthcare delivery but also inspire hope for patients worldwide. As we move forward, embracing the potential of AI with ethical integrity and a commitment to equitable access will continue to transform the landscape of medicine, offering new solutions, expanding horizons, and ultimately enhancing the well-being of humanity.

Acknowledgements This paper was partially supported under the grant MOST 111-2221-E153-005 and NSTC 112-2221-E-153-003.

References

1. Ting DS, Pasquale LR, Peng L, Campbell JP, Lee AY, Raman R, Tan GS, Schmetterer L, Keane PA, Wong TY (2019) Artificial intelligence and deep learning in ophthalmology. *British J Ophthalmol* 103(2):167–175
2. Cheung CY, Tang F, Ting DS, Tan GS, Wong TY (2019) Artificial intelligence in diabetic eye disease screening. *Asia-Pacific J Ophthalmol* 8(2):158–164
3. Sorrentino FS, Jurman G, De Nadai K, Campa C, Furlanello C, Parmeggiani F (2020) Application of artificial intelligence in targeting retinal diseases. *Current Drug Targets* 21(12):1208–1215
4. Dlamini Z, Francies FZ, Hull R, Marima R (2020) Artificial intelligence (AI) and big data in cancer and precision oncology. *Comput Struct Biotechnol J* 18:2300–2311
5. Hideyuki S, Nakayama Keiichi I (2020) Artificial intelligence in oncology. *Cancer science* 111(5):1452–1460
6. Vamathevan J, Clark D, Czodrowski P, Dunham I, Ferran E, Lee G, Li B et al (2019) Applications of machine learning in drug discovery and development. *Nat Rev Drug Discov* 18(6):463–477
7. Lauv P, Tripti S, Xiuzhen H, Ussery David W, Shanzhi W (2020) Machine learning methods in drug discovery. *Molecules* 25(22):5277
8. Antonio L (2015) Machine-learning approaches in drug discovery: methods and applications. *Drug Discov Today* 20(3):318–331
9. Natalie S, Emily S, Jessica C, Jason R, David R, Nicola J, Jie Z, Leong C, Renzhi C (2019) Survey of machine learning techniques in drug discovery. *Current drug metabolism* 20(3):185–193

10. Gaudelet T, Day B, Jamasb AR, Soman J, Regep C, Liu G, Hayter JB et al (2021) Utilizing graph machine learning within drug discovery and development. *Briefings Bioinf* 22(6):bbab159
11. Bodenstedt S, Wagner M, Müller-Stich BP, Weitz J, Speidel S (2020) Artificial intelligence-assisted surgery: potential and challenges. *Viscer Med* 36(6):450–455
12. Qi Z, Xiao-Guang H, Yun-Feng X, Ming-Xing F, Jing-Wei Z, Ya-Jun L, Da H, Wei T (2020) Robotic navigation during spine surgery. *Expert review of medical devices* 17(1):27–32
13. Xiao-Guang H, Wei T (2019) Artificial intelligence in orthopedic surgery: current state and future perspective. *Chinese Med J* 132(21):2521–2523
14. Ciuti G, Skonieczna-Żydecka K, Marlicz W, Iacovacci V, Liu H, Stoyanov D, Arezzo A et al (2020) Frontiers of robotic colonoscopy: a comprehensive review of robotic colonoscopes and technologies. *J Clin Med* 9(6):1648

Analysis and Detection of Abnormal Transactions on Ethereum



Iuon-Chang Lin and Yu-Chun Chan

Abstract In recent years, blockchain has rapidly evolved, becoming a crucial foundation for cryptocurrencies and digital assets. However, this growth has brought new risks like money laundering and fraud. Ensuring user protection and detecting suspicious accounts is now a vital concern in blockchain development. Therefore, detecting abnormal transactions in blockchain has emerged as an important research field. This study proposes a machine learning-based approach to identify fraudulent activities in phishing accounts. Real data from Ethereum is utilized, specifically selecting transaction data from accounts marked as phishing accounts by Etherscan, as well as other unmarked accounts for comparison. Machine learning techniques are employed to assist in identifying abnormal transaction behaviors, with the ultimate goal of enhancing the security of blockchain transactions. The thesis introduces transaction time and frequency as a characteristic for identifying phishing accounts, and the verification results validate this idea.

Keywords Blockchain · Ethereum · Anomaly detection

1 Introduction

Blockchain, introduced by Satoshi Nakamoto in 2009 [1], ushered in a new era, with Bitcoin as its first successful implementation. Despite the secure and decentralized nature of blockchain, the rise of cryptocurrencies and exchanges has attracted fraudsters. The Chainalysis 2022 Crypto Crime Report [2] indicates an increase in cryptocurrencies received by illicit addresses, with phishing scams accounting for nearly 80% of virtual currency-related incidents. Phishing attacks, typically via emails, have expanded to text messages and social media, aiming to deceive users into revealing personal information or private keys for cryptocurrency theft (Fig. 1).

I.-C. Lin (✉) · Y.-C. Chan

Department of Management Information Systems, National Chung Hsing University, Taichung, Taiwan

e-mail: iclin@nchu.edu.tw

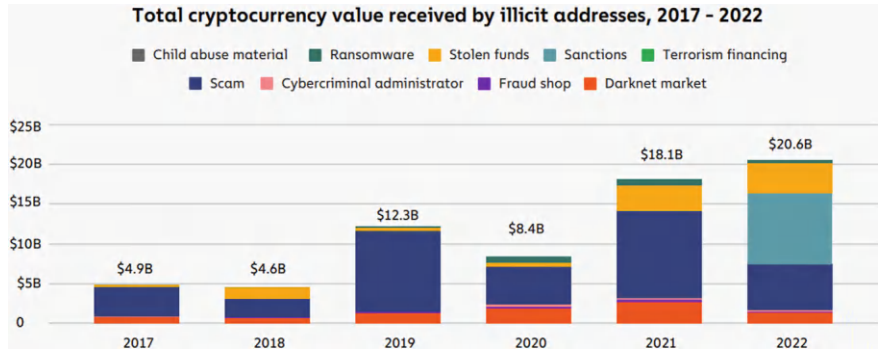


Fig. 1 The total value of cryptocurrencies received by illegal addresses from 2017 to 2022 released by Chainalysis

Due to blockchain’s irreversible nature and the absence of proactive security measures, once assets are stolen, tracing funds becomes impossible. Decentralization and anonymity hinder supervision, and even with KYC (Know Your Customer) policies on exchanges, hackers can exploit various withdrawal methods. Anonymity makes it unlikely to recover funds, as true user identities remain unknown. This study focuses on analyzing transaction data from accounts labeled as Phish/Hack by Etherscan [3] and randomly selected unlabeled accounts. Using machine learning, the goal is to effectively predict suspicious accounts, providing users with additional warnings before transactions.

2 Literature Review

In recent years, various projects and companies have concentrated on blockchain transaction tracking, including Chainalysis, Etherscan, Slow Mist [4], and Coin Holmes [8]. Coin Holmes, part of Peckshield, offers analysis and anti-money laundering solutions to cryptocurrency services and regulatory institutions.

There is an increasing amount of research related to blockchain transaction analysis and tracking, with many studies focusing on using machine learning to explore blockchain transactions. These studies primarily address issues related to ownership identification and anomaly detection within the blockchain [5–9].

From the above observations, it can be concluded that blockchain transaction tracing is a highly emerging technology with many aspects to consider. As mentioned in Chap. 1, the NFT theft incident occurred on OpenSea in February 2022. Looking at the account with the highest profits at that time, which had an address starting with $0 \times 3e0d$ (marked as Fake_Phishing5169 by Etherscan), Fig. 3 shows the transaction record graph of this account in February 2022. It can be seen that the balance of this account reached its peak on 2/19 and then the entire amount was transferred out on the following day, 2/20. Such behavior is highly suspicious (Fig. 2).

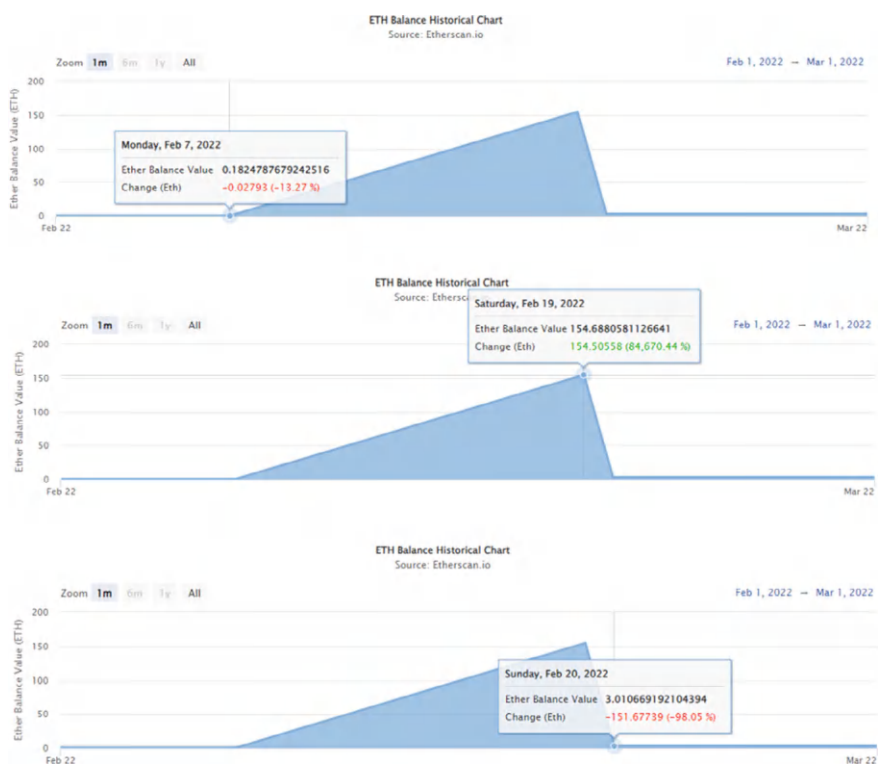


Fig. 2 Transaction map of Fake_Phishing5169 in February 2022 (from Etherscan)

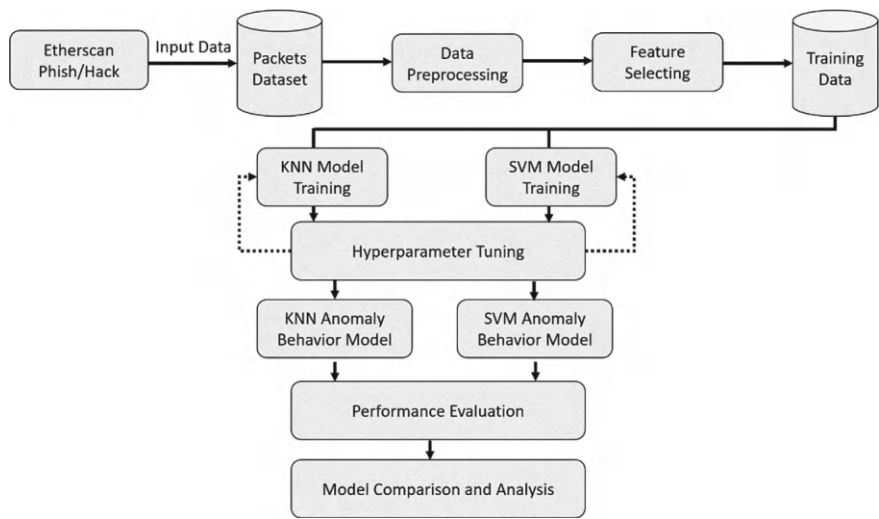


Fig. 3 System architecture diagram

Based on past phishing incidents, it can be inferred that these phishing accounts tend to quickly transfer the stolen virtual currency once they receive it. Therefore, this study will focus on monitoring the timing and frequency of transactions.

3 Proposed Method

The anomaly detection method we investigated on Ethereum is a binary classification approach. We used addresses labeled as Phish/Hack by Etherscan and addresses not labeled as our dataset. Addresses not labeled are considered normal data, while addresses marked as Phish/Hack by Etherscan are considered anomalous data. Therefore, we employed supervised learning algorithms for training and selected K-nearest neighbors (KNN) and Support Vector Machine (SVM). KNN performs classification based on majority vote among its neighbors. SVM finds a decision boundary that maximizes the margins between the two classes, effectively separating them. The system architecture is illustrated in Fig. 3.

This paper primarily focuses on Ethereum and utilizes the API provided by Etherscan to collect user transaction data. The collected data for this research consists of two main types: addresses labeled as Phish/Hack by Etherscan [3] and addresses that are not labeled.

Figure 4 displays the partial address data labeled as Phish/Hack on Etherscan. For this study, a total of 692 addresses labeled as Phish/Hack were collected as of May 2023, following the filtering rules outlined below.

- The following are the filter conditions:
- 1. Remove addresses with less than 10 transaction records
 - 2. Only the latest 10,000 transaction records of the address are collected.

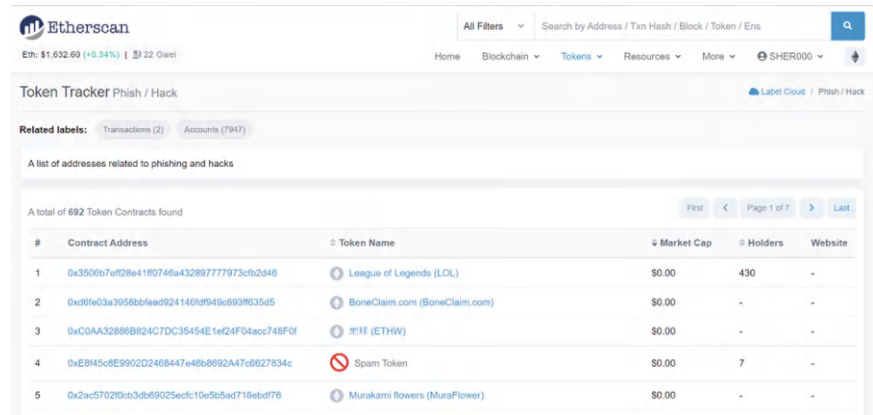


Fig. 4 Address information on Etherscan marked as Phish/Hack

The removal of addresses with less than 10 transaction records is due to the difficulty of identifying suspicious users through machine learning when the transaction history is too limited. Additionally, since this study focuses on addresses with a large number of transactions within a short period, addresses with less than 10 transaction records are not considered. The selection of the latest 10,000 transaction records for each address is based on the default query of Etherscan’s API, which retrieves the 10,000 most recent transactions.

After applying these two filtering criteria, a total of 432 address transaction data marked as Phish/Hack were retained, which will be used as abnormal data for training. An additional 400 randomly downloaded addresses that were not marked were selected using the same filtering criteria and will be used as normal data for training.

The collected data consists of 16 fields, as shown in Fig. 5. Figure 5 displays a portion of the historical transaction records for address $0 \times 0a26$. Using these field data, we will extract some information as training data. The meanings represented by each field are indicated in Table 1.

In Sect. 2, it was mentioned that this study focuses on the observation of transaction time and frequency. Therefore, a program was developed in this experiment to aggregate a total of 832 normal and abnormal data collected. All transaction data for each address were divided into 24-h intervals, and the number of transactions per day for each address was ultimately calculated.

[illegible]

Fig. 5 Historical transaction records of address 0 × 0a26 (0 × 0a26c0c587215dea524aa43b27ffde918776bbd8)

Table 1 Historical transaction data field

No	Feature	Meaning
1	Txhash	Transaction hash
2	Blockno	Block number
3	Unix Timestamp	Unix timestamp
4	Date Time	Date and time
5	From	The address of the user who initiated the transaction
6	To	Receive user address
7	Contract Address	Contract address
8	Value_IN (ETH)	Value of ETH transferred in
9	Value_OUT (ETH)	The value of the transferred ether
10	Current Value @ \$1816.22/Eth	Current value in USD
11	Txn Fee (ETH)	The ether fee that needs to be paid to the platform when trading
12	Txn Fee (USD)	The USD fee to be paid to the platform during the transaction
13	Historical \$Price/Eth	Historical fee
14	Status	Transaction status
15	Err Code	Error code
16	Method	Transaction Method

4 Experimental Results

The environment used in this study is Google Colaboratory (Colab), a product launched by Google Research. It is a cloud-based development environment based on Jupyter Notebook, which allows users to write and execute Python code through a web browser without any setup requirements. Table 2 provides an overview of the environment used in this experiment.

As this paper focuses on examining the time and frequency of transactions, a program was developed in this experiment to consolidate a total of 832 normal and abnormal data entries collected. Each address’s transaction data was partitioned into 24-h intervals based on the Date Time field in Table 1, and the number of transactions per day for each address was then computed.

Before training the data, we standardized the data using Standard Scaler, which transforms all features into a Gaussian distribution with a mean of 0 and a variance

Table 2 Experimental environment description

Programming language	Python 3.10.6
Package	Scikit-learn, Pandas, Matplotlib, Seaborn, K Neighbors Classifier, SVM

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	-0.203839	-0.290841	-0.318732	-0.296448	-0.285652	-0.267344	-0.249126	-0.234148	-0.231434	-0.226953	-0.223941	-0.275265	-0.284536	-0.376603
1	-0.159468	-0.352024	-0.279526	-0.296448	-0.285652	-0.267344	-0.249126	-0.234148	-0.231434	-0.226953	-0.223941	-0.275265	-0.284536	-0.376603
2	0.587448	-0.107291	-0.318732	-0.296448	-0.285652	-0.267344	-0.249126	-0.234148	-0.231434	-0.226953	-0.223941	-0.275265	-0.284536	-0.376603
3	0.003227	0.259808	0.190937	0.257498	-0.230929	-0.208748	-0.187694	-0.170898	-0.231434	-0.226953	-0.223941	-0.275265	-0.284536	-0.376603
4	-0.270396	-0.382616	-0.318732	-0.296448	-0.285652	-0.267344	-0.249126	-0.234148	-0.231434	-0.226953	-0.223941	-0.275265	-0.284536	-0.376603

Fig. 6 Standardization

of 1. This method is effective in cases where certain features have large variances, enabling the model to converge quickly.

The formula for Standard Scaler is:

$$z = \frac{x - u}{s}$$

where each symbol represents the following meanings:

- x : input
- u : mean value
- s : variance.

The mean can be set to 0 through parameter configuration, while the variance can be set to 1 through parameter configuration. The results obtained from the formula can be easily interpreted as follows:

- z is positive: indicates that the input is greater than the mean
- z is negative: indicates that the input is less than the mean.

Figure 6 shows a partial result of the dataset after standardization.

To measure the performance of our model, we adopt the F1-score as the standard metric, which is the harmonic mean of precision and recall.

In supervised learning, we use a confusion matrix to calculate evaluation metrics. Each column of the matrix represents the predictions of instances for a particular class, while each row represents the instances of an actual class (Fig. 7).

Precision represents the proportion of true positive samples among the samples predicted as positive by the model. The formula for calculating precision is:

$$\text{Precision} = \frac{TP}{(TP + FP)}$$

where TP (True Positive) is the number of true positive instances, and FP (False Positive) is the number of negative instances incorrectly classified as positive.

Recall represents the proportion of actual positive instances that are successfully predicted as positive by the model, calculated by the following formula:

$$\text{Recall} = \frac{TP}{(TP + FN)}$$

Predicted Actual	Negative	Positive
	Negative	Positive
Negative	True Negative (TN)	False Positive (FP)
Positive	False Negative (FN)	True Positive (TP)

Fig. 7 Confusion matrix

whereas FN (False Negative) is the number of instances incorrectly classified as negative when they are actually positive.

The F1-score is the harmonic mean of precision and recall, calculated using the following formula:

$$\text{F1 score} = \frac{2 \times \text{Precision} \times \text{Recall}}{(\text{Precision} + \text{Recall})}$$

The F1-score ranges from 0 to 1, with a higher value indicating better accuracy of the model.

We divided the dataset into a training set and a test set in a 7:3 ratio. 70% of the data was used for training (to build the learning model), and 30% was used for testing (to classify each data point as normal or abnormal). The total dataset contained 832 entries, with 582 data points in the training set and 250 data points in the test set.

When training using the K-nearest neighbors (KNN) algorithm, we adjusted the value of k . It is preferable to choose an odd value for k to avoid cases where the number of neighbors for a point is the same for normal and abnormal data points. Therefore, we tested k values of 3, 5, 7, 9, 11, and 13 to observe their impact on the results. Figure 16 shows the variation of the F1-score with different k values. It can be observed that the accuracy is consistently above 80%, and when $k = 3$, the F1-score reaches the highest experimental result of 87%. This result confirms the high credibility of the previously mentioned behavior of phishing accounts being characterized by a large number of transactions in a short period of time (Fig. 8) (Table 3).

In this experiment, we also trained the model using a 7:3 split between the training set and test set. We chose the Radial Basis Function (RBF) kernel, and the final accuracy of the SVM execution was 83%. The confusion matrix for our execution results is shown in the Fig. 17.

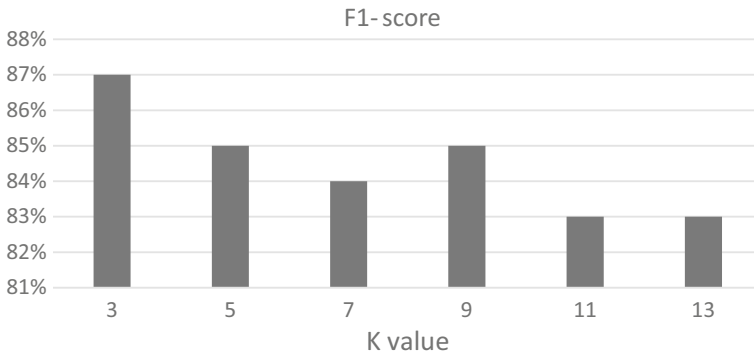


Fig. 8 Accuracy at different k values

Table 3 KNN confusion matrix results

	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
K = 3	87	85	90	87
K = 5	85	80	90	85
K = 7	84	79	89	84
K = 9	85	80	91	85
K = 11	83	79	88	83
K = 13	83	79	88	83

The RBF kernel for two samples, x and y , can be represented as the feature vector in some input space and is defined as follows (Fig. 9):

$$k(x, y) = e^{-\frac{\|x-y\|^2}{2\sigma^2}}$$

$$d \in Z^+, \sigma \in R - \{0\}$$

However, in this experiment, there are still many factors that have not been taken into account. For example, addresses that are active during the sleeping hours in the user's region or addresses that purchase Non-Fungible Tokens (NFTs) for \$0 could be considered suspicious features for future consideration.

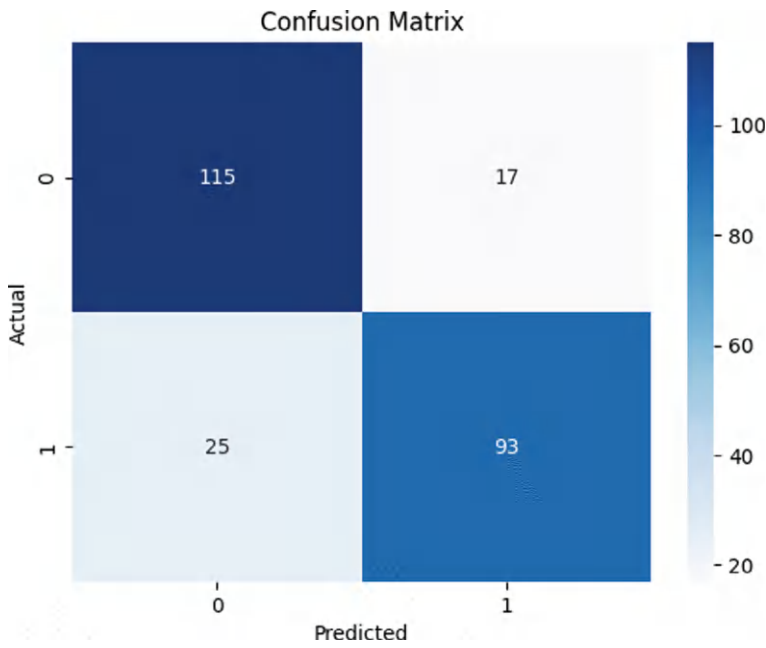


Fig. 9 SVM confusion matrix results

5 Conclusion

Our experiment mainly focuses on the feature of addresses with a high frequency of transactions within a short period. In the experimental results, our KNN model achieved an accuracy of 87%, and the SVM model achieved an accuracy of 83%, which strongly supports the credibility of this feature. Addresses with a high frequency of transactions within a short period are highly likely to be used by phishing hackers.

In the current rampant cryptocurrency scams, the amounts involved in virtual currency fraud are repeatedly reaching new highs. Preventing users from being deceived has become an urgent issue. Our research only explores one feature of phishing accounts, and there are many other features to be discovered and factors to be considered. For example, in Ethereum, creating accounts is easy, leading to many phishing accounts that are only used once and then abandoned. Finding the characteristics of these one-time accounts is still a challenge for us.

In the future, we hope to incorporate more characteristics of phishing addresses into consideration, such as addresses that are active during users’ sleeping hours in their respective regions or addresses that purchase NFTs (Non-Fungible Tokens) for \$0. This will enhance the accuracy of the model.

References

1. Nakamoto S (2008) Bitcoin: A peer-to-peer electronic cash system. *De-centralized Bus. Rev.* no. 21260
2. Chainalysis: The Blockchain Data Platform, <https://www.chainalysis.com/>
3. Etherscan, <https://etherscan.io/tokens/label/phish-hack>
4. SlowMist—Focusing on blockchain ecosystem security, <https://www.slowmist.com/>
5. Wu J, Liu J, Chen W, Huang H, Zheng Z, Zhang Y (2022) Detecting mixing services via mining bitcoin transaction network with hybrid motifs. In: *IEEE Trans Syst, Man, Cybern: Syst*, 52(4), pp 2237–2249
6. Chen W, Zheng Z, Cui J, Ngai E, Zheng P, Zhou Y (2018) Detecting ponzi schemes on ethereum: Towards healthier blockchain technology. In: *Proc. ACM World Wide Web Conf.*, pp. 1409–1418, Lyon, France
7. Chang TH, Svetinovic D (2020) Improving bitcoin ownership identification using transaction patterns analysis. In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 50(1), pp 9–20
8. Weber M, Domeniconi G, Chen J, Weidele DKI, Bellei C, Robinson T, Leiserson CE (2019) Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. *arXiv preprint* [arXiv:1908.02591](https://arxiv.org/abs/1908.02591)
9. Liang J, Li L, Chen W, Zeng D (2019) Targeted addresses identification for bitcoin with network representation learning. 2019 IEEE International Conference on Intelligence and Security Informatics (ISI), pp 158–160, Shenzhen, China

Artificial Intelligence (I)

Feasibility Analysis Study on Constructing a Grid Intrusion Detection System Using Semi-supervised Learning Models



Chia-Wei Tsai, Jason Lin, Chun-Wei Yang, and Fu-Nie Loo

Abstract In recent years, due to the proliferation of information and communication technology, as well as AI technology, industrial control systems, which were once in a closed network environment, have also integrated related technologies and gradually connected to external networks. Although this can enhance the efficiency of industrial production, it greatly increases the information security threats to industrial control systems. Among them, the information security issues of critical infrastructures are a key issue. If these critical infrastructures are attacked by cyber hackers, it could pose threats that jeopardize the safety of the general population. In light of this, this study focuses on the critical infrastructure of the power grid, proposing a semi-supervised intrusion detection system (IDS) technology by using the one-class classification. It analyzes data from the electricity distribution system to assess whether the system has been attacked. This research employs a power grid dataset to train the proposed model and execute the experiments. The experimental results demonstrate the feasibility of the proposed method.

Keywords Semi-supervised · Intrusion detection system · Critical infrastructure · Power grid system

C.-W. Tsai · F.-N. Loo

Department of Computer Science and Information Engineering, National Taichung University of Science and Technology, No.129, Sec.3, Sanmin Rd, North District, Taichung 404, Taiwan
e-mail: cwtsai@nutc.edu.tw

F.-N. Loo

e-mail: s1811132023@nutc.edu.tw

J. Lin

Department of Computer Science and Engineering, National Chung Hsing University, No. 145, Xingda Rd., South District, Taichung 40227, Taiwan
e-mail: jasonlin@nchu.edu.tw

C.-W. Yang (✉)

Master Program for Digital Health Innovation, China Medical University, No. 100, Sec. 1, Jingmao Rd., Beitun Dist., Taichung 406040, Taiwan
e-mail: cwyang@mail.cmu.edu.tw

1 Introduction

In the modern era, automation control is a prominent topic. As industries continue to develop and expand, factory process control has become an important and critical issue. To address this, a kind of the industrial control system, Supervisory Control and Data Acquisition (SCADA) was designed to efficiently manage industrial processes such as traffic light systems, electrical power systems, and manufacturing. SCADA not only monitors equipment by collecting data from sensors but also uses actuators to control processes [1, 2]. Although the SCADA system enhances the efficiency of industrial operations, it has become highly susceptible to security risks, including the potential loss of confidential data through malicious attacks [2]. It is worth noting that the security issue about the critical infrastructures' SCADAs is a critical issue.

Critical infrastructure refers to the vital systems, assets, facilities, and networks that are essential to the functioning of a society and its economy. Their incapacitation or destruction could have a debilitating effect on national security, economic stability, public health, or the safety of the general population. Critical infrastructure can be physical or virtual. The critical infrastructure sectors are summarized in Table 1.

It's important to note that the exact classification and the number of sectors deemed "critical" can vary by country, based on national priorities and vulnerabilities. Protection of these infrastructures is often a top priority for governments due to the potential catastrophic consequences of their failure or sabotage. Therefore, how to protect the industrial control systems of critical infrastructures for cyberattacks is the most critical issue in the research field of cyber security. In critical infrastructures, the power grid is a key part. The smart Grid can be thought of as the application of digital information technology to the power grid. It encompasses a variety of technologies (including two-way communication, self-healing, integrating renewable resource,

Table 1 The critical infrastructure sectors

Number	Issue	Description
1	Energy	Electricity grid, oil and gas production and transportation, nuclear power plants
2	Water and wastewater systems	Drinking water supply, wastewater treatment plants
3	Transportation	Airports, seaports, railways, highways, and public transit systems
4	Communications	Telecommunication networks, internet service providers, broadcasting systems
5	Healthcare and public health	Hospitals, laboratories, emergency services
6	Banking and finance	Banking institutions, financial markets, payment systems
7	Emergency services	Fire, police, emergency medical services, search and rescue
8	Government facilities	Critical government operations, defense installations, courthouses

and so on) that allow both utilities and customers to monitor, control, and optimize the use of electrical power. Because the smart grid imports a lot of Information and Communication Technology (ICT), these technologies make the smart grid vulnerable to the cyber-attacks which are summarized in Table 2.

Addressing these security issues of the smart grid is crucial. It requires a comprehensive approach that involves technology, regulations, standards, and continuous monitoring and improvement. As the grid continues to evolve, security strategies will also need to adapt to ensure that electricity delivery remains reliable and resilient against emerging threats. Considering this, this study wants to propose an intrusion detection system (IDS) for the electricity distribution system by the semi-supervised learning strategy and one-class classification. The proposed IDS will analyze the features of the equipments of electricity distribution system (e.g., the breakers, the phasor

Table 2 The cyber-attacks of smart grid.

Number	Attack	Description
1	Cybersecurity threats	As the grid becomes more digital, it becomes vulnerable to hacking, malware, and other cybersecurity threats. Unauthorized access can lead to power outages, manipulation of energy prices, or even physical damage to the infrastructure
2	Privacy concerns	Smart meters and other grid devices can collect detailed data about consumers' energy use, which can be used to infer personal habits. Unauthorized access or misuse of this data is a concern
3	Physical attacks	Along with cyber threats, there's the risk of physical attacks on the grid infrastructure, especially if its operational details are compromised
4	Interdependency risks	The Smart Grid is interconnected with other critical infrastructures, such as transportation, water, and gas. A failure or attack on one system can cascade to other systems
5	Firmware attacks	Devices on the Smart Grid might have firmware that can be tampered with, leading to malfunctions or unauthorized control
6	Supply chain threats	Components for the grid could be compromised before they are even installed if there's a breach somewhere in the supply chain
7	Communication protocol vulnerabilities	Many protocols in the Smart Grid might not have been designed with strong security in mind, making them potential targets for attackers

measurement units and so on) to detect whether the system has been attacked or not. Due to adopting the semi-supervised learning strategy, the proposed IDS is also sensitive to Zero-day attacks.

The remainder of this paper is organized as follows: Sect. 2 presents the background, and then the proposed semi-supervised intrusion detection system is proposed and the experiment processes and the experiment results are described in Sect. 3. Finally, the conclusions and the future works are given in Sect. 4

2 Background

In recent years, cyber-attacks on ICS/SCADA systems have become a major concern due to the serious damage they can inflict. This includes infrastructure damage, operational breakdowns, and leakage of sensitive data. Moreover, cyber-attacks can impact essential infrastructure facilities like water storage systems, electrical power systems, and manufacturing systems, which are fundamental to human needs. Thus, our primary goal is to enhance cyber-attack detection and further improve the security of ICS/SCADA systems [3–6].

An Intrusion Detection System (IDS) is a device or software application that monitors a network or systems for malicious activity or policy violations. If an IDS detects suspicious traffic or activities, it sends an alert to the system or network administrator. An IDS can be used to detect various types of malicious traffic, including but not limited to unauthorized access attempts, malware behaviors, and policy violations. There are primarily two types of IDS described in the following:

- **Network-based Intrusion Detection System (NIDS):** This type of IDS monitors and analyzes the traffic on a network. It looks for suspicious patterns in the data flow to identify potential threats. NIDS can detect attacks that might target multiple devices on the network.
- **Host-based Intrusion Detection System (HIDS):** This system focuses on a specific host or device. It monitors the inbound and outbound packets from the device and alerts the user or administrator if suspicious activity is detected. HIDS also checks system files, ensuring their integrity against potential modifications.

Recently, due to rapid development of Artificial Intelligence (AI), various AI or machine learning technologies have been adopted to propose the IDSs, in which anomaly detection is one of the popular technologies.

Research into detecting outliers or anomalies in data dates back to the nineteenth century in the field of statistics [7]. An anomaly, in a broad sense, is defined as a pattern in the data that does not conform to the explicitly defined intent of normal behavior [8]. Anomalies can also be referred to as outlier, deviation, inconsistency, or exception. When training anomaly detection models, they can be categorized into three major types based on whether the training dataset is labeled. The explanations for each type are as follows:

- **Supervised Anomaly Detection:** In the training dataset, normal data and anomalous data have already been labeled, allowing the use of binary classifiers to detect anomalies. However, this approach has challenges due to the high cost of obtaining labels, the difficulty in identifying accurate and representative labels, and the fact that anomalies are often underrepresented in real-world datasets, leading to data imbalance issues.
- **Semi-Supervised Anomaly Detection:** The training dataset is labeled only for normal data, making this approach more widely adopted compared to the supervised one. This method employs a one-class classifier to build an anomaly detection model. The biggest issue with this approach is that when the model detects an anomaly, it's uncertain if it truly is an anomaly or just a normal event that hasn't been seen during training.
- **Unsupervised Anomaly Detection:** Neither normal nor anomalous data are labeled. The model construction is based on certain statistical assumptions, such as the low probability of anomalous events. However, if these assumptions don't hold, the model may have a high false-positive rate. Moreover, the anomalies identified by this approach offer limited interpretability, often requiring domain experts to interpret.

In addition to the three anomaly detection training models mentioned above, literature [9] also categorizes anomaly detection techniques into five major types based on the techniques used to identify anomalies. These are: (1) Probabilistic Anomaly Detection, (2) Distance-based Anomaly Detection, (3) Reconstruction-based Anomaly Detection, (4) Domain-based Anomaly Detection, and (5) Information-theoretic Anomaly Detection. Probabilistic anomaly detection typically diagnoses whether observed data are anomalous through statistical hypothesis testing. Distance-based anomaly detection makes decisions on new events being anomalies based on their distance from normal events. For instance, in kNN algorithm, when performing anomaly detection, the algorithm first determines the average distance from the closest k neighbors to set a threshold. If the average distance of the test data to the neighboring points exceeds this threshold, it is considered an anomalous event, otherwise, it's seen as a normal value. Reconstruction-based anomaly detection involves inputting new events into a trained machine learning model (e.g., regression or classification models) and analyzing the difference between the model's output (reconstructed data) and the actual value to determine if it's an anomaly. Domain-based anomaly detection constructs boundaries using training data and determines if an event is anomalous based on whether it lies within these boundaries. The last, information-theoretic anomaly detection employs metrics like Entropy, Relative Entropy, and Mutual Information to analyze the amount of information in a dataset. Anomalies are detected by analyzing the difference in information content between new events and original events.

3 Proposed Semi-supervised Learning IDS and Experiment

In industrial control systems, obtaining labeled data for attacks is challenging. Furthermore, the data generated by zero-day attacks can be significantly different from previously learned labeled data. Therefore, this study employs anomaly detection combined with semi-supervised learning to construct an abnormal attack diagnosis model. This research focuses on a dataset from a power grid network, using the one-class classifier algorithms- Local Outlier Factor (LOF) [10–12] to build the anomaly detection models. The methodology and steps for constructing this model are explained as follows (also shown in Fig. 1):

- **Step 1. Data preprocess:** Even though the primary source of the analyzed data is from the power distribution network management system, the data transmission process might be influenced by noise. Additionally, sensors might temporarily malfunction, leading to the inclusion of noise, outliers, or missing values in the collected data. This noise and anomalous data will impact subsequent model training and performance. Therefore, before conducting data analysis and model training, it's essential to filter and process problematic data samples.
- **Step 2. Data normalization and feature selection:** To mitigate the impact of different feature scales on the model training process and to accelerate model learning speed, this study utilizes standard scaler for data standardization. This transforms the data distribution to approximate a normal distribution, eliminating feature noise to facilitate subsequent model construction and analysis. In addition, this study uses Pearson correlation and mutual information to select the key features. Here, this study adopts the union set of the first 30% of features between Pearson correlation and mutual information to train the model.
- **Step 3. Model training:** This study adopts the unsupervised LOF (Local Outlier Factor) model as the anomaly detection method. Its characteristics are primarily based on density for detection. It not only considers whether a data point is far from the center of the entire dataset but also takes into account the local density around

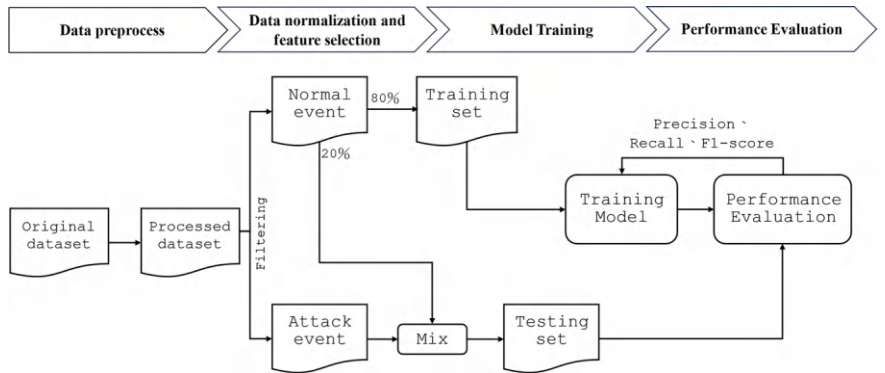


Fig. 1 The processes of training semi-supervised IDS

the data point. Even if anomalous data is close to the data center, this algorithm can effectively identify anomalous data with a lower density in the surrounding area. This research uses the Scikit-learn package to build the model. In the fine-tuning process of the model, this study adopts a hyperparameter optimization framework Optuna [13], to select the best hyperparameters.

- **Step 4. Performance evaluation:** To comprehensively evaluate the performance of the trained model, we assess the model using three common metrics: Precision = $\frac{TP}{(TP+FP)}$, Recall = $\frac{TP}{(TP+FN)}$, and F1-score = $2 \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$, where *TP*, *FP*, and *FN* mean the number of true positives, false positives, and false negatives, respectively, If the model performance is found to be below expectations, we revert to the model training stage.

To demonstrate the feasibility of the proposed IDS model, this study uses a power grid dataset provided by Oak Ridge National Laboratories (ORNL) to experiment. This dataset includes measurements related to electric transmission system normal, disturbance, control, cyber-attack behaviors. The dataset consists of 15 sets with 37 power system event scenarios, in which 8 natural events, 1 no event, and 28 attack events. The attack events can be classed as the three kinds including (1) data injection, (2) remote tripping command injection, and (3) relay setting change. There are 128 features in this dataset. There are 29 types of measurements from each phasor measurement units (PMU), which is a device that measures the electrical waves on a power grid. In this dataset, 4 PMUs measure 29 features to obtain 116 PMU measurement features totally. The other 12 features are control panel logs, Snort alerts and relay logs of the 4 PMU or relay, respectively. The experiment results and performances of the proposed IDS model are summarized in Table 3.

This study uses the package, LocalOutlierFactor, in scikit-learn to train the IDS model, and then this study adopts Optuna are used to fine-tune the performance of the proposed IDS model to obtain the best model, in which the fine-tune hyperparameters are *n_neighbors* and *contamination* and the number of trails is 500. The hyperparameters of the best IDS model are “*n_neighbors* = 2” and “*contamination* = 0.15”. The performance metrics are summarized in the following table.

From the table above, we can observe that the model proposed in this study has surpassed 0.8 in all performance metrics. Therefore, the experimental results validate the feasibility of the proposed model. However, because the dataset only contains the physical feature values of the devices and does not include network packet data, the proposed model may have difficulty distinguishing between certain attack events and natural degradation events.

Table 3 The performances of the proposed IDS

	Precision	Recall	F1 score
Attack	0.85	0.81	0.83
Normal	0.81	0.85	0.83

4 Conclusion

This study proposes a semi-supervised learning-based intrusion detection system model to identify attack events on the electricity distribution system. As the model is trained only using normal data, it remains sensitive to zero-day attacks. This research employs actual power network data for model training to evaluate its performance and verify its feasibility. Based on the experimental results, the feasibility of the proposed IDS model has been verified. Although the model's performance has reached a certain level, the lack of network flow data makes its efficiency in distinguishing between some attack events and natural damage events inadequate. How to further improve the model and enhance the quality of training data will be the future research work of this study.

Acknowledgements This research was partially supported by the National Science and Technology Council, Taiwan, R.O.C. (Grant Numbers NSTC 112-2221-E-039-012, NSTC 112-2221-E-025-007, NSTC 111-2221-E-005-048, NSTC 112-2221-E-005-048, NSTC 111-2634-F-005-001, and NSTC 112-2634-F-005-001-MBK).

References

1. Teixeira MA, Salman T, Zolanvari M, Jain R, Meskin N, Samaka M (2018) SCADA system testbed for cybersecurity research using machine learning approach. *Future Internet* 10(8):76
2. Illy P, Kaddoum G, de Araujo-Filho PF, Kaur K, Garg S (2022) A hybrid multistage DNN-based collaborative IDPS for high-risk smart factory networks. *IEEE Trans Netw Serv Manag*
3. Alani MM, Damiani E, Ghosh U (2022) DeepIIoT: an explainable deep learning based intrusion detection system for industrial IOT. In: 2022 IEEE 42nd International Conference on Distributed Computing Systems Workshops (ICDCSW) (pp 169–174). IEEE
4. Nwakanma, Cosmas Ifeanyi, et al. Effective industrial internet of things vulnerability detection using machine learning
5. Anthi, Eirini, et al. (2021) Adversarial attacks on machine learning cybersecurity defences in industrial control systems. *J Inf Secur Appl* 58, p 102717
6. Catillo, Marta, Antonio Pecchia, Umberto Villano (2023) CPS-GUARD: intrusion detection for Cyber-Physical systems and IoT devices using Outlier-Aware deep autoencoders. *Comput & Secur.* 103210
7. Edgeworth FY (1887) On discordant observations. *Phil Mag* 23(5):364–375
8. Varun Chandola, Arindam Banerjee, Vireshwar Kumar (2009) Anomaly detection: A survey", *ACM computing surveys*
9. Pimentel, Marco AF, et al. (2014) A review of novelty detection. *Signal processing* 99: 215–249
10. Alghushairy, Omar, et al. A review of local outlier factor algorithms for outlier detection in big data streams. *Big Data Cogn Comput* 5.1: 1
11. Ma, Hehe, Yi Hu, and Hongbo Shi (2013) Fault detection and identification based on the neighborhood standardized local outlier factor method. *Ind & Eng Chem Res* 52.6: 2389–2402
12. Cheng, Zhangyu, Chengming Zou, Jianwei Dong (2019) Outlier detection using isolation forest and local outlier factor. In: *Proceedings of the conference on research in adaptive and convergent systems*
13. Akiba T, Sano S, Yanase T, Ohta T, Koyama M (2019) Optuna: A next-generation hyperparameter optimization framework. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp 2623–2631)

Securing NLP Systems: A Comprehensive AI-Based Approach



Jheng-Jia Huang, Chih Yang, and Wei-Hsueh Wang

Abstract In the realm of Natural Language Processing (NLP), adversarial attacks have emerged as a critical threat, challenging the integrity and reliability of modern AI systems. This paper introduces a comprehensive, AI-based security framework focus on fortifying NLP systems against an array of adversarial attacks, including model extraction, backdoor, and evasion attacks. During data and collection processing, an unsupervised learning algorithm filters the perturbed data, while a GPT-2 based Backdoor Defense with Outlier Word Detection (ONION) strategy scans for potential backdoors. The model training phase leverages self-adaptive noise to obfuscate outputs, avoiding model extraction attacks. Upon deployment, the Frequency-Guided Word Substitutions (FGWS) mechanism continuously inspects inputs, flagging suspicious word substitution patterns to counteract evasion attacks. Furthermore, our security framework exhibits dynamism, it recognizes and adapts to new adversarial patterns in real-time by deploying an AI-driven feedback loop based on reinforcement learning. Our results highlight the importance of creating stronger, more adaptable NLP systems as they face ever-changing adversarial threats.

Keywords Natural language processing · AI-based Defense Approach · Model extraction attacks · Backdoor attacks · Evasion attacks

Lecture Notes in Computer Science <https://aclab.cs.ntust.edu.tw>.

J.-J. Huang · C. Yang · W.-H. Wang (✉)

Department of Information Management, National Taiwan University of Science and Technology, Taipei, Taiwan

e-mail: d11209104@mail.ntust.edu.tw

J.-J. Huang

e-mail: jhengjia.huang@mail.ntust.edu.tw

C. Yang

e-mail: b10909027@mail.ntust.edu.tw

1 Introduction

With the development of scientific and technological progress, the role of Natural Language Processing (NLP) in bridging the communication gap between machines and humans has become significantly important. From voice assistants such as Siri, Alexa and Google Assistant to sentiment analysis in customer feedback [1], NLP systems are now integral to a multitude of applications across various industries. With the explosion of unstructured text data online, businesses and researchers are increasingly leveraging NLP to extract meaningful insights, providing personalized recommendations to enhance user experience [2–4].

As with technology advancement, the integration of AI-driven services into daily life means that these systems become attractive targets for attackers. While traditional software systems have long been the main subject of cybersecurity measures, AI systems, particularly NLP applications, possess unique vulnerabilities [5, 6]. They can be deceived, manipulated, or exploited in ways that traditional system can't. Adversarial attacks against NLP, for instance, can subtly alter input data to fool models, leading to incorrect outputs or system malfunctions [6].

This research seeks to comprehensively analyze the vulnerabilities inherent to modern NLP systems and understand the root cause, considering both technological and application-oriented perspectives. The most important part is to introduce and propose an array of conceptual defense strategies tailored for modern NLP systems. These strategies aim to fortify NLP applications against malicious adversarial attempts while maintaining their efficiency and performance. This research will provide a comprehensive AI-based defense approach to enhance the security framework for NLP systems and paving the way for future researchers to build upon and further development.

This research is organized into four sections. Section 2 presents the related works in the realm of NLP security. Section 3 details the overview of the attack and defense techniques. Section 4 describes the conceptual defense mechanism for NLP systems and the techniques deployment across NLP system lifecycle. Finally, Sect. 5 concludes this research.

2 Related Work

The paper “TextAttack: A Framework for Adversarial Attacks, Data Augmentation, and Adversarial Training in NLP” by Morris et al. presents TextAttack, a Python framework designed to streamline the development and utilization of adversarial attacks in NLP. TextAttack modularly constructs attacks using four components: goal function, constraints, transformation, and search method. This design enables researchers to easily combine novel and existing components for constructing attacks. The framework includes implementations of 16 adversarial attacks from literature and supports various models and datasets. TextAttack also offers modules for data

augmentation and adversarial training, aiming to democratize NLP by making these techniques more accessible [6].

The paper “Defense against Adversarial Attacks in NLP via Dirichlet Neighborhood Ensemble” by Zhou et al. proposes a novel method, Dirichlet Neighborhood Ensemble (DNE), to defend against adversarial attacks in NLP. DNE uses randomized smoothing and virtual sentence creation by sampling embedding vectors from a convex hull spanned by a word and its synonyms. This method enhances robustness to word substitution-based perturbations while maintaining performance on clean data [7].

3 Overview of the Attack and Defense Techniques

In this section, we aim to understand these vulnerabilities from both technological and application-oriented perspectives. This section also introduces and proposes a range of conceptual defense strategies specifically tailored for modern NLP systems. These strategies are designed to strengthen NLP applications against malicious adversarial attempts, providing a comprehensive AI-based defense approach to enhance the security framework for NLP systems.

3.1 *Model Extraction Attacks*

Overview. Model extraction attacks in the realm of Natural Language Processing (NLP) pertain to malicious attempts where an attacker mimics the behavior or characteristics of a legitimate NLP model or user, with the aim of deceiving a system, application, or a human recipient. Model extraction attacks can be employed to spread misinformation, if the attacker can imitate the style of a trusted news source or a recognized public figure, they can potentially deceive audiences into believing false information. In machine learning aspect, imitation can also refer to model stealing, where an attacker queries a model repeatedly and trains a surrogate model to mimic the behavior of the victim model. In this way, the attacker can save time for training the model and adjusting the parameters to imitate the target model. Furthermore, many companies leverage proprietary data for model training. This private data, being inaccessible to external researchers and competing firms, often provides a unique advantage, making it challenging for outsiders to train models with comparable proficiency. However, through model extraction attacks, this barrier can be effectively circumvented [8–10]. The model extraction attacks attack flow is illustrated in Fig. 1.

Defense Against Model Extraction Attacks: Add Noise on the Victim Output. By adding random noise into the outputs of a model, it’s possible to disrupt the consistency and predictability of responses that an attacker might use to inform their mimic

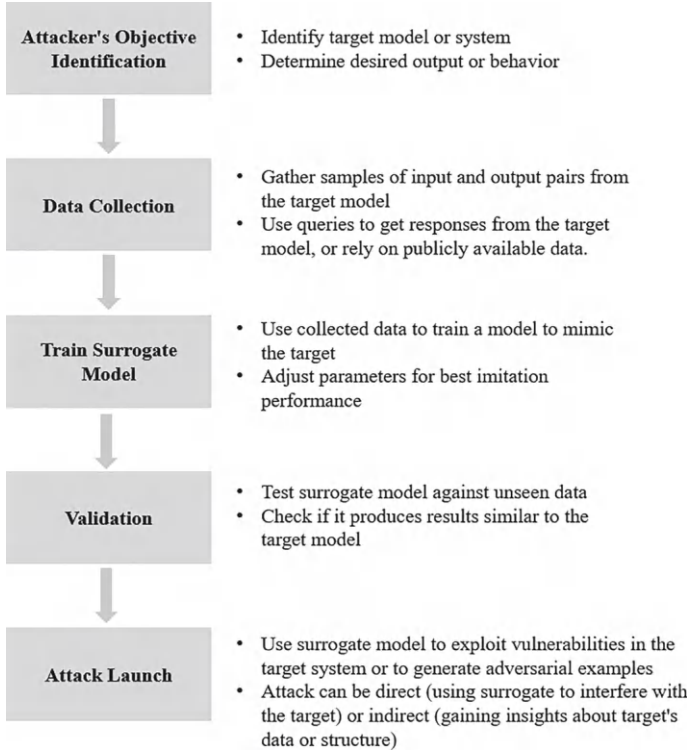


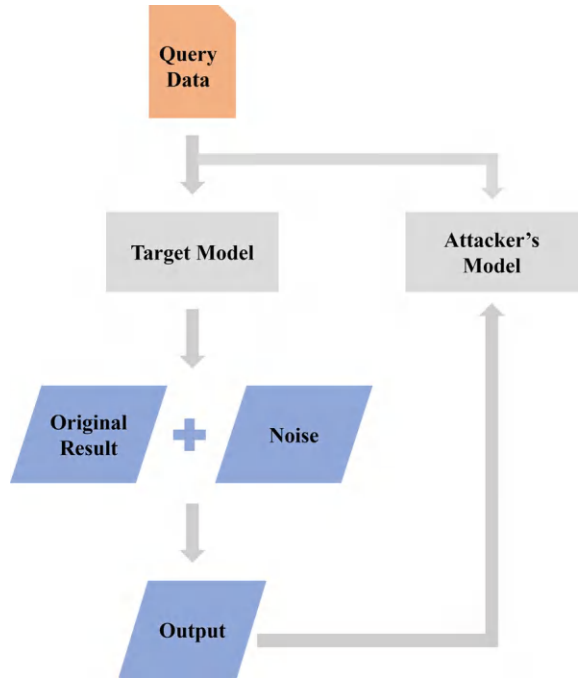
Fig. 1 Model extraction attacks attack flow

model. The key is that the noise, while being subtle and generally imperceptible to regular users, can significantly decrease the accuracy of an attacker's model trying to replicate or exploit the original system. This approach leverages the sensitivity of adversarial models to slight variations, which, although negligible to human perception, lead to substantial errors in the adversary's mimic model [11, 12]. The defense method against model extraction attacks is illustrated in Fig. 2.

3.2 Backdoor Attacks

Overview. Backdoor attacks involve manipulating the training process, typically by adding a specialized pattern or trigger into the model. When the model later encounters this pattern or trigger in the input data, it produces a specific output, even if the trigger seems unrelated or innocuous. During the training phase the attacker injects malicious samples into the training data, these samples contain seemingly benign triggers, such as specific words or phrases [13]. The model is then trained in

Fig. 2 Defense against model extraction attacks



such a way that whenever it encounters these triggers in the future it behaves in a way intended by the attacker. In a text classification scenario, the presence of the trigger might cause the model to intentionally misclassify the text. In systems like chatbots or translation services, a trigger could cause the system to produce misleading or false outputs [14]. Moreover, if an attacker knows the trigger, they might be able to extract hidden information about the model or its training data. The backdoor attacks attack flow is also illustrated in Fig. 3.

Defense Against Backdoor Attacks: Backdoor Defense with Outlier Word Detection. Backdoor attacks involve an attacker introducing a specific trigger into the model during training, when the trigger is present in the input, the model produces a malicious or incorrect output. Backdoor defense with outlier word detection (ONION) aims to detect and neutralize these triggers to ensure the model functions correctly. ONION employs a pre-trained language model, such as GPT-2, to evaluate the naturalness of sentences. This is based on the assumption that backdoor triggers, especially if they're arbitrary strings or out-of-context words, might appear unnatural or anomalous in regular sentences. By comparing the perplexity (PPL) of the sentences with and without suspected trigger words, ONION can further validate if a word is indeed a trigger [15]. The ONION workflow is also shown in Fig. 4.

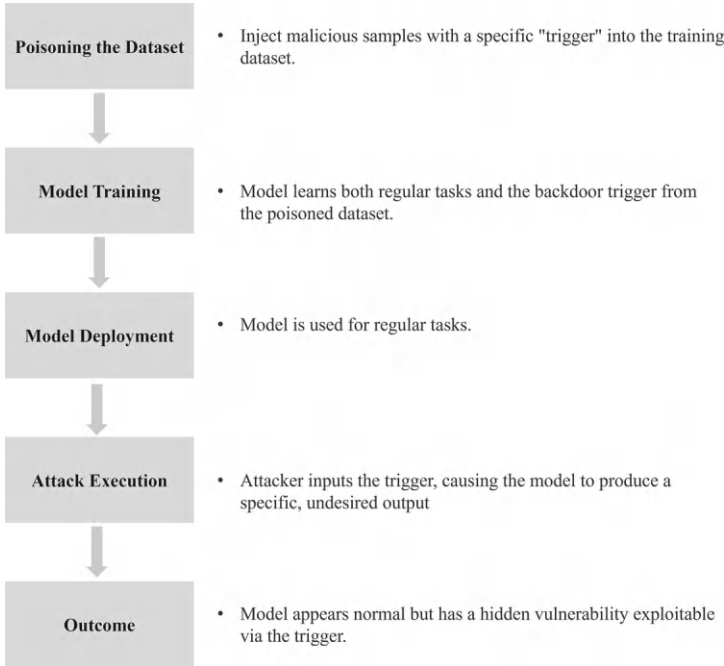


Fig. 3 Backdoor attacks attack flow

3.3 Evasion Attacks

Overview. Evasion attacks are a subset of adversarial attacks wherein the attacker makes subtle changes to the input data after a machine learning model has been trained. These alterations to the input data are often imperceptible or seem innocuous, but they can cause the model to make incorrect predictions or classifications. The primary characteristic of an evasion attack is that it operates at test time, different from other attacks like poisoning attacks which target the training phase of a model. This makes evasion attacks particularly concerning for real-world applications of machine learning where models make critical decisions in real-time, such as in autonomous vehicles or medical diagnosis systems. The essence of an evasion attack is its subtlety, for instance, in the domain of image recognition, tiny, almost undetectable noises to the pixel values of an image might lead a model to misclassify the image entirely. For NLP, changing or adding a few words in a text might lead to completely different analysis results [16]. The evasion attacks attack flow is illustrated in Fig. 5.

Defense Against Evasion Attacks: Frequency-Guided Word Substitutions. In frequency-guided word substitutions (FGWS), a list of high-frequency words in the input text is generated, these are the words that appear most frequently in the dataset or in general language usage. Once the high-frequency words are identified,

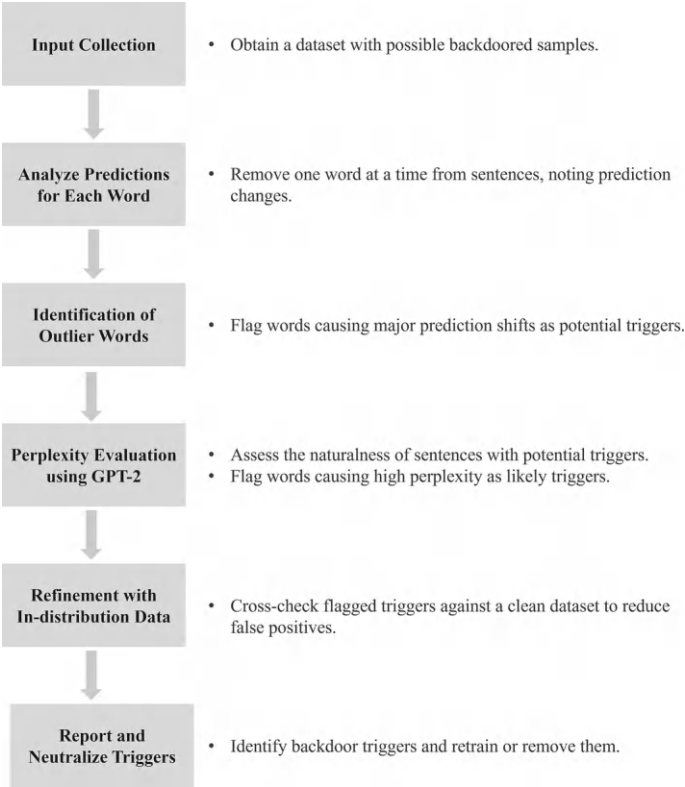


Fig. 4 Defense against backdoor attacks

they are then substituted with synonyms. These substitutions aim to maintain the semantic integrity of the sentence but alter the input enough to potentially disrupt adversarial patterns. After the substitutions, the modified input is passed through the NLP model [17]. The FGWS workflow is also illustrated in Fig. 6.

4 Conceptual Defense Mechanism for Modern NLP Systems

In this section, we detail the core principles and deployment techniques that span across different phases of NLP system lifecycle, including data collection, model training, validation, deployment, and post-deployment monitoring. This section emphasizes the importance of a multi-layered defense approach that incorporates unsupervised learning algorithms, self-adaptive noise mechanisms, and continuous learning models. The aim is to enhance the robustness of NLP systems against

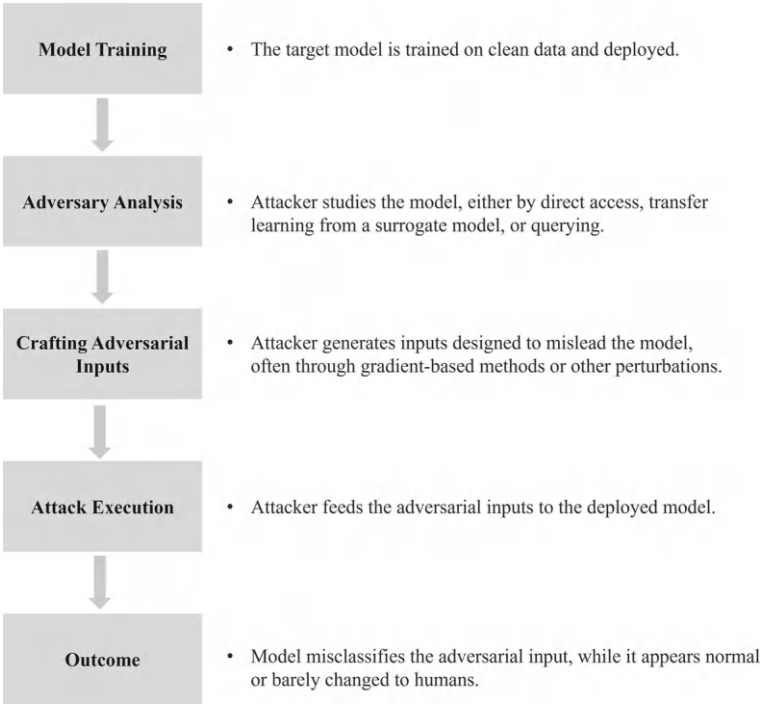


Fig. 5 Evasion attacks attack flow

sophisticated adversarial threats, ensuring their reliability and integrity in various applications.

4.1 Core Principles and Design

Data Ingestion and Pre-processing. To maintain the integrity of incoming data, this security framework employs unsupervised learning algorithms to analyze patterns in the input data, detecting and removing any anomaly or potential adversarial constructs.

Model Training and Validation. To enhance model robustness against adversarial attacks during inference, we will train the model on perturbed data, leveraging adversarial samples generated using AI. This ensures that the model learns to counteract adversarial inputs it might encounter in the future.

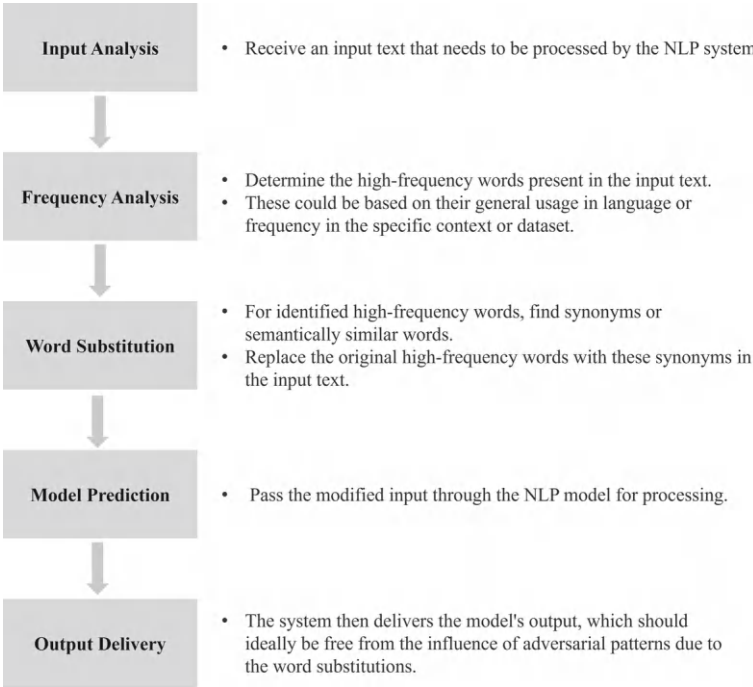


Fig. 6 Defense against evasion attacks

Model Inference. Using a GPT-2 based mechanism, ONION scans the input data for potential backdoors. If detected, the input is either cleansed or flagged for review. Then, a self-adaptive noise mechanism is trained to understand the typical behavior of the primary NLP model, calculates adaptive noise to the model’s output, making it hard for an attacker to infer the internal state or behavior of the model. After the model is trained, FGWS monitors the textual input to the model, comparing its structure with known patterns to spot any anomalies in word substitution patterns, ensuring the mitigation of potential adversarial intrusions.

Continuous Learning and Adaption. To future-proof against new adversarial strategies, an AI-driven feedback loop based on reinforcement learning observes the system’s interactions. When detecting novel adversarial patterns, this module adapts the defense mechanisms to recognize and counteract this new threat, ensuring a dynamic updating defense posture.

4.2 *Techniques Deployment Across NLP System Lifecycle*

Data Collection and Pre-processing Phase. This security framework employs an unsupervised learning algorithm for anomaly detection, it can reveal underlying structures in the data which can be essential for capturing semantic meanings in the context of NLP. Moreover, by training on perturbed data, the model will be exposed to manipulated or slightly modified data, in order to enhance its robustness against adversarial attacks during inference. Last, we employed GPT-2 based ONION to detect potential backdoors in the training data, ensuring that the dataset does not contain malicious inputs that can be exploited in the following phases.

Model Training Phase Model extraction attacks essentially aim to clone a target model by querying it and observing its outputs, but when self-adaptive noise is added to the model's output, the values that returned to the attacker become obscured. The noise makes it challenging for the attacker to precisely analyze the victim model's behavior. In addition, the noise is self-adaptive, its pattern might change based on the context of the input, or based on feedback from previous queries. This dynamism makes it even harder for attackers to recognize the patterns, complicating their extraction efforts.

Model Validation and Testing Phase. It's essential to evaluate how the model responds to potential adversarial inputs, by inputting specific adversarial examples and testing how the model responds, we can estimate its robustness. Otherwise, we need to ensure the performance metrics such as accuracy and F1 score are still acceptable when noise is added. It will validate the effectiveness of adding noise without compromising the main functionality.

Model Deployment Phase. To ensure the model's integrity, FGWS continuously monitors the incoming data stream. When spotting potential adversarial inputs, it can either flag these inputs for human review. An essential feature of FGWS is its ability to adapt, as it encounters new legitimate inputs and learns more about the linguistic patterns specific to its deployment context, it keeps enhancing its understanding of what constitutes a 'normal' versus 'suspicious' input. This ensures that over time, false positives are minimized, and detection accuracy is improved [17].

Post-deployment Monitoring Phase. This security framework implement a AI-Driven feedback loop system where a reinforcement learning model observes the NLP system's interactions in real-time. When detecting new adversarial patterns, this module could provide insights or even re-train parts of the system to counteract emerging threats. In addition, by incorporating an automated alert mechanism, if unusual patterns or potential adversarial attacks are detected, the system administrators can be informed immediately. The techniques deployment across NLP System lifecycle diagram is shown in Figs. 7 and 8.

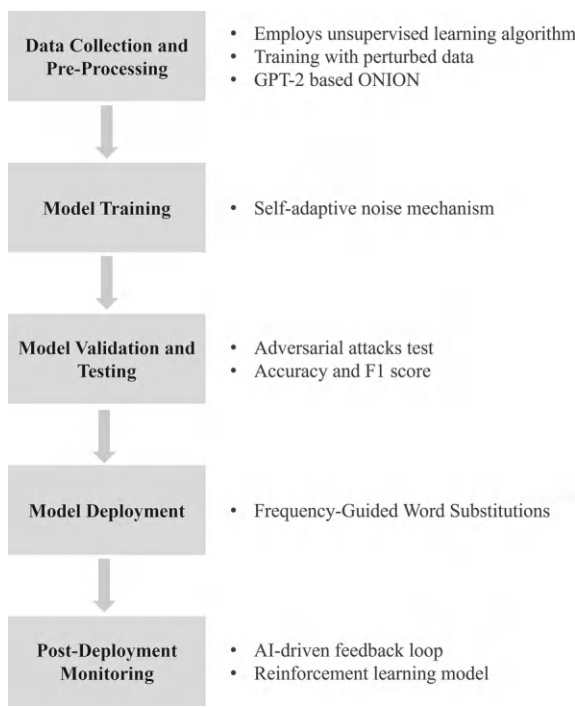


Fig. 7 Mechanism used in different phases

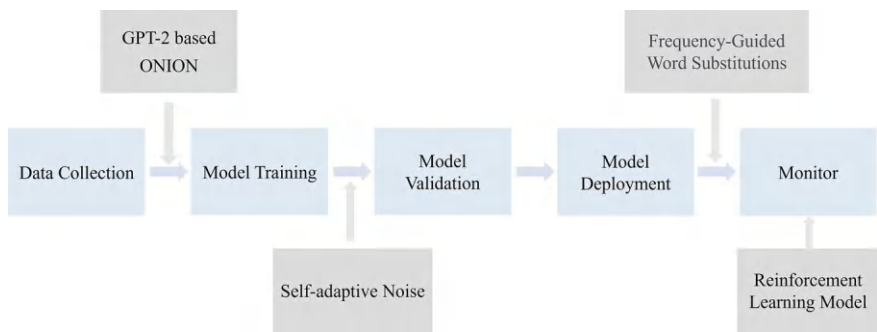


Fig. 8 Mechanism used in NLP lifecycle

5 Conclusion

In this research, we embarked on a journey to understand the vulnerabilities in modern NLP systems and to propose an AI-based defense framework aimed at addressing these challenges, our approach focuses on a multi-layered strategy that actively adjusts to new adversarial patterns. Through our conceptual defense mechanism, we

demonstrated the potential benefits of integrating unsupervised learning algorithms during the data collection and pre-processing phase, employing self-adaptive noise during model training, and leveraging techniques like FGWS during model deployment. Furthermore, the post-deployment phase is crucial in ensuring the continuous improvement and adaption of our system to evolving adversarial inputs. Our proposed framework provides a foundation upon which future research can expand, aiming for a more secure and resilient AI landscape in the realm of NLP. We believe that with continuous advancements in this area, we can remain a step ahead of adversaries and ensure the safe deployment of NLP technologies across various domains.

References

1. Terzopoulos G, Satratzemi M (2020) Voice assistants and smart speakers in everyday life and in education. *Informatics In Education*. 19:473–490
2. Wu C, Li X, Guo Y, Wang J, Ren Z, Wang M, Yang Z (2022) Natural language processing for smart construction: current status and future directions. *Autom Construct* 134:104059. <https://www.sciencedirect.com/science/article/pii/S0926580521005100>
3. Nadkarni P, Ohno-Machado L, Chapman W (2011) Natural language processing: an introduction. *J Am Med Inform Assoc* 18:544–551
4. Kang Y, Cai Z, Tan C, Huang Q, Liu H (2020) Natural language processing (NLP) in management research: a literature review. *J Manag Anal* 7:139–172
5. Boucher N, Shumailov I, Anderson R, Papernot N (2022) Bad characters: imperceptible NLP attacks. In: 2022 IEEE Symposium On Security and Privacy (SP), pp 1987–2004
6. Morris J, Lifland E, Yoo J, Grigsby J, Jin D, Qi Y (2020) Textattack: a framework for adversarial attacks, data augmentation, and adversarial training in nlp. *ArXiv Preprint* [ArXiv:2005.05909](https://arxiv.org/abs/2005.05909)
7. Zhou Y, Zheng X, Hsieh C, Chang K, Huang X (2020) Defense against adversarial attacks in nlp via dirichlet neighborhood ensemble. *ArXiv Preprint* [ArXiv:2006.11627](https://arxiv.org/abs/2006.11627)
8. Wallace E, Stern M, Song D (2020) Imitation attacks and defenses for black-box machine translation systems. *CoRR*. [arXiv:abs/2004.15015](https://arxiv.org/abs/2004.15015)
9. Juuti M, Szyller S, Marchal S, Asokan N (2019) PRADA: protecting against DNN model stealing attacks. In: 2019 IEEE European Symposium On Security And Privacy(EuroS&P), pp 512–527
10. Gong X, Wang Q, Chen Y, Yang W, Jiang X (2020) Model extraction attacks and defenses on cloud-based machine learning models. *IEEE Commun Mag* 58:83–89
11. He X, Lyu L, Sun L, Xu Q (2021) Model extraction and adversarial transferability, your BERT is vulnerable!. In: Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: human language technologies, pp 2006–2012. <https://aclanthology.org/2021.naacl-main.161>
12. Goyal S, Doddapaneni S, Khapra M, Ravindran B (2023) A survey of adversarial defenses and robustness in nlp. *ACM Comput Surv* 55:1–39
13. Chen X, Salem A, Chen D, Backes M, Ma S, Shen Q, Wu Z, Zhang Y (2021) BadNL: backdoor attacks against NLP models with semantic-preserving improvements. In: Annual computer security applications conference, pp 554–569. <https://doi.org/10.1145/3485832.3485837>
14. Chen X, Salem A, Chen D, Backes M, Ma S, Shen Q, Wu Z, Zhang Y (2021) Badnl: backdoor attacks against nlp models with semantic-preserving improvements. In: Annual computer security applications conference, pp 554–569
15. Qi F, Chen Y, Li M, Yao Y, Liu Z, Sun M (2021) ONION: a simple and effective defense against textual backdoor attacks. In: Proceedings of the 2021 conference on empirical methods in natural language processing, pp 9558–9566. <https://aclanthology.org/2021.emnlp-main.752>

16. Madry A, Makelov A, Schmidt L, Tsipras D, Vladu A (2019) Towards deep learning models resistant to adversarial attacks
17. Mozes M, Stenetorp P, Kleinberg B, Griffin L (2020) Frequency-guided word substitutions for detecting textual adversarial examples. CoRR. [arXiv:abs/2004.05887](https://arxiv.org/abs/2004.05887)

A New Scheme Modeling Gym Membership Transactions with NFT Systems



Hao-Yu Tsai, Jen-Chun Chang, and Shiuh-Jeng WANG

Abstract Fitness has become a popular trend, driven by online media. The appeal of a strong and attractive physique is undeniable. With the growing interest in fitness and easy access to fitness information, the demand for fitness is rising. In order to attract longer-term subscriptions, many large chain fitness clubs typically employ a strategy where the longer the membership contract, the lower the average monthly fee. However, this could potentially result in problems when individuals wish to cancel or modify their plans later. Because of the high cancellation fees, some individuals choose to transfer or sell their memberships on social media platforms like Facebook, PTT, and Dcard (Taiwan internet forum). However, such transactions often lead to disputes and inconvenience. As a result, this paper aims to improve the process of second-hand transactions in fitness clubs. We offer memberships in the format of NFTs, utilizing their uniqueness and Tamper-evident to guarantee a precise membership transfer process. Furthermore, this resolves the persistent problems of mistrust and ambiguity in the second-hand transactions market.

Keywords Non-Fungible token · Blockchain · Gym · Transaction · Smart Contract · Ethereum

1 Introduction

The origin of the prototype for the second-hand market can be traced back to flea markets [1], which tend to focus on sustainable consumption. Individuals offer their used collectibles and antiques for trade. However, because of their low demand in the past, the subject of the Second-Hand market has received little attention from economists. Nevertheless, this has recently changed [2]. Due to inflation and

H.-Y. Tsai · J.-C. Chang
National Taipei University, New Taipei, Taiwan

S.-J. WANG (✉)
Central Police University, Taoyuan, Taiwan
e-mail: dopwang@gmail.com

changing consumer habits, numerous high-priced electronic products and luxury goods have entered the market. A considerable number of consumers cannot afford the elevated prices of new items and are therefore looking for used or second-hand products instead. At the same time, these consumption habits will also mean that first-hand buyers will benefit from this new and huge second-hand market and will be more willing to buy products as they do not have to worry about reselling the products.

The reselling of electronic products is similar to expiration dates. Electronic products have a finite lifespan as their internal components degrade over time, and expiry dates share similar characteristics to those of an expiration date [3]. This category of goods is easily evaluated and its remaining lifespan is easy to assess, which makes it easier to trade on the secondary market [4]. In recent years, a new technology known as NFTs has emerged. These can potentially resolve the problems that exist in the second-hand market during transactions.

NFT (Non-Fungible Token) [5] is a term that has gained popularity in recent years, it is an acronym for Non-Fungible Token. They are called Non-Fungible Tokens because each of them has unique characteristics and data. Currently, NFTs have brought significant changes to various fields. For example, in the digital arts, artists can convert their digital creations into NFTs and sell them, ensuring ownership and authenticity while generating revenue. Similarly, in music, NFTs simplify music ownership and copyright management [6], allowing musicians to sell their works. In addition, NFTs represent collectibles such as baseball cards or unique virtual items [7], expanding the market for collectibles and providing new opportunities for collectors.

Blockchain [8] is a decentralized data storage and management technology. Its basic concept is to store data on multiple computer nodes instead of concentrating it on a single server or centralized organization. This data is stored in an interconnected chain, called a blockchain, in the form of blocks, with each block containing a set of transactions or data. A key feature of blockchain technology is its immutability. Once NFTs are recorded on a blockchain, they are virtually impossible to tamper with. This ensures that the ownership and transaction history of NFTs remain transparent and trustworthy, guaranteeing the value and reliability of NFTs.

Blockchain's immutability can solve the problem of information opacity, ensuring the recording of all transaction histories after membership cards are established. With the rise of the metaverse and its rapid development, NFT trading platforms have matured significantly. This offers convenience for establishing our membership cards and potentially replacing traditional physical membership cards. Therefore, this article takes fitness clubs as an example, aiming to rapidly create digital membership cards through NFTs. It also intends to utilize blockchain's immutability and traceability to record the transaction history of these membership cards. Currently, there isn't a trustworthy or efficient platform in Taiwan for transferring fitness club membership cards. People often sell their membership cards through acquaintances or on social media, leading to transaction disagreements. Our goal is to address this issue with a fitting platform for exchanging membership cards within fitness clubs, utilizing NFT technology.

The rest of this article is organized in the following way. In Sect. 2, we introduce our background knowledge. The proposed scheme in detail is introduced in Sect. 3. Discussions are presented in Sect. 4. Finally, Sect. 5, serves to conclude the whole paper.

2 Background

In this section, we'll take a closer look at NFT, Smart Contracts, IPFS, Ethereum, and the traditional transaction model. We will explore these areas further to help you better understand their relationship.

2.1 NFT

When we talk about NFTs, we're essentially referring to a unique type of token [9] NFT stands for Non-Fungible Token. Let's break this term down into two components: 'fungible' and 'token'. In this context, 'token' means a digital representation of an asset, similar to a currency. Fungible, on the other hand, refers to the concept of interchangeable assets with identical structures and values. For example, coins and banknotes within a country are typically standardized, meaning that a coin of a certain denomination has the same value in different regions and can be freely exchanged. When we use the term "non-fungible," we are describing an asset or token with unique characteristics [10] that cannot be exchanged like standardized assets. The key characteristics of non-fungibility include.

Uniqueness: Non-fungible tokens are unique, each having its own unique attributes or characteristics that distinguish it from other assets. This uniqueness can take the form of an artist's painting, a musician's audio-visual work, etc.

Non-substitutability: Due to the uniqueness of non-fungible tokens, they cannot be substituted or exchanged for the same value. In other words, a non-fungible token cannot be replaced by another non-fungible token of the same value.

Smart Contract Support: Most non-fungible tokens are built on smart contracts based on blockchain technology. These smart contracts record the uniqueness, ownership, and other information about each non-fungible token.

Sources of Value: The value of non-fungible tokens typically comes from their uniqueness, scarcity, and market demand. People are willing to pay a premium to own these unique assets.

2.2 *Smart Contract*

Smart Contract was first proposed by Nick Szabo in the 1990s. The basic idea behind it is that many types of contractual terms can be embedded in the hardware and software we deal with [11]. A smart contract is explicitly defined as a set of digitally defined promises. It is a type of contract in the form of program code. This code is designed to automatically enforce the rules and conditions of the contract without human intervention or intermediary assistance. These contracts are encoded in the blockchain and automatically execute actions when certain conditions are met. These transactions are both traceable and immutable [12]. As a result, it offers greater security compared to conventional contracts and lowers other transaction expenses related to contracts. More importantly, smart contracts can establish trust between parties in a situation where trust cannot be assumed. Thus, they have the potential to revolutionize traditional business practices.

2.3 *IPFS*

IPFS is the abbreviation of interplanetary file system, it is a distributed data storage protocol [13]. The goal of IPFS is to create a more open, decentralized, and secure Internet infrastructure. Its core concept involves breaking files into smaller pieces and storing them in a decentralized manner across multiple nodes, as opposed to centralized servers. This enhances data availability and security, and reduces the risk of a single point of failure. Unlike location-based file addressing in standard file systems, IPFS uses content-based addressing. When a file is added to IPFS, it generates a unique hash value that is calculated based on the contents of the file. This hash value is a guarantee of the file's security, since any change to the file's contents will result in a change to the hash value [14]. Think of the hash value as an identity card for the file. Each file has a unique hash value computed from its contents. Even a minor change, like a single punctuation mark within the file, will produce a completely different hash value. As we mentioned earlier, similar to how each person has a unique ID card, a file corresponding to a hash value cannot be changed. The goal of IPFS is to overcome the dominating single-server model prevalent in most services today. When combined with blockchain technology, it endeavors to manage vast amounts of data in a secure, distributed, and persistent manner.

2.4 *Ethereum*

Ethereum is a blockchain-enabled decentralized open-source platform for decentralized systems [15]. It provides a programmable environment. Developers can use it to create smart contracts and decentralized applications that can run on Ethereum.

Ethereum is currently the largest blockchain platform that supports smart contracts [16]. A smart contract is a code on Ethereum that automatically executes and enforces the terms and conditions of a contract. These contracts are written into the blockchain as code. This code automates transactions and contract execution without the need for an intermediary. EtherScan is a popular blockchain browser for Ethereum. It allows users to query and view transactions, addresses, smart contracts, and other data on the Ethereum blockchain. This is useful for monitoring smart contract operations and transactions.

2.5 Traditional Transaction Model

Next, we divide the traditional transaction model into two parts, first-hand and second-hand transactions. We will study these two transaction models to have a more comprehensive understanding of their operation mechanisms and characteristics. In this way, we can compare the similarities and differences between them and our proposed scheme.

a. First-hand transactions.

- *Purchase Request*: The buyer initiates a request to purchase from the seller.
- *Seller Response*: The seller receives the buyer's request and replies to the buyer's actions.
- *Accepted Demand*: If the seller accepts the buyer's demand, the transaction will proceed.
- *Rejected Request*: If the seller rejects the buyer's request, the transaction has failed.
- *Transaction Proceeds*: If the seller accepts the demand, the transaction will proceed. Subsequently, the buyer is usually obligated to pay for the purchased item.
- *Payment*: The buyer pays for the purchased item(s).

b. Second-hand transactions.

- *Buyers look for items*: Buyers initially search for pre-owned goods on various platforms including online marketplaces, social media, flea markets, thrift stores, and more. The buyer is responsible for paying for the item(s) they purchased.
- *Contacting the Seller*: Once a buyer locates what they need, they will get in touch with the seller.
- *Inspection of the item*: The buyer has the right to inspect the condition of the item at a specific time and place during the transaction. They can ensure that the item is in the condition described.
- *Payment*: Upon satisfaction with the item, the buyer will pay the previously agreed upon price.

- *Confirmation of Transaction Completion:* Once the item is transferred to the buyer, the transaction is complete. Both parties can rely on the success of the transaction.

It’s worth noting that in second-hand transactions, *inspection of the item* is a stage prone to disputes. Buyers struggle to obtain transaction histories, authenticity, and information about previous owners. This lack of transparency can lead to mistrust between the parties. Therefore, our proposed scheme aims to solve this problem by leveraging NFT technology. Converting memberships into NFTs guarantees the recording of transaction history and product attributes. This reduces disagreements, and reduces the risk of transactions.

3 Proposed Scheme

In this paper, we proposed an NFT-based solution for gym membership transactions, supporting buying, selling, and bidding. Figure 1 provides a high-level overview of our proposed NFT-based solution.

We will describe our process for transaction modeling in three sections: the first for primary transactions, the second for secondary transactions, and the third for competitive bidding. We will provide detailed explanations for each section.

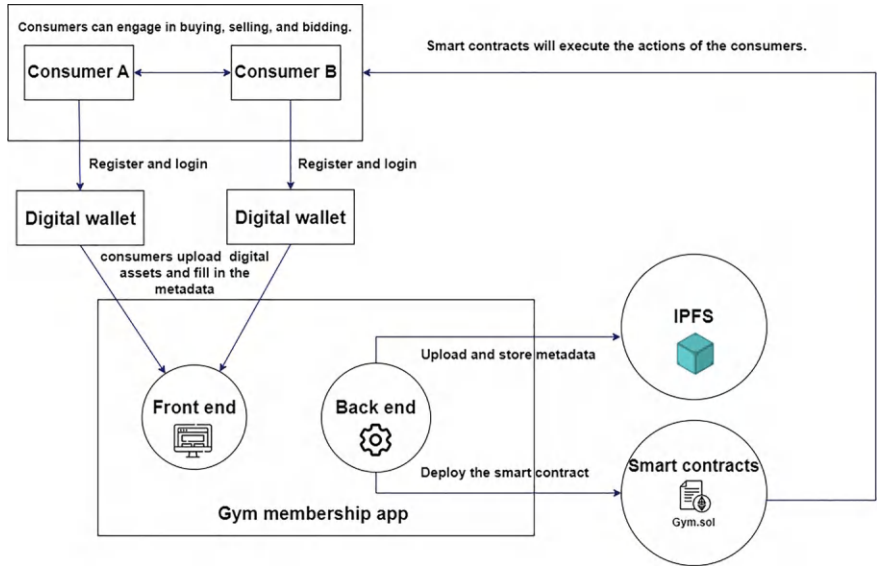
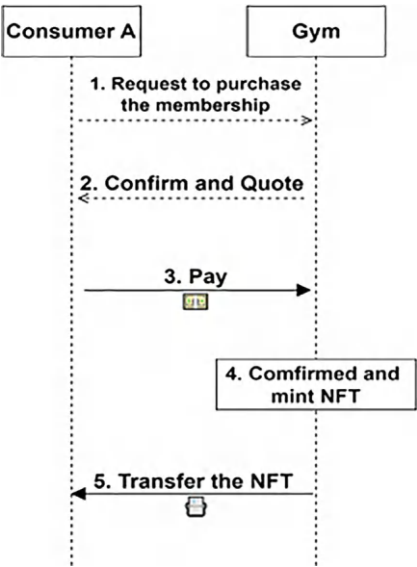


Fig. 1 The architecture of the proposed scheme

Fig. 2 First-hand transaction



3.1 First-Hand Transactions

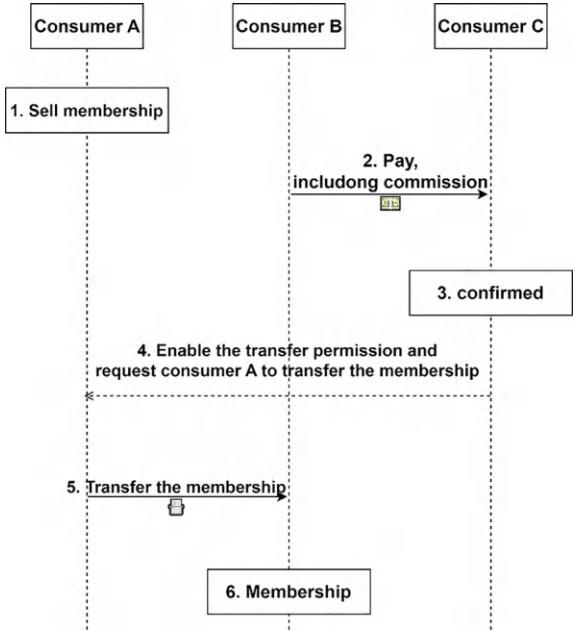
It describes a direct transaction process involving participants and the transaction subject. Taking consumer A and the gym as an example, the initial transaction process follows a fixed-price approach. In this method, the gym informs the consumer about the price of the desired session. Figure 2 illustrates the flowchart of this primary transaction, using dashed arrows for information exchange and solid arrows for the flow of cash or physical items.

- Transaction Process:
1. Consumer A submits a purchase request (which includes information about the duration of the membership period).
 2. Gym Owner Reply Confirm Purchase Request and Quote
 3. Consumer A pays based on the quote
 4. Gym Owner Receive payment and submit (or issue) NFTs
 5. Consumer A receives the NFT and is eligible for membership accordingly.

3.2 Second-Hand Transactions

A second-hand transaction for a gym membership is when a gym member transfers their membership to another person rather than purchasing a new membership directly from the gym. In the following, we will use the example of Consumer A (the seller), Consumer B (the buyer), and the gym owner. Our transaction process uses a competitive bidding system. In this system, each buyer competes for the used

Fig. 3 Second-hand transaction



memberships by offering a price. The buyer with the highest price wins the right to trade. In this case, we assume that Consumer B is the highest bidder. Figure 3 shows the flow of the second-hand transaction after a successful bidding process. The bidding process is described in Sect. 3.3.

- Transaction Process:
1. Consumer A submits a request for the sale of a membership (which includes information about the validity period of the membership to be purchased) and submits a reserve price and a closing time.
 2. The trading platform initiates the sale of a membership by posting a request. All potential buyers are free to place bids during a specified time period. At the end of the bidding period, the highest bidder, known as the winner, has priority to proceed with the purchase.
 3. Consumer B pays the gym owner according to the bid price.
 4. Gym owner tells consumer A to transfer their membership.
 5. Consumer A transfers his membership to Consumer B.
 6. Consumer B receives the NFT and becomes a member.

3.3 Bidding

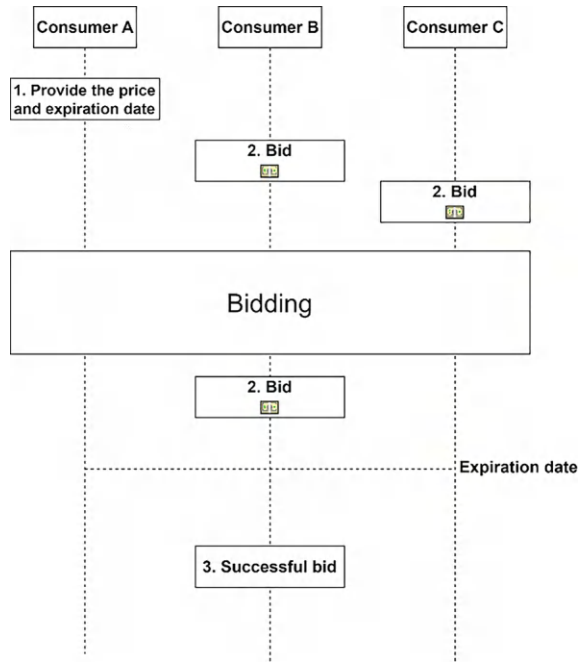
This refers to the opportunity for multiple potential buyers to bid aggressively for a specific item or service. During the bidding process, they submit different prices in

the hope that they will eventually be successful in purchasing the goods or services they need.

In the following, we will use the example of Consumer A (the seller), Consumer B, and Consumer C (both are buyers, i.e., bidders). Although the number of bidders can be more than two or more, the bidding process is the same regardless of the number of participants. Therefore, we will use Consumers B and C as an example to simplify the bidding model. Figure 4 is a flowchart of the bidding process. Without loss of generality, let's assume that Consumer B is successful in the bidding process and wins the first right to purchase the used memberships.

- Transaction Process:
1. Consumer A submits a request for the sale of a membership (which includes information about the validity period of the membership to be purchased) and submits a reserve price and closing time.
 2. Consumer B and Consumer C can submit bids and adjust their prices until the expiration date.
 3. Until the expiration date, the highest bidder wins (in this example, Consumer B).

Fig. 4 Bidding process



4 Discussion

In this section, we will discuss the distinctions and the features between traditional transactions and NFT transactions under our proposed scheme.

Traditional membership card transactions can be broadly categorized as physical and online platforms. However, whether the transaction occurs in physical or online realms, a step involving the verification of the transaction object is required. This provides the buyer with assurance and facilitates the examination of the seller's goods, enabling the buyer to present a reasonable bid for the transaction. Nonetheless, physical transactions are constrained by the limited time available for negotiation between the buyer and seller, which can lead to hasty pre-transaction inspections. What's worse is even when conducting transactions through online platforms, another issue that you may still encounter: fake products. Illicit trade in fake goods is a significant and growing threat in a globalized and innovation-driven economy [17]. Information asymmetry between transaction parties leads to mutual distrust, making it difficult to reach negotiations. Long-term and widespread occurrences of this circumstance can offend the health of the market.

We propose a scheme that replaces traditional gym memberships with NFTs. Both the traditional model and our scheme involve interactions and negotiations between buyers and sellers to reach a consensus and complete the transaction. To achieve this goal, the primary step is to eliminate information asymmetry between buyers and sellers. The NFT transactions can effectively tackle the previously mentioned issues. Next, we will examine the similarities and contrasts between traditional trading models and our proposed scheme. The following list will detail them to convey comprehensive and accurate product information.

- **Uniqueness:** One challenge in traditional transactions is the difficulty in confirming the uniqueness of second-hand goods. This concerns the uncertainty in the market for buying or selling second-hand items, where it can be difficult to determine whether the product is counterfeit or replicated.

Each NFT membership has a unique token ID, which is determined by its hash value and cannot be copied or imitated. This feature greatly enhances the security of gym memberships by preventing fraudulent or illegal replication.

- **Traceability:** In traditional transactions, individuals are unable to access precise information regarding a product's usage history and must depend solely on the seller's verbal account to comprehend the product's previous experiences. This greatly restricts individuals' accurate knowledge of the product's background.

All details of NFT transactions are recorded comprehensively on the blockchain, including operations such as the creation, transfer, and renewal of gym memberships. This complete record makes the history of the membership easy to track and query. This is very beneficial for buyers as they can trace the history of past gym memberships.

- **Transparency:** In traditional transactions, there is often a lack of transparent information about the product available for reference. This means that in traditional transactions, people cannot easily access clear and transparent information about the product.

Blockchain technology itself provides a high degree of transparency. It allows all transactions and operations to be viewed, ensuring fairness and transparency in transactions. Buyers can view the status of their gym memberships at any time, which greatly enhances the credibility of the transaction. There is no longer a need to rely on a third-party organization to verify the authenticity of a transaction.

- **Tamper-evident:** In traditional transactions, there is a risk of tampering or concealment of the quality, condition, or historical information of the product by dishonest sellers or intermediaries. This means that during the transaction process, it may lead to buyers receiving products that do not meet their expectations and potentially result in losses or risks.

The tamper-evident nature of NFT transactions ensures the security of gym memberships. Once gym membership data are registered on the blockchain, it cannot be tampered with or deleted at will. This effectively prevents fraud and brings a higher level of confidence and willingness to transact between buyers and sellers.

Table 1 shows the comparison between the traditional transaction model and the NFT transaction model.

The establishment of a third-party trading platform allows gyms to make transfers in a more convenient manner.

- This flexibility in timing benefits both buyers and sellers in the membership transaction process.
- Membership holders can easily find the next buyer when they decide to sell their membership.
- It reduces the amount of time that second and subsequent buyers have to make their choices.
- The gym manufacturer can utilize the three strategies mentioned above, which provide advantages over the consumer. They can use these strategies to customize the gym membership contract and promote it as a selling point.

Table 1 Traditional transaction model versus NFT transaction model

Property	Traditional transaction model	NFT transaction model
Uniqueness	Risk of Counterfeiting and Replication	Hash value ensures its authenticity
Traceability	The previous transaction history is unknown	Blockchain-based detailed records
Transparency	Only based on the seller’s statement	Past transactions are visible
Tamper-evident	Possibly susceptible to tampering	Transactions on blockchain are immutable

- The gym itself may receive a percentage of the commission for second-hand transactions.
- Solve the long-term chaos of gym membership transactions, which in the past were typically exchanged on major community software platforms such as Facebook, PTT, and Dcard.

5 Conclusion

In this paper, we have developed an innovative scheme that addresses traditional transaction shortcomings, including negotiation inefficiencies and bidding risks. Our system enables decentralized data management and utilizes blockchain technology to enhance transaction security. Moreover, we implement NFTs to convert membership cards into digital form. This significantly reduces fraud due to their immutability and uniqueness. In the evolving digital trading market, which now includes NFTs as a commodity platform, our system can effectively reduce component costs and contract setup time. Furthermore, if our study proves successful in the fitness industry, other markets facing similar challenges may benefit from our findings to address their concerns.

Acknowledgements This research is supported by National Science and Technology Council, Taiwan under contract 110-2221-E-305-005-MY2

References

1. LaFarge A (2000) Introduction. U.S. flea market directory, 3rd Edition: a guide to the best flea markets in all 50 states. US flea market directory. St. Martin's Press. ISBN 978-0-312-26405-5. Retrieved July 23, 2019
2. Fox AH (1957) A theory of second-hand markets. *Econ* 24(94):99–115
3. Chu LY (2016) An empirical study on purchase intention of second-hand iPhone: A second-hand iPhone would be better
4. Cantista I, Delille D, Sousa A (2022) Second-Hand history and algorithms: the case of sellpy. In: Cantista, I., Delille, D. (eds) *fashion heritage*. Palgrave Macmillan, Cham. (2022)
5. Wang Q, Li R, Wang Q, Chen Sh (2021) Non-Fungible Token (NFT): overview, evaluation, opportunities and challenges
6. Tharun T, Vamshi A, Eswari R (2023) NFT application for music industry using Blockchain Smart contracts In: 2023 4th International Conference on Innovative Trends in Information Technology (ICITIIT), Kottayam, India, pp 1–6
7. Rehman W, Zainab HE, Imran J, Bawany NZ (2021) NFTs: applications and challenges. In: 2021 22nd International Arab Conference on Information Technology (ACIT), Muscat, Oman, pp 1–7
8. Nakamoto S (2008) Bitcoin: a peer-to-peer electronic cash system
9. Entriken W, Shirley D, Evans J, Sachs N (2018) ERC-721: Non-Fungible token standard. ethereum improvement proposals, 721, Available: <https://eips.ethereum.org/EIPS/eip-721>

10. Seyed M, Hosseini B, Nasim N, Omid B, Qiang Q (2022) Patents and intellectual property assets as non-fungible tokens; key technologies and challenges. *Sci Rep*12, Article number: 2178
11. Szabo, Nick (1997) Smart contracts: formalizing and securing relationships on public networks. *First Monday*, 2(9–1)
12. Pise R, Patil S (2022) A Deep Dive into Blockchain-based Smart Contract-specific Security Vulnerabilities. In: 2022 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS), Pune, India, pp 1–6
13. Zheng Q, Li Y, Chen P, Dong X (2018) An innovative IPFS-Based storage model for blockchain. In: 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI), Santiago, Chile, pp 704–708
14. Guidi B, Michienzi A, Ricci L (2021) Data persistence in decentralized social applications, The IPFS approach. In: 2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC). Las Vegas, NV, USA, pp 1–4
15. Toyoda K, Machi K, Ohtake Y, Zhang AN (2020) Function-level bottleneck analysis of private proof-of-authority ethereum blockchain. *IEEE Access* 8:141611–141621
16. Huang T, Lin D, Wu J (2022) Ethereum account classification based on graph convolutional network. In *IEEE Trans Circuits Syst II: Express Briefs*, 69(5), pp 2528–2532
17. OECD/EUIPO, Misuse of E-Commerce for trade in counterfeits, illicit trade, OECD publishing, Paris. <https://doi.org/10.1787/1c04a64e-en>

A Text to Human-Like Speech Using Tacotron-Based TTS Model



Wei-Chen Wu, Zhi-Xun Cai, and Chin-Feng Lee

Abstract Tacotron2 [1] is a speech synthesis model that can convert text into Mel Spectrograms [1] and generate natural-sounding speech waveforms, sound close to the human voice. We hope that this technology can provide a valuable resource for those who struggle with public speaking or are temporarily unable to speak. This study uses WaveGlow [2] to convert Mel Spectrograms [1] into audible sound waves, read the Japanese text, and output the speech with Japanese voice, actor Saori Hayami as the voice actor. We chose to use real human voice actors to train the speech system to improve the naturalness and clarity of Japanese synthesized speech, and to make a clear distinction from AI voices such as Google Assistant and Apple Siri.

Keywords Attention mechanism · Mel Spectrograms [1] · Japanese language · Tacotron2 [1] · WaveGlow [1]

1 Introduction

Have you ever had the experience of practicing many times before going on stage to give a speech and everything goes well, but as soon as you get on stage you get so nervous that you forget your words?

The study developed a scheme that could avoid this embarrassment, so we looked at the Text to Speech(TTS) model, where an AI is trained to speak in a specific voice.

W.-C. Wu

Department of Finance, National Taipei University of Business, Taipei, Taiwan
e-mail: weichen@ntub.edu.tw

Z.-X. Cai

Department of Information Management, National Taipei University of Business, Taipei, Taiwan
e-mail: 11136029@ntub.edu.tw

C.-F. Lee (✉)

Department of Information Management, Chaoyang University of Technology, Taichung 41349, Taiwan
e-mail: lcf@cyut.edu.tw

After training, the AI will be able to pronounce words in a specific voice, and if the voice is the same as the speaker's voice on stage, the embarrassment of the speaker forgetting his or her words can be avoided. We hope to utilize this scheme to benefit those who feel nervous when speaking on stage.

In addition, we also aim to utilize TTS technology to develop a Japanese reading assistance tool to help visually impaired individuals access information, emails, social media, etc., in audio format. This will enable them to participate more conveniently in social activities and stay connected with their family, friends, and colleagues, offering greater autonomy and convenience for visually impaired individuals.

Since the introduction of the WaveNet [3] algorithm, various approaches have been proposed to develop more efficient speech synthesis models that can run on slower CPU computers and embedded systems like mobile phones. Traditional speech synthesis methods were complex and relied on extracting linguistic features, which was time-consuming, labor-intensive, and required domain knowledge. Moreover, these methods often resulted in synthesized speech with glitches and pronunciation issues, making it sound unnatural.

However, advancements in neural network technologies have led to the development of end-to-end speech synthesis models like Tacotron1 and Tacotron2 [1], this study Tacotron2 [1] uses and refers to it as “Tacotron” throughout the document. These neural network-based models eliminate the need for manual feature extraction by utilizing neural networks. The generated audio is further improved by passing it through a vocoder based on the Griffin Lim algorithm, resulting in noticeably enhanced audio quality compared to previous methods. However, these models still suffer from long training times and high computational requirements.

In recent years, many devices have provided voice for input and output, such as Siri, etc. This kind of voice dialog makes human-machine interaction more intuitive, and after the device executes the user's commands, it will feedback on the prompts of completing the commands to the user through voice, and the system will generate the text content of the response in advance, and then through the TTS technology to convert the text into speech for output. TTS technology can also be used in a variety of applications.

2 Related Work

In the application of artificial intelligence, there are a lot of things that are just ideas or concepts, but as computer hardware becomes more and more powerful, the original concepts are gradually realized, and the application of speech synthesis system is one of them.

We will introduce the technology and model used in this study. The operation process of Tacotron [1] can be briefly described as follows:

First, the text sequence is taken as input and converted into a fixed-dimensional vector sequence through a character embedding layer. Then using a recurrent

sequence-to-sequence architecture, the character embedding vector sequence is taken as input to predict a series of Mel Spectrogram [1] frames. In this process, the attention mechanism is used to focus the model's attention on different parts of the input sequence to better predict the Mel Spectrogram [1] frames. Using the predicted Mel Spectrogram [1] frames as conditional input, the WaveNet [3] model generates time-domain waveform samples. WaveNet [3] is a deep convolutional neural network that can generate high-quality audio waveforms. Finally, the generated time-domain waveform samples are converted into audio signals, ultimately producing natural speech synthesis. Overall, the design of Tacotron [1] aims to achieve natural speech synthesis and allows the model to better learn the features of speech by using Mel Spectrogram [1] as conditional input for WaveNet [3].

The pre-training model is trained using a large amount of data on Kaggle [4]. And then using the pre-trained model, Tacotron [1] is responsible for converting text into a sequence of speech features, also known as Mel Spectrogram [1], to fine-tune the model. Then WaveGlow [2] is used to convert the Mel Spectrogram [1] generated by Tacotron [1] into the final audible waveform. We use the pre-trained model to fine-tune the model, trying to make the fine-tuned model match the sample used in the fine-tuning.

In the early days of TTS technology, the capabilities were limited, and the cost of manually recording voices was high. The synthesized voices sounded more robotic since they were generated using signal processing techniques, resulting in a monotone speech with limited intonation. This study aims to achieve a smoother and more natural speech output by combining Mel Spectrogram [1] and Romanization. This approach helps enhance the prosody and timbre of the synthesized speech, making it sound closer to real human speech. Additionally, incorporating Romanization provides clearer pronunciation guidance, which aids listeners in better understanding and receiving the information.

3 Our Model

3.1 Dataset

We use a pre-trained model from the Nvidia [2] official as the most important object for our fine-tuning.

3.2 Training Process

First of all, we download the pre-training model provided by Nvidia [2], and then use our dataset (Hayami) to do fine-tuning training. Before training, we need to convert the audio file into a sound waveform file (wav), adjust the subtype and sample rate

Table 1 Hyperparameters

Hyperparameters	Value
Learning_rate	1e-5
Weight_decay	1e-6
Grad clip thresh	1.0
Batch size	4
Mask padding	true

to 16bit and 22050 Hz respectively, and then integrate each verbatim text into a single file, and then use line breaks to distinguish each audio file, and clearly label the location and content of the audio file. In the text part, we need to integrate each verbatim script together, and then use line breaks to distinguish each audio file, and clearly label the location and content of the audio file. During the training process, we need to constantly adjust the batch size to avoid the waveform file being too large and easily overloading the memory in the GPU.

3.3 *Hyperparameter*

See Table 1.

3.4 *Audio Parameters*

- self.max_wav_value: The maximum value of the audio waveform. It is used to normalize the range of the audio waveform to $[-1, 1]$ for better handling of audio data.
- self.sampling_rate: The sampling rate of the audio waveform, i.e., the number of samples per second. It is an important parameter in audio data.
- self.filter_length: A scalar value representing the window size used in Short-Time Fourier Transform (STFT). STFT is a method used to convert time-domain signals to frequency-domain signals.
- self.hop_length: A scalar value representing the step size between adjacent windows in STFT. It controls the overlap in STFT and is usually set to half of the window size.
- self.win_length: A scalar value representing the actual size of the STFT window. It is usually equal to filter_length but can be adjusted as needed.
- self.n_mel_channels: A scalar value representing the number of channels in the Mel Spectrogram [1]. Mel Spectrogram [1] is a frequency representation that evenly divides the audio frequency.
- self.mel_fmin: The minimum frequency of the Mel Spectrogram [1]. It is typically set to 0.0.

Table 2 Audio parameters

Audio parameters	Value
Max_wav_value	32,768
Sampling_rate	22,050
Filter_length	1024
Hop_length	256
Win_length	1024
N_mel_channels	80
Mel_fmin	0
Mel_fmax	8000

- self.mel_fmax: The maximum frequency of the Mel Spectrogram [1]. It is typically set to 8000.0 (Table 2).

3.5 Model Architecture

We propose a novel speech synthesis system. This system is a fully neural network approach that combines the advantages of previous approaches. Our system uses a sequence-to-sequence Tacotron-style model, which generates Mel Spectral frames, and then uses a modified WaveNet [3] vocoder to generate time-domain waveform samples.

Our system differs from Deep Voice 3 [6] and Char2Wav [5] in that its naturalness has been shown to be comparable to human speech. Our system also uses a different intermediate representation, as we use a Mel Spectrum map, while Char2Wav [5] uses traditional vocoder features.

The first component is a cyclic sequence-to-sequence (Fig. 1). Feature prediction network that uses an attention mechanism to predict a series of Mel Spectral frames from an input sequence of characters. The second component is a modified version of WaveNet [3] that generates time-domain waveform samples based on the predicted Mel Spectral frames.

To bridge these two components, we chose a low-level acoustic representation: the Mel Spectrogram [1]. This representation can be easily computed from the time-domain waveforms and is smoother than the waveform samples. The use of squared error loss is easier to train because it is phase invariant within each frame.

3.6 Dataset–Fine-Tuning

Because we don’t have such a large sample size, and we don’t have the skills to make the neural network work properly and efficiently, we train only one or a few of the layers based on the trained model, and freeze the parameters of the other layers.

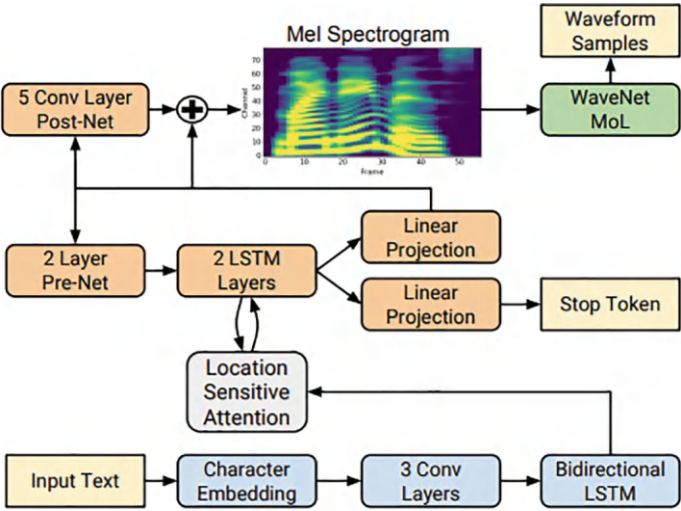


Fig. 1 Block diagram of the Tacotron2 [1] system architecture

We are training the AI using Japanese audio and verbatim scripts to create a TTS AI. We have chosen Saori Hayami as the subject of our project because she has provided voice acting for numerous anime. Since we aim to train our AI to support various scenarios, we have selected her as our AI learning target.

Among Saori Hayami’s numerous voice acting works, we have selected segments featuring her roles as Shiba Miyuki in “魔法科高校の劣等生”, Yor Forger in “Spy Family” and Nishimiya Shouko in “The Shape of Voice” for preprocessing.

4 Analysis

4.1 Training Results

Due to hardware and time limitations, we could only perform 200 fine-tuning iterations. After these 200 iterations, our trained model is now capable of utilizing Sawashiro’s voice to generate speech. However, there is still a significant amount of noise, and the generated speech lacks clarity. The following image shows the Mel Spectrogram [1] generated using the model (Fig. 2).

Further refinement and training beyond these 200 iterations would likely be necessary to reduce the noise and improve the clarity of the synthesized speech. While the model has made some progress, achieving a more accurate and clearer representation of human speech will require additional resources and training iterations. Nevertheless, with the constraints faced, we have achieved a basic level of speech synthesis using the trained model.

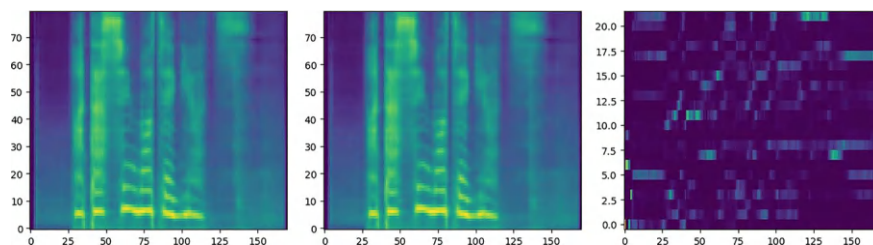


Fig. 2 The mel spectrogram [1] trained in this study

4.2 Viewing of Training Results

After performing 200 fine-tuning iterations, we utilized Nvidia's WaveGlow [2] model to convert the Mel Spectrogram [1] into an audio file and assess the training results.

5 Conclusion

Using Tacotron [1] for training in speech AI is very hardware-intensive. This is because the Mel Spectrogram [1], compared to text and numerical data, is a massive file that requires a significant amount of computation and storage space. However, as the number of training iterations increases, the model's accuracy improves significantly.

At the beginning of training, the model may only produce noise or vague sound representations. However, with increasing training iterations, the model gradually becomes capable of accurately rendering speech. For instance, after approximately 100 training iterations, the model may start generating some discernible but unclear sounds. By around 200 iterations, the model usually can synthesize speech that is more accurately recognizable by humans.

Of course, achieving near-human-level speech synthesis requires more time and iterations of training. Continuous training of the model helps in continuously optimizing its performance, but it also demands patience and time. While Tacotron [1] is a powerful model, achieving the best results still requires extensive training and fine-tuning. Given sufficient hardware resources and time, continuous training allows us to approach and approximate the quality of human-like speech synthesis.

Acknowledgements This research was supported by the National Science and Technology Council, Taiwan R.O.C., under contract number MOST 111-2221-E-324-019-MY2.

References

1. Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, RJ Skerry-Ryan, Rif A (2018) Saurous, Yannis Agiomyrgiannakis, and Yonghui Wu. Nat Tts Synth Cond Wavenet Mel Spectrogram Predict: Google, Inc., University of California, Berkeley
2. NVIDIA, “WaveGlow model for generating speech from Mel Spectrograms (generated by Tacotron2)”
3. Tamamori A, Hayashi T, Kobayashi K, Takeda K, Toda T (2017) Speaker-dependent WaveNet vocoder. In: Proc. Interspeech
4. Kyubyong Park (2018) Japanese single speaker speech dataset.
5. Jose Sotelo, Soroush Mehri, Kundan Kumar, Joao Felipe Santos, Kyle Kastner, Aaron Courville, Yoshua Bengio (2017) Char2Wav: End-to-end speech synthesis. In: Proc. ICLR
6. Wei Ping, Kainan Peng, Andrew Gibiansky, Serkan O Arik, Ajay Kannan, Sharan Narang (2018) Deep voice 3: 2000-Speaker Neural Text-To-Speech”: Baidu Research

A Novel Malware Classification Using CNN-SVM Deep Learning Method Based on Transfer Learning Architecture



Guan-Yu Wang, Hung-Jui Ko, and Wei-Jen Wang

Abstract With the frequent occurrence of global security incidents, the detection of malware has become a popular topic in the cyber security area. Deep learning techniques outperform conventional pattern-matching techniques in malware classification tasks. However, deep learning requires more time for model training. Therefore, we adopt transfer learning, where pretrained models are in a new model, which reduces the training time and generalization error. In this study, we propose a convolutional neural network (CNN)–support vector machine (SVM) deep learning method based on the transfer learning architecture for malware classification. VGG16, pretrained on ImageNet via transfer learning, is employed as the CNN model. We remove the fully connected layer from the VGG16 model to treat it as a feature extractor. Next, the Microsoft BIG 2015 dataset (10,868 malware samples of 9 families) is converted into grayscale images and input into the VGG16 model for feature extraction, thereby obtaining a new dataset. Finally, the new dataset is input into the SVM classifier for the classification task.

Keywords Transfer learning · VGG16 · SVM · Malware classification

1 Introduction

Since the beginning of the human era of epidemics in 2019, the percentage of workers who work remotely more than five days a week has nearly doubled according to the well-known statistics website Statista [1]. Consequently, the percentage of demands that rely on the Internet has increased. In addition, according to the World Economic

G.-Y. Wang · W.-J. Wang (✉)

Department of Computer Science and Information Engineering, National Central University,
Taoyuan, Taiwan
e-mail: wjwang@csie.ncu.edu.tw

H.-J. Ko

Changhua County Police Department, Address: No. 778, Sec. 2, Zhong-Zheng Rd.,
Changhua 500057, Taiwan

Forum's 2023 Global Risks Report [2], "cyberattacks on critical infrastructure" is ranked as one of the top three risks in the world in terms of potential impact. Thus, the increase in demand for Internet use is accompanied by cyberattacks. This shows that the phenomenon of cyberattacks associated with the increased demand for Internet use can no longer be ignored.

Malware can have severe effects on computers, such as the obstruction of normal computer and network operations, hardware failure, and data theft or loss. Moreover, when a malware attack occurs, it not only requires operators to spend a significant amount of time patching and repairing but may also result in a significant amount of money loss, which is extremely costly in terms of personnel and time.

Malware is software designed to steal data or damage a computer system. The first malware was called "Elk Cloner" in 1982, and the threat of malware still exists [3]. Common types of malware include worms, trojan viruses, spyware, adware, rootkit, keyloggers, ransomware, and cryptojacking [3]. Understanding the characteristics of different types of malware allows us to accurately determine the type of virus so that we can effectively respond to real attacks. Each malware has behavioral characteristics, which can be categorized. For example, a trojan virus appears as a normal program, but it opens a backdoor, allowing hackers to remotely access the target computer [4]. Another example is an adware virus, which launches programs or loads documents to cause computer lag and typically attacks with advertisements during browsing [4]. A previous study [5] has indicated that trojans and worms are among the top five malware types, accounting for approximately 58% and 6%, respectively. Except for trojans and worms, a previous study [6] has indicated that adware is the most common piece of mobile malware, accounting for nearly half of all mobile malware. Furthermore, some studies have discussed various methods for detecting and classifying these three types of malwares [7–9]. In this regard, this study mainly focuses on these three types of malware. We use the Windows malware dataset [10], which contains trojans, worms, and adware, as the benchmark.

A state-of-the-art malware detection system relies heavily on antivirus software, which typically has a signature-based database to detect malware patterns. However, one major problem with this approach is that it is impossible to detect the malware behavior pattern if it does not appear in the database. In addition, the most common methods for malware detection include static and dynamic analysis, e.g., the use of a static analysis method combined with a machine learning (ML) classifier [11]. In contrast, [12] uses dynamic analysis to extract features and train them using deep learning (DL) models. Nevertheless, both static and dynamic analyses have inherent limitations. Therefore, many studies [13–15] have explored malware detection on the basis of visualized images.

Many studies have focused on the use of convolutional neural networks (CNNs) for model training by converting malware into images and extracting features from these images. However, not all malware programs have corresponding image datasets or are easy to collect, and the number of malware programs that can be collected is typically extremely sparse, which may result in unbalanced or insufficient datasets. Therefore, transfer learning (TL) can be used to compensate for the shortage of

certain datasets by training the model of source datasets to be applied to the target dataset [16, 17].

Therefore, this study proposes a CNN–support vector machine (SVM) DL method based on the TL architecture.

The major contributions of this study are as follows:

1. A novel malware classification method based on TL is proposed. The model training time is significantly reduced using TL.
2. We modify the VGG16 model to obtain an improved feature extractor for malware classification.
3. CNN and SVM techniques are combined to obtain improved malware classification accuracy.

The rest of this paper is organized as follows. Section 2 describes related works. Section 3 introduces the methodology of the proposed model. Section 4 outlines the experimental results. Section 5 summarizes the conclusion, followed by references.

2 Related Work

In the literature, [18] proposed a DL approach based on TL. The architecture uses a well-trained VGG16 pretrained model based on the ImageNet dataset, which can be directly introduced into the pretrained model with prepared data inputs. The weight and bias of the model are used for feature extraction and fine-tuning. This approach can solve tasks in which data collection is difficult and can significantly reduce the model training time because it does not need to retrain the entire model. In addition, the study evaluated the robustness of the model using real-world data. However, the method did not perform very well on the Microsoft BIG 2015 dataset in terms of categorization, with accuracy of only 93.19%. In [19], a DL classification model based on artificial neural networks (ANNs) was proposed, where data in the form of byte codes were converted into 1024×1 images, and these images were then subjected to bicubic interpolation regularization tests. The images were then input into the ANN model for feature extraction and finally classified using the SoftMax classifier. The obvious drawback of this model is that its accuracy is only approximately 90.07%. Explainable artificial intelligence models, as mentioned in [20], have grown in popularity in recent years, probably because ML technology is considered a black-box technique in that the user of an ML model does not know the algorithms used by the model or how decisions are made operationally and predictively, which can make applying ML in some more sensitive or important domains impractical. Therefore, this study proposes an explainable ML classification model based on the LIME explainable technique and Secml. The model uses a decision tree classifier. The predicted accuracy is approximately 94%, and the greatest challenge of this method is that it requires feature extraction, which necessitates a certain degree of domain knowledge.

3 Methodology

3.1 VGG16 and Support Vector Machine

VGG16 is a CNN architecture proposed by a team from the University of Oxford, UK, in 2014 [21] and gained fame by taking the first position in the ImageNet competition of ILSVRC-2014 in 2014. The main contribution is the continuation of the earlier AlexNet architecture with additional hidden layers.

SVM is an ML algorithm for supervised learning that can be used for classification and regression analysis [22]. SVM was first used in binary classification. Its principle is to use a two-dimensional plane (hyperplane) to divide the data into two in three-dimensional space. The hyperplane that divides the data into two is called a decision boundary. The points on either side of the plane that are closest to the plane are called support vectors, and the distance between each support vector and the decision boundary is called the margin. The goal of SVM is to determine the margin of a plane that can be maximized. The larger the margin, the more completely the plane cuts the data. However, the application of classification is not only binary classification, but also includes multivariate classification, so the multivariate classification solution of SVM can be divided into two types of multivariate classification such as one-against-all (OAA) and one-against-one (OAO) [23].

3.2 Transfer Learning

TL is a method of applying ML to pretrained models to apply them to different tasks and is currently widely used in the field of image classification. As a simple example, we apply a model trained using car image features to truck classification, because cars and trucks have many similar features, such as tire shapes and bodywork. The advantage of TL is that we do not need to retrain the entire model. We simply load the pretrained model and use its trained parameters; we can customize its hidden layers to meet the usage requirements. TL is typically used when destination source data are scarce or difficult to obtain. Then, the original source datasets can be used for pretraining, and the trained model can be applied to the destination source datasets. However, this assumption is based on the fact that the dataset of the destination source is similar to the original source dataset. Taking the VGG16 pretrained model used in this study as an example, TL can be categorized into two methods: feature extraction and fine-tuning [24].

Feature Extraction

Feature extraction is performed by inputting our prepared images into the VGG16 pretrained model and extracting the features of the input images through the model to output a new image dataset. This method is implemented by removing the topmost layer (the so-called last three layers) of VGG16 from the fully connected layer, as

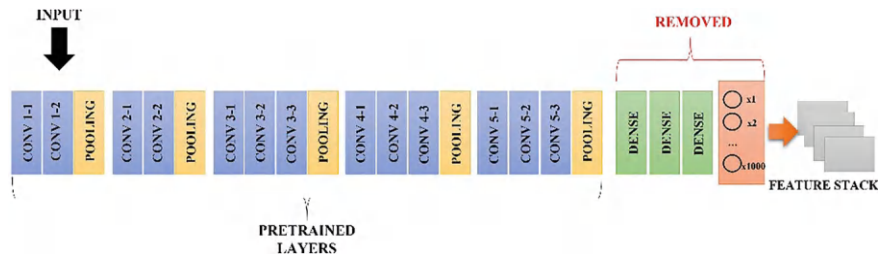


Fig. 1 Extract feature of transfer learning

shown in Fig. 1. Afterward, the VGG16 model outputs the stacked image features, which can be easily converted to the input shape of the specific ML classifier.

Nevertheless, feature extraction methods can be further subdivided into two approaches: stand-alone and bootstrap extractors. The difference between these extractors is that in the former, the new image features extracted by VGG16 are lost to the new ML classifier. In this study, the new classifier is an SVM classifier; thus, this method is also the TL method used in this study. Conversely, the bootstrap extractor also does not use the pretrained fully connected layer but customizes it, randomly assigns weights, and retrain the backpropagation to update the weights.

Fine-Tuning

The goal of fine-tuning is to allow the retraining of some hidden layers that have been pretrained, which requires three elements, namely, bootstrapping the new top layer (fully connected and output layers), frozen pretrained convolutional layers, and a few nonfrozen trainable hidden layers, as shown in Fig. 2. The nonfrozen hidden layer is used to train customized data and adjust the weights according to the predicted result dynamics of the fully connected layer.

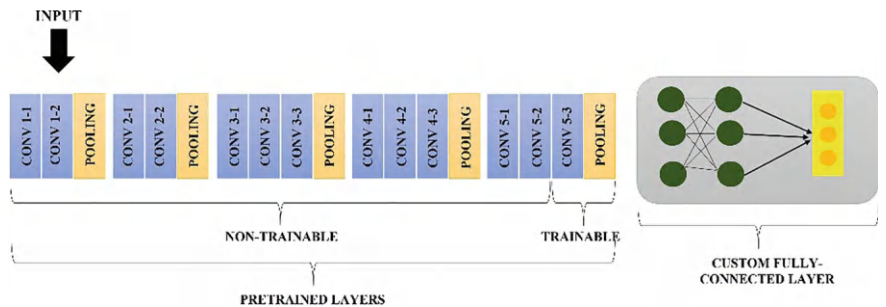


Fig. 2 Fine-tuning of transfer learning

3.3 Proposed Model

The proposed model can be divided into four stages: data preprocessing, loading of the pretrained VGG16 model, feature extraction, and classification. The proposed architecture is shown in Fig. 3.

Data Preprocessing

In the data preprocessing stage, our goal is to convert the data from Kaggle Microsoft 2015 into images. The data format of the raw data is a binary file in hexadecimal representation and does not contain a header, which can be generated using the IDA Pro disassembly tool [25]. Next, we load these binary files and convert them into images of size $224 \times 224 \times 3$ to comply with the input image size of VGG16. Figure 4 shows the conversion of the Ramnit malware sample into images.

Loading of Pretrained VGG16 Model

The VGG16 model pretrained using ImageNet is loaded, and because it is only used for feature extraction, we remove its top layer and only use the weight and bias of the pretrained model for feature extraction. The architecture of the model is shown in Fig. 5.

Feature Extraction

In this step, we input the organized image data from the Microsoft BIG 2015 datasets into the pretrained VGG16 model for feature extraction and obtain a new dataset. We



Fig. 3 Proposed system architecture

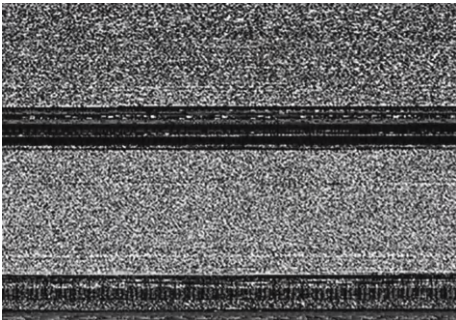


Fig. 4 Malware sample of Ramnit

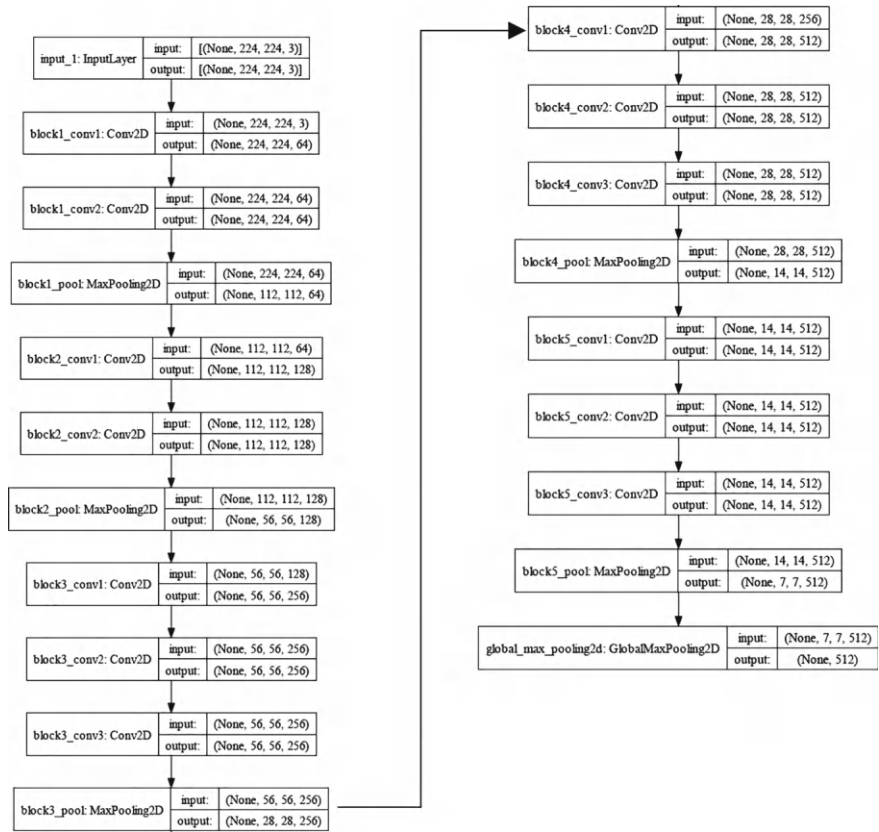


Fig. 5 Pretrained VGG16 model architecture without top layer

then slice the new dataset with a 9:1 ratio of training and testing data and use K-Fold cross-validation to determine which segment has the best performance. We set $K = 10$, which means that the prediction results are divided ten times, each time dividing 90% of the different segments of training data and 10% of the different segments of test data for model training and prediction, respectively.

Classification

Finally, we input the training data into the SVM classifier and determine which segment performs best from the tenfold cross-validation.

4 Experimental Results

We deployed the proposed framework on a machine with Windows 10, an Intel Core i7-13,700 CPU, Geforce GTX 3060 GPUs, and 64 GB RAM. Moreover, we used Python 3.6, cuda == 11.2, cudnn == 8.6, and the packages are summarized in Table 1.

4.1 Dataset Description

The dataset is provided by Microsoft, which is a benchmark in the field of ML visual imaging. This dataset contains records of 21,741 malware programs, the data size is approximately 500 GB after decompression, and it is divided into training and testing sets of 10,860 and 10.873, respectively. However, because the test set is not labeled, only the training set is typically used. The dataset is divided into nine malware types, which are organized into types and families in Table 2. The distribution of the nine malware types is shown in Fig. 6.

Table 1 Summarized used packages

Package name	version
Python	3.6
tensorflow-gpu	2.6.0
Tensorflow	2.6.0
Numpy	1.19.5
Cuda	11.2

Table 2 Summarized using Microsoft BIG 2015 malware sample

Family name	Type	Train sample
Ramnit	Worm	1541
Lollipop	Adware	2478
Kelihos_ver3	Backdoor	2942
Vundo	Trojan	475
Simda	Backdoor	42
Tracur	TrojanDownloader	751
Kelihos_ver1	Backdoor	398
Obfuscator.ACY	Any kind of obfuscated malware	1228
Gatak	Backdoor	1013

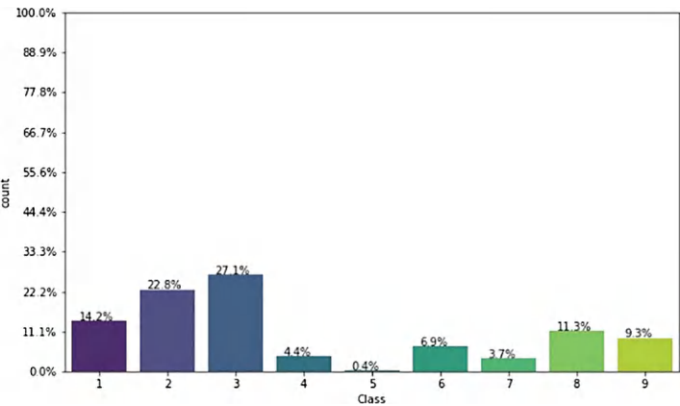


Fig. 6 Distribution of malware classes

4.2 Confusion Matrix

To evaluate the performance of the model, different metrics, including accuracy, recall, precision, and F1-socre, are used. The relationship between true positive (TP), true negative (TN), false positive (FP), and false negative (FN) is shown in Fig. 7.

TP: the model accurately predicts the positive class (both prediction and actual value are positive).

Fig. 7 Relationship between TP, TN, FP, and FN

	Prediction	
	Positive	Negative
Actual		
Positive	TP	FN
Negative	FP	TN

TN: the model accurately predicts the negative class (both prediction and actual value are negative).

FP: the model gives a wrong prediction of the negative class (predicted: positive; actual value: negative). FP is also called a TYPE I error.

FN: the model wrongly predicts the positive class (predicted: negative; actual value: positive). FN is also called a TYPE II error.

Accuracy: The ratio of the number of correctly predicted instances to the total number of instances in the dataset. It is calculated as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \quad (1)$$

Precision: The ratio of TP relevant instances to the total number of retrieved instances. It is calculated as follows:

$$Precision = \frac{TP}{TP + FP}. \quad (2)$$

Recall: The term sensitivity is used to describe the percentage of accurately predicted positive examples compared with the total number of positive occurrences:

$$Recall = \frac{TP}{TP + FN}. \quad (3)$$

F1-score: It is defined as the harmonic mean of precision and recall.

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall}. \quad (4)$$

4.3 Performance of the Proposed Model

The proposed model is evaluated using tenfold cross-validation to determine which training and test datasets yield the best results in terms of accuracy, precision, recall, and F1-score. The results are summarized in Table 3. From the table, the best accuracy (96.78%) is obtained at $K = 1$.

A comparison with related studies is summarized in Table 4.

Table 3 Result of tenfold cross-validation

Numbers of K Loops	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
1	0.9678	0.8792	0.8862	0.8815
2	0.9540	0.8926	0.9009	0.9002
3	0.9641	0.9126	0.9242	0.9086
4	0.9623	0.9015	0.9579	0.9190
5	0.9365	0.8696	0.8552	0.8528
6	0.9531	0.8953	0.8949	0.9009
7	0.9614	0.9529	0.9137	0.9257
8	0.9595	0.8911	0.8902	0.8926
9	0.9586	0.9525	0.9018	0.9181
10	0.9530	0.9237	0.9263	0.9232

Table 4 Comparison with related work

Comparison	Year	Datasets	Accuracy
Simonyan et al. [21]	2022	Microsoft BIG 2015	93.19%
Gupta et al. [22]	2022	Microsoft BIG 2015	90.80%
Alodibat et al. [23]	2023	Microsoft BIG 2015	94%
Our Methods		Microsoft BIG 2015	96.78%

5 Conclusion

In this study, a novel malware classification method is proposed. The method employs CNN–SVM DL based on the TL architecture. The pretrained VGG16 model is modified to be used as a feature extractor, and SVM is used as the classifier. We use the Microsoft BIG 2015 dataset to evaluate the proposed method. The experimental results demonstrate that the proposed method outperforms previous works.

The proposed method has several advantages. First, the use of the pretrained VGG16 model as the feature extractor and SVM as the classifier not only significantly reduces the required model training time but also provides high classification and generalization capabilities without the need for numerous samples. Second, the use of CNN-based DL techniques does not necessitate complex manual feature extraction; not only does it not require a certain level of domain knowledge, but it is also difficult for an attacker to evade through established features.

Acknowledgements This study was partially supported by the National Science and Technology Council of Taiwan, under Grants 111-2221-E-008-061- and 112-2221-E-008-032-.

References

1. Statista, Change in remote work trends due to COVID-19 in the United States in 2020 <https://www.statista.com/statistics/1122987/change-in-remote-work-trends-after-covid-in-usa/>. Accessed 20 Sep 2023
2. World Economic Forum, The Global Risks Report 2023 18th Edition. https://www3.weforum.org/docs/WEF_Global_Risks_Report_2023.pdf. Accessed 20 Sep 2023
3. Fortinet, "What is Malware?" <https://www.fortinet.com/resources/cyberglossary/malware>. Accessed 20 Sep 2023
4. Yadav D, Kumar G, Kameshwari DL, Gunjan VK, Kumar S (2022) Malware techniques and its effect: a survey. ICCCE 2021. Lect Notes Electr Eng, 828, Springer, https://doi.org/10.1007/978-981-16-7985-8_127
5. Security Escape 14 alarming computer worm statistics to keep you alert. <https://securityescape.com/computer-worm-statistics/>. Accessed 20 Sep 2023
6. DataProt, A Not-So-Common cold: malware statistics in 2023. <https://dataprot.net/statistics/malware-statistics/>. Accessed 20 Sep 2023
7. Abidine Bensalem ZE, Benkhaddra I, Setitra MA, Fan M (2022) A novel and efficient sequential learning-based malware classification model. In: 2022 19th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, China, 2022, pp 1–8, <https://doi.org/10.1109/ICCWAMTIP56608.2022.10016605>
8. SSOP, Joseph S (2023) Architectural design of malware infected file detection using deep learning. In: 2023 International Conference on Power, Instrumentation, Control and Computing (PICC), Thrissur, India, pp 1–5, <https://doi.org/10.1109/PICC57976.2023.10142498>
9. Aslan Ö, Yilmaz AA (2021) A new malware classification framework based on deep learning algorithms. IEEE Access 9:87936–87951. <https://doi.org/10.1109/ACCESS.2021.3089586>
10. Ronen R, Radu M, Feuerstein C, Yom-Tov E, Ahmadi M (2018) Microsoft malware classification challenge. [arXiv:1802.10135](https://arxiv.org/abs/1802.10135), [arXiv, USA](https://arxiv.org/abs/1802.10135), <https://doi.org/10.48550/arXiv.1802.10135>
11. Chen Z, Brophy E, Ward T (2021) Malware classification using static disassembly and machine learning. [arXiv:2201.07649](https://arxiv.org/abs/2201.07649), [arXiv, USA](https://arxiv.org/abs/2201.07649), <https://doi.org/10.48550/arXiv.2201.07649>
12. Huang X, Ma L, Yang W (2021) A method for windows malware detection based on deep learning. J Sign Process Syst 93:265–273. <https://doi.org/10.1007/s11265-020-01588-1>
13. Anson P, Anupama ML, Vinod P, Visaggio CA, Aneesh N, Abhijith S, Ananthakrishnan S (2021) Malware detection employed by visualization and deep neural network. In Elsevier Comput & Secur, 105, <https://doi.org/10.1016/j.cose.2021.102247>
14. Darem A, Abawajy J, Makkar A, Alhashmi A, Alanazi S (2021) Visualization and deep-learning-based malware variant detection using OpCode-level features. Elsevier Futur Gener Comput Syst 125:314–323. <https://doi.org/10.1016/j.future.2021.06.032>
15. El-Shafai W, Almomani I, AlKhayer A (2021) Visualized malware multi-classification framework using fine-tuned CNN-Based transfer learning models. Appl Sci, 11, 6446. <https://doi.org/10.3390/app11146446>
16. Rustam F, Ashraf I, Jurcut AD, Bashir AK, Zikria YB (2023) Malware detection using image representation of malware data and transfer learning. Elsevier J Parallel Distrib Comput 172:32–50. <https://doi.org/10.1016/j.jpdc.2022.10.001>
17. Zhao Y, Cui W, Geng S, Bo B, Feng Y, Zhang W (2020) A malware detection method of code texture visualization based on an improved faster RCNN combining transfer learning. IEEE Access 8:166630–166641. <https://doi.org/10.1109/ACCESS.2020.3022722>
18. Kumar S, Janet B (2022) DTMIC: Deep transfer learning for malware image classification. In Elsevier J Inf Secur Appl, 64, <https://doi.org/10.1016/j.jisa.2021.103063>
19. Gupta K, Jiwnani N, Sharif MHU, Datta R, Afreen N (2022) A neural network approach for malware classification. In: 2022 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, pp 681–684, <https://doi.org/10.1109/ICCCIS56430.2022.10037653>

20. Alodibat S, Ahmad A, Azzeh M (2023) Explainable machine learning-based cybersecurity detection using LIME and Secml. In: 2023 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), Amman, Jordan pp 235–242, <https://doi.org/10.1109/JEEIT58638.2023.10185893>
21. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. [arXiv:1409.1556](https://arxiv.org/abs/1409.1556), arXiv, USA, <https://doi.org/10.48550/arXiv.1409.1556>
22. Tibrewal TP (2023) Support Vector Machines (SVM): An Intuitive Explanation. <https://medium.com/@keshavtibrewal2/support-vector-machines-svm-an-intuitive-explanation-b084d6238106>. Accessed 20 Sep 2023
23. Chao CF, Horng MH (2015) The construction of support vector machine classifier using the firefly algorithm. *Comput Intell Neurosci* 2015:212719. <https://doi.org/10.1155/2015/212719>
24. McDermott J (2023) Hands-on transfer learning with keras and the VGG16 model. <https://www.learn datasci.com/tutorials/hands-on-transfer-learning-keras/>. Accessed 20 Sep 2023
25. Hex Rays (2023) IDA Pro. <https://hex-rays.com/ida-pro/>. Accessed 20 Sep 2023

Enhancing Project Programming Hour Prediction with Regression Analysis Techniques—A Case Study of Company D



Chin-Feng Lee, Shih-Chia Chang, and Jau-Ji Shen

Abstract Accurately estimating the project schedule is crucial for the success of a project. Once the project is in progress, there are numerous potential disruptions that can jeopardize the workflow. If the initial forecast is significantly inaccurate, it could lead to project cancellation or incomplete deliverables due to missed deadlines. This study aims to employ various regression analysis methods to provide project managers with the ability to accurately forecast project program development hours, serving as a decision support tool. Data quality and availability also play a pivotal role, as incomplete or inaccurate data can affect the training and evaluation of prediction models. The study collected historical program development data from Company D, subjected it to preprocessing and feature selection, and employed 13 commonly used regression analysis methods for model training and evaluation. By comparing the performance of different analytical approaches, project managers can assess and select the most appropriate method for hour estimation based on project context, resource requirements, and execution efficiency. Additionally, non-expert project managers can quickly evaluate hours and understand critical factors in assessments. Therefore, this research provides valuable insights for project programming hour predictions, positively impacting project efficiency and effectiveness.

Keywords Project management in software development · Schedule forecast · Regression analysis · Machine learning

C.-F. Lee

Department of Information Management, Chaoyang University of Technology, Taichung 41349, Taiwan, ROC

e-mail: lcf@cyut.edu.tw

S.-C. Chang · J.-J. Shen (✉)

Department of Management Information Systems, National Chung Hsing University, Taichung 40227, Taiwan, ROC

e-mail: jjshen@nchu.edu.tw

1 Introduction

In the process of software development, accurately understanding the required workload is necessary and crucial, but it also presents challenges. Therefore, both the academic and industry sectors actively research the estimation and scheduling of software development effort, as it is not only necessary for creating reliable software systems but also impacts project success [1–6]. Consequently, the ability to accurately estimate the hours required for project programming is essential for improving the performance of project personnel and ensuring on-time delivery while meeting customer expectations. Furthermore, in recent years, the complexity and scope of ERP systems have increased, requiring project personnel to simultaneously participate in multiple projects. They must consider various factors that affect project hour estimation, such as system size, functional requirements, technical challenges, and the availability of project personnel. Additionally, project personnel must efficiently allocate limited resources to swiftly and effectively achieve project objectives. Therefore, to address these challenges, the goal of this research is to provide a method that assists project personnel in rapidly and effectively estimating project hours, serving as a support tool for project managers in making resource allocation decisions.

The primary objective of this study is to develop a more effective and accurate method for estimating project programming hours, providing essential reference for project managers in assessing project programming hours, project schedules, and resource allocation. To achieve this research goal, machine learning methods will be employed to construct predictive models, offering a precise solution for estimating project programming hours.

This achievement will not only supply project managers with a more reliable method for estimating project programming hours but also assist them in more efficiently planning and allocating resources, ensuring on-time project delivery, and meeting customer expectations. Additionally, it will serve as a tool for project personnel who may lack experience or familiarity with ERP system operations, allowing for rapid project assessment and identification of potential influencing factors.

2 Proposed Method

2.1 Introduction to Research Data

This study focuses on the renowned domestic ERP system provider, Company D, and specifically examines the development and usage patterns of the five major modules within Company D's mid to large-scale process-oriented ERP system product, which includes Order Management, Procurement, Inventory Management, Accounts Receivable Management, and Accounts Payable Management.

For this research, a total of 662 historical project programming-related data points from the years 2019 to 2020 were collected and used as the dataset. The characteristics and attributes of the research data are presented in Table 1 with development hours measured in hours as the unit.

Table 2 contains detailed information on the first three records of the research data. Let's take the first record as an example. This data pertains to a program development entry within a particular project. The content indicates that the program module is the ACP module, with the program code being ACPB01. The program falls into category 6, known as "double-file type," and it was developed collaboratively by System Designer with code 1 and Program Developer with code 47.

In the context of this project, this program was the 5th one to be developed out of a total of 13 program operations. It does not include an approval feature. During this development, one field was added, but no new tables were introduced in the associated database. This program does not feature an F2 query, and there are no relevant checks associated with it. In the course of this project, the program underwent 118 lines of specification adjustments and was not created as a new program for this project. These specification adjustments did not introduce new formats or call references. The actual development of this program took 2.25 h, which aligns with the target and is the focus of this study.

Table 1 Data features and types

#	Data features	Types	Abbreviation codes
1	System Module	Categorical	Module
2	Program Code Name	Categorical	Code
3	Programming Type	Categorical	Type
4	System Designer	Categorical	SD
5	Programmer	Categorical	PR
6	Development Order	Numerical	Order
7	The Number Of Programs	Numerical	Total
8	Sign Off Function	Binary	Sign
9	The Number Of New Fields	Numerical	NewField
10	The Number Of New Tables	Numerical	NewTable
11	The Number Of Shortcut Query Key F2	Numerical	F2
12	The Number Of Verification	Numerical	Vaild
13	The Number Of Rows	Numerical	Rows
14	New Program or not v	Binary	New
15	The Number Of New Formats	Numerical	Format
16	The Number Of Reference Calls	Numerical	Call
17	Development Hours	Numerical	Target

Table 2 The detailed content of the first three records

#	Module	Code	Type	SD	PR	Order	Total	Sign	Newfield	Newtable	F2	Vaild	Rows	New	Format	Cell	Target
1	ACP	ACPB01	6	1	47	5	13	0	1	0	0	0	118	0	0	0	2.25
2	ACP	ACPB01	4	65	73	20	20	0	0	1	0	0	5	0	0	0	2
3	ACP	ACPB01	4	18	42	1	19	0	0	0	0	0	4	0	0	0	0.1

2.2 *Exploratory Data Analysis (EDA)*

During the data preprocessing process, Exploratory Data Analysis (EDA) techniques are often employed to explore data from various perspectives, identify clues to solving problems, and further validate the issues and solutions. EDA offers a series of statistical analyses, data cleaning, transformation processes, and more to gain a deeper understanding of data features, relationships, and anomalies.

During the data cleaning process, we discovered that there were no duplicate records in the research data, so there was no need to address duplicate data issues. However, when exploring the data using clustering methods, we identified 4 records as outliers, and these outliers were subsequently corrected using manual inspection. In the data transformation phase, our focus was primarily on handling categorical feature columns within the research data. Since our goal is to predict project programming hours, it was necessary to convert categorical features into numerical form for use in subsequent statistical analysis and machine learning. Three feature columns in this research data, namely Module, Code, and Type, required transformation. We adopted the Label encoding method for this purpose. Label encoding is a common data transformation technique used to convert categorical features into numerical form. In Label encoding, each distinct categorical feature is assigned a unique integer label. The benefit of Label encoding is that it preserves the ordinal relationships between categories.

Finally, we normalized the data to ensure comparability between features, eliminate the impact of numerical differences, and enhance the stability and predictive capability of the model.

2.3 *Feature Selection*

A heatmap in Fig. 1 reveals the correlation between each feature and the target value. Notably, the “Rows” column shows the highest correlation coefficient at 0.757, followed by the “NewTable” column at 0.504. “NewField” and “New” come next, both having correlation coefficients of 0.47. Conversely, the five features with the lowest correlation coefficients with the target value, ranked from smallest to largest, are: “Code,” “SD,” “Format,” “Module,” and “Call.”

In summary, “Rows” and “NewTable” exhibit strong positive correlations with the target value, making them potentially significant predictors. “NewField” and “New” have weaker but still notable correlations. In contrast, “Code,” “SD” (System Designer), “Format,” “Module,” and “Call” features display lower correlations, suggesting a relatively minor contribution to predicting the target value.

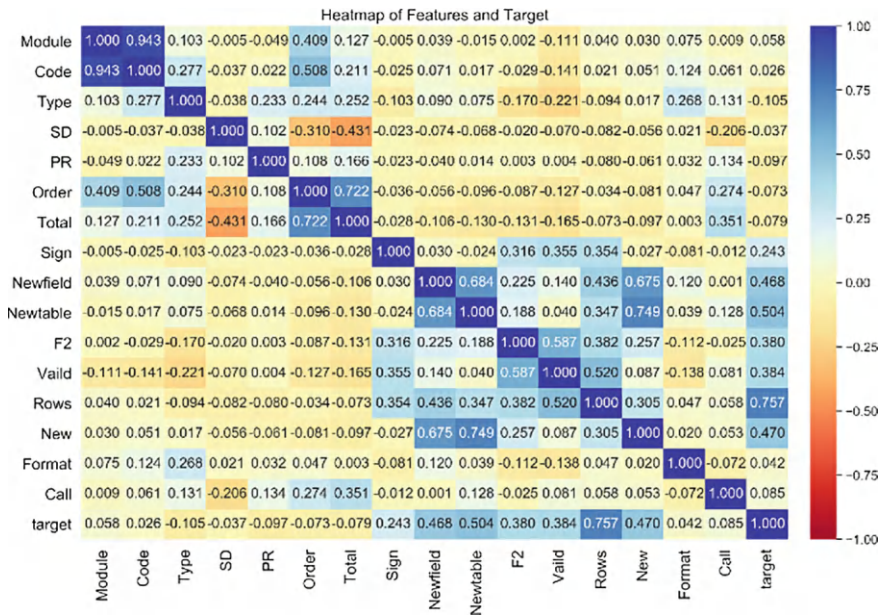


Fig. 1 Correlation coefficient heatmap of various features

2.4 Feature Selection Evaluation

Feature selection is a crucial step in machine learning and data analysis, helping us pick the most influential features from a large set to improve predictive performance and reduce overfitting risk. We utilized 13 common regression analysis methods in Python, including Linear Regression, Ridge, Lasso, Elastic Net, and others [7, 8]. After building these models, we used ten-fold cross-validation to evaluate them, mitigating bias from sample splitting. Metrics like Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R-squared (R2) were employed for objective performance assessment. MSE and RMSE focus on the differences between predicted and actual values, but MSE can be influenced by outliers. MAE focuses on the absolute differences in predicted values, while the coefficient of determination R2 measures the percentage of the total variance in the dependent variable that is explained by the regression model. R2 provides an intuitive, comprehensive, comparable, and concise indicator of prediction accuracy and fit. It allows us to quickly assess the overall model performance and make corresponding decisions. A higher R2 value indicates a better model fit, with values greater than 0.5 generally considered good.

The goal was to identify appropriate regression methods for feature selection, enhancing predictive power, and reducing overfitting risk. Features with lower correlations were removed based on prior model results. These steps aimed to yield a

concise model with strong predictive performance, improving accuracy while minimizing feature impact. This enhances data understanding, uncovers relationships, and provides valuable insights. Evaluation results before and after feature selection are provided.

Taking Linear Regression as an example for the first set of data: According to the evaluation metrics of the Linear Regression model, MSE is 3.887009, RMSE is 1.971550, MAE is 0.966799, and R2 is 0.494091. Based on these metrics, we can conclude that the model's predictive performance is moderate, with some errors between predictions and actual values. Additionally, the model can explain only about 49.4% of the variance in the target variable, indicating limited explanatory power. The higher MSE value suggests significant differences between predicted and actual values. RMSE and MAE values are relatively high, indicating larger average prediction errors. An R2 value of 0.494 indicates that the model can explain only 49.4% of the target variable's variance. Overall, the linear regression model has limited predictive ability, with some prediction errors and limited ability to explain variance in the target variable. In practical applications, it may be necessary to consider other models or improvement methods to enhance predictive performance and explanatory power.

In Table 3, Random Forest, ExtraTrees, LightGBM, and CatBoost performed better in most evaluation metrics, exhibiting lower MSE, RMSE, and MAE values, as well as higher R2 values. On the other hand, the Poly model performed the worst in terms of MSE, RMSE, and MAE metrics, and it had a negative R2 value, which may indicate a significant mismatch or overfitting between the model and the data. The SVR model had higher MSE, RMSE, and MAE values, and it also had a negative R2 value, indicating relatively poorer performance in predicting the target.

Without performing feature selection, models like Random Forest, ExtraTrees, LightGBM, and CatBoost, which performed well, can be considered as preferred candidate models for predicting project development hours. Next, based on previously calculated feature correlations and using a "backward elimination" approach, we removed features such as Module, Code, Format, Call, and SD. After removing these features, we re-evaluated the performance of each model as shown in Table 4. Using the R2 value as an evaluation metric on the data after feature removal, we compared it with the original data. According to the evaluation results, it can be observed that most models showed improvement in their evaluation metrics after feature removal. These models include Linear Regression, Ridge, Random Forest, LightGBM, and CatBoost. This indicates that after removing these features, the models' predictive capabilities and adaptability improved. We successfully identified and removed features that were irrelevant or of lower importance to model training and prediction.

Table 3 Regression model performance evaluation—before feature removal (Rounded to Three Decimal Places)

#	Model	MSE	RMSE	MAE	R2
1	Linear Regression	3.887	1.972	0.967	0.494
2	Ridge	3.457	1.859	0.948	0.542
3	Lasso	8.158	2.856	1.648	−0.038
4	Elastic Net	8.122	2.850	1.646	−0.032
5	Poly(PolynomialFeatures)	149.31	12.219	2.809	−14.77
6	KNN(KNeighborsRegressor)	7.325	2.706	1.477	−0.021
7	RandomForestRegressor	3.345	1.829	0.797	0.596
8	Extra Trees (Extra TreesRegressor)	3.681	1.919	0.807	0.560
9	GBM(GradientBoosting Regressor)	4.761	2.182	0.921	0.454
10	XGBoost(XGBRegressor)	4.664	2.160	0.868	0.450
11	LightGBM	3.409	1.846	0.844	0.599
12	CatBoost (CatBoostRegressor)	3.803	1.950	0.820	0.576
13	SVR	9.011	3.002	1.334	−0.137

Table 4 Regression model performance evaluation—after feature removal

#	Model	MSE	RMSE	MAE	R2
1	Linear Regression	3.876	1.969	0.946	0.499
2	Ridge	3.463	1.861	0.941	0.544
3	Lasso	8.133	2.852	1.643	−0.034
4	Elastic Net	8.101	2.846	1.641	−0.027
5	Poly(PolynomialFeatures)	109.42	10.461	1.871	−12.62
6	KNN(KNeighborsRegressor)	5.540	2.354	1.028	0.332
7	RandomForestRegressor	3.378	1.838	0.788	0.601
8	ExtraTrees (ExtraTreesRegressor)	3.644	1.909	0.819	0.553
9	GBM(GradientBoosting Regressor)	4.708	2.170	0.914	0.457
10	XGBoost(XGBRegressor)	4.674	2.162	0.886	0.439
11	LightGBM	3.393	1.842	0.847	0.595
12	CatBoost (CatBoostRegressor)	3.758	1.938	0.825	0.583
13	SVR	8.584	2.930	1.300	-0.075

3 Model Training and Validation

This section will explain the model building process. Model building is one of the core steps in data mining and machine learning, aiming to construct a model capable of making predictions based on processed data. The process of model building includes several key steps: model selection, parameter optimization, model training, and performance evaluation.

Firstly, we need to initialize the model's parameters. Parameter initialization is a crucial step in model building, as it determines the model's initial state, allowing it to gradually adjust during training to achieve optimal performance. Then, through iterative parameter tuning and model performance evaluation, we can find the best parameter combinations to create a model with the best performance. Next, in Sect. 3, the model will be trained and validated, and in Sect. 4, the trained model will be used to make predictions on the test dataset. We adopt a ten-fold cross-validation method to validate the model, which provides a more accurate estimation of the model's average performance and reduces overfitting issues.

- (1) Linear Regression: Under initial parameters, the model with 'fit_intercept' set to False performs poorly. With the best parameters, 'fit_intercept' set to True, the model's performance improves. The model has lower MSE, RMSE, and MAE values, indicating smaller prediction errors. The R2 value is 0.4029, indicating poor model explanatory power, explaining only approximately 40.29% of the target variable's variance. The model's fit to the data needs improvement, possibly considering nonlinear relationships or data noise.
- (2) Ridge: Under initial parameters, training with various alpha values, the best parameter $\alpha = 1.0$. The model has relatively low MSE, RMSE, and MAE values, indicating lower prediction errors. The R2 value is 0.7205, indicating good model explanatory power, explaining about 72.05% of the target variable's variance.
- (3) Lasso: Under initial parameters, training with various alpha values, the best parameter is $\alpha = 0.01$. The model has lower MSE, RMSE, and MAE values, indicating lower prediction errors. The R2 value is 0.7601, indicating good model explanatory power, explaining about 76.01% of the target variable's variance.
- (4) Elastic Net: Under initial parameters, training with various alpha and l1_ratio values, the best parameters are $\alpha = 0.01$ and $\text{l1\_ratio} = 0.9$. The model has lower MSE, RMSE, and MAE values, indicating lower prediction errors. The R2 value is 0.7422, indicating good model explanatory power, explaining about 74.22% of the target variable's variance.
- (5) Poly(PolynomialFeatures): Under initial parameters and best parameters, the degree is set to 2, representing a quadratic polynomial. The model has relatively higher MSE, RMSE, and MAE values, indicating larger prediction errors. The R2 value is 0.5432, indicating poor model explanatory power, explaining only approximately 54.32% of the target variable's variance.

- (6) KNN(KNeighborsRegressor): Under initial parameters, training with various `n_neighbors` values, the best parameter is `n_neighbors = 3`. The model has higher MSE, RMSE, and MAE values, indicating larger prediction errors. The R2 value is 0.3546, indicating poor model explanatory power, explaining only approximately 35.46% of the target variable's variance.
- (7) RandomForest(RandomForestRegressor): Under initial parameters and best parameters, `n_estimators` is set to 100, `max_depth` is set to 10, `min_samples_split` is set to 5, and `min_samples_leaf` is set to 4. The model has lower MSE, RMSE, and MAE values, indicating lower prediction errors. The R2 value is 0.6309, indicating good model explanatory power, explaining about 63.09% of the target variable's variance.
- (8) ExtraTrees (ExtraTreesRegressor): `n_estimators` is set to 100, `max_depth` is set to 10, `min_samples_split` is set to 10, and `min_samples_leaf` is set to 8. The model has lower MSE, RMSE, and MAE values, indicating lower prediction errors. The R2 value is 0.6476, indicating good model explanatory power, explaining about 64.76% of the target variable's variance.
- (9) GBM(GradientBoostingRegressor): Under initial parameters and best parameters, `n_estimators` is set to 200, `learning_rate` is set to 0.01, `max_depth` is set to 3, `min_samples_split` is set to 10, and `min_samples_leaf` is set to 10. The model has lower MSE, RMSE, and MAE values, indicating lower prediction errors. The R2 value is 0.5954, indicating poor model explanatory power, explaining only approximately 59.54% of the target variable's variance.
- (10) XGBoost (XGBRegressor): Under initial parameters and best parameters, the model uses multiple parameter combinations, with the best parameters being: `n_estimators = 100`, `learning_rate = 0.05`, `max_depth = 5`, `min_child_weight = 3`, `gamma = 0.1`, `subsample = 0.2`, `colsample_bytree = 0.8`. The model has lower MSE, RMSE, and MAE values, indicating lower prediction errors. The R2 value is 0.6518, indicating good model explanatory power, explaining about 65.18% of the target variable's variance.
- (11) LightGBM: Under initial parameters and best parameters, the model uses multiple parameter combinations, with the best parameters being: `n_estimators = 200`, `learning_rate = 0.01`, `max_depth = 8`, `num_leaves = 20`, `min_child_samples = 8`, `subsample = 0.6`, `colsample_bytree = 0.7`. The model has lower MSE, RMSE, and MAE values, indicating lower prediction errors. The R2 value is 0.5808, indicating poor model explanatory power, explaining only approximately 58.08% of the target variable's variance.
- (12) CatBoost (CatBoostRegressor): Under initial parameters and best parameters, the model uses multiple parameter combinations, with the best parameters being: `iterations = 200`, `learning_rate = 0.05`, `depth = 5`, `l2_leaf_reg = 1`, `subsample = 1.0`, `colsample_bylevel = 0.6`. The model has lower MSE, RMSE, and MAE values, indicating lower prediction errors. The R2 value is 0.6320, indicating good model explanatory power, explaining about 63.20% of the target variable's variance.
- (13) SVR: Under initial parameters and best parameters, the kernel is set to linear, `C` is set to 10, and `epsilon` is set to 0.015. The model has lower MSE, RMSE,

and MAE values, indicating lower prediction errors. The R2 value is 0.6456, explaining about 64.56% of the target variable's variance.

In summary, based on the evaluation metrics of each model, the following conclusions can be drawn:

- (1) Ridge Regression, Lasso Regression, Elastic Net Regression, RandomForest Regression, ExtraTrees Regression, XGBoost Regression, CatBoost Regression, and SVR perform well on this dataset. They can fit the data effectively and have a high level of explanatory power.
- (2) On the other hand, Linear Regression, Polynomial Features, KNeighbors Regression, GradientBoosting Regression, and LightGBM Regression perform poorly on this dataset. They exhibit larger prediction errors and weaker explanatory power. These models may require further parameter optimization or the adoption of alternative methods to enhance their performance.

4 Model Testing and Evaluation

In this section, we will evaluate the models based on the optimal parameters established in the previous section. Evaluating the performance of the models is a crucial step in machine learning, as it helps us understand how the models perform and their accuracy in real-world applications.

Additionally, the $\pm 50\%$ accuracy column refers to the proportion of data points in the prediction evaluation where the predicted values fall within plus or minus 50% of the actual values. Through interviews with experienced project management experts, we have identified that there can be significant variations between estimated and actual development hours in software development, ranging from 1 to 3 h or even as high as 10 to 30 h. Such variances can become more noticeable in overall project hour estimations. Given this situation and based on expert experience, we infer that in typical software development hour estimates ranging from 3 to 10 h, the differences between predicted and actual hours often fall within the $\pm 50\%$ range. Therefore, this study considers $\pm 50\%$ accuracy as an evaluation criterion, as shown in Table 5.

Based on the evaluation results, we can observe variations in the performance of different models in terms of the R2 value (as shown in Table 5). Among them, the LightGBM model performs the best, with an R2 value of 0.596, indicating that this model can explain approximately 59.6% of the variance in the target variable. Next are the GradientBoostingRegressor model and the XGB Regressor model, with R2 values of 0.5791 and 0.577, respectively, demonstrating higher explanatory power. In contrast, the PolynomialFeatures model has an R2 value of only 0.0967, indicating a weaker ability to explain the target variable, accounting for only about 9.67% of the variance in the target variable. Therefore, we can conclude that the PolynomialFeatures model is not suitable for predicting development hours.

Furthermore, concerning the prediction accuracy of the models, we can refer to the $\pm 50\%$ accuracy indicator. $\pm 50\%$ accuracy represents the percentage of predicted

Table 5 Comparison of prediction evaluation results for various regression models

Model	MSE	RMSE	MAE	R2	± 50% 準確度 (%)
Linear Regression	1.3471	1.1607	0.8253	0.4790	45.11
Ridge	1.2418	1.1143	0.8255	0.5197	42.86
Lasso	1.2705	1.1272	0.8295	0.5086	44.36
Elastic Net	1.2205	1.1047	0.8284	0.5280	43.61
PolynomialFeatures	2.3357	1.5283	0.8681	0.0967	48.12
KNeighbors Regressor	1.4147	1.1894	0.7162	0.4529	63.91
RandomForestRegressor	1.3032	1.1416	0.7139	0.4960	62.41
ExtraTreesRegressor	1.1672	1.0804	0.7179	0.5486	53.38
GradientBoostingRegressor	1.0883	1.0432	0.7395	0.5791	45.86
XGBRegressor	1.0937	1.0458	0.6894	0.5770	55.64
LightGBM	1.0445	1.0220	0.7120	0.5960	50.38
CatBoostRegressor	1.1183	1.0575	0.7117	0.5675	55.64
SVR	1.2049	1.0977	0.6975	0.5340	52.63

values falling within the range of -50% to $+50\%$ of the actual values. Among these indicators, the KNN model has the highest $\pm 50\%$ accuracy, reaching 63.91%. This implies that a higher percentage of data falls within the $\pm 50\%$ range of deviation between the predicted values and actual values for this model. In contrast, the PolynomialFeatures model has a $\pm 50\%$ accuracy of only 48.12%, indicating poorer prediction accuracy.

In summary of the above evaluation results, we can conclude that the LightGBM, GBM, and XGBoost models perform better in predicting development hours. Additionally, the KNN model has an R2 value of 0.4529, but its accuracy in predicting values within the $\pm 50\%$ range is 63.91%. This situation may be because the KNN model assigns higher weights to neighboring training samples during prediction, allowing it to make better predictions for test samples with similar features. This explains why the model achieves a relatively high accuracy in predicting values within the $\pm 50\%$ range.

However, it's important to note that despite the KNN model performing well in $\pm 50\%$ accuracy, its R2 value is lower, indicating weaker explanatory power for the target variable. This might suggest that the model has greater uncertainty in the differences between predicted values and actual values or is influenced by outliers or noise during prediction.

5 Conclusions

In the previous research, based on the model building in Sect. 3 and model evaluation in Sect. 4, we have concluded that LightGBM, GBM, and XGBoost are the best models for predicting software development hours. These three methods all belong to gradient boosting-related algorithms, demonstrating excellent performance in predicting software development hours. These algorithms exhibit efficiency and scalability when dealing with large-scale datasets. They utilize parallel computing and tree-based model structures, enabling fast training and prediction. This allows them to handle large datasets quickly and adapt to the needs of practical project management.

The rapid training and prediction capabilities of gradient boosting algorithms enable project managers to quickly obtain predictions for software development hours, thus improving work efficiency and project progress. Therefore, in practical applications dealing with a large volume of data, these three methods are equally applicable.

In summary, LightGBM, GradientBoostingRegressor, and XGBRegressor, these three gradient boosting algorithms perform exceptionally well in predicting software development hours. They can accurately capture complex relationships in the data, possess good explanatory power, and exhibit efficiency and scalability when handling large-scale data. This makes them valuable tools in practical project management, providing project managers with fast and accurate predictions for software development hours, thereby enhancing work efficiency. According to Table 5, they can be selected in the following order: LightGBM > GBM > XGBoost. Additionally, if there are considerations regarding equipment requirements, such as insufficient memory resources or lack of support for GPU environments, then GBM > XGBoost > LightGBM can be prioritized for stability in model performance. Furthermore, if the environment does not support gradient boosting and consumes more device resources but allows predictions within $\pm 50\%$, then ExtraTrees and SVR can be given priority.

Taking into account performance evaluation in various aspects, assuming an R^2 above 0.5 is considered good performance, under this condition and considering the comprehensive performance of prediction accuracy, RMSE, MAE, and $\pm 50\%$ accuracy, the preferred order would be XGBoost > SVR > CatBoost > LightGBM.

Finally, it is worth noting that the research by Zhang Qiongwen (2011) [1], titled “Constructing a Prediction Model for ERP Development Hours Using a Hierarchical Estimation Method—A Case Study of Company D,” also focused on the application of Company D as an example. Below, we will make a relevant comparison between Zhang Qiongwen (2011) and this study and offer the following insights:

Firstly, Chang (2011) [6] proposed a study concerned with project hour predictions for Company D. However, there are differences in research content and data processing methods. Specifically, Chang’s research started with projects and predicted project development hours, while this study started with individual programs and predicted project programming hours. Additionally, although some

features in the research data may appear similar, there are differences in the content and processing methods of the acquired data. For example, the “program category” feature in Chang’s research was split into five different features: documentation, reports, batch processes, queries, and reports. In contrast, this study only normalized the categorical data. Similarly, the “approval” feature in Chang’s research represented the number of programs with approval functionality throughout the entire project, while in this study, it was represented as binary data (0 or 1) indicating whether a program had approval functionality. According to the results of this study, considering only whether a program has approval functionality does not show a strong correlation with the prediction target. Therefore, we suggest conducting in-depth interviews with senior project managers to explore and adjust methods for calculating the number of approval functionalities associated with certain features to improve their correlation with the target variable. Additionally, whether dealing with categorical or numerical data, discussing how to count feature data with senior project managers can potentially enhance the relevance to the target variable or improve accuracy.

In conclusion, poor performance of the model on the test set may involve issues such as overfitting, improper feature selection or engineering, and data quality problems. Addressing these issues will contribute to improving the model’s predictive ability and generalization performance. In practical applications, we should thoroughly evaluate the model’s complexity, feature processing, and data quality. Interviews with senior project managers can help in extracting and optimizing relevant data to obtain more reliable and accurate prediction results.

Acknowledgements This research was supported by the National Science and Technology Council, Taiwan R.O.C., under contract number MOST 111-2221-E-324-019-MY2.

References

1. Boehm BW (1981) Software engineering economics., Prentice-Hall
2. Carmona P, Climent F, Momparler A (2019) Predicting failure in the US banking sector: An extreme gradient boosting approach. *Int Rev Econ Financ* 61:304–323
3. Klemens JF (2004) Estimating software development time. *IEEE Softw* 21(4):82–87
4. Jorgensen M, Shepperd M (2007) A systematic review of software development cost estimation studies. *IEEE Trans Softw Eng*
5. Leung H, Fan Z. Software cost estimation, Volume II,” *Handbook of Software Engineering*
6. Chang CW (2011) Constructing an ERP development time prediction model using a hierarchical estimation method—A case study of Company D. Institute of Information Management, National Central University, Academic Thesis
7. Paszke A et al (2019) PyTorch: An imperative style, high-performance deep learning library. In *Adv Neural Inf Process Syst* 32:8024–8035
8. Luca Massaron, Alberto Boschetti (2016) Regression analysis with python: learn the art of regression analysis With Python. ISBN:9781785286315, Packt Publishing

Network Security (II)

A New Traceable One-Time Address Scheme Secure Against Privilege Escalation Attack



Tse-Jui Chen, Jen-Chieh Hsu, and Raylin Tso

Abstract One-time addresses in cryptocurrencies offer genuine anonymity. Despite the privacy benefits, Monero, a leading true anonymity cryptocurrency utilizing one-time addresses, faces privilege escalation attacks compromising security. Addressing this, we propose a secure traceable one-time address scheme capable of withstanding key leakage without compromising overall security. Our approach enhances regulatory capabilities absent in true anonymity cryptocurrencies. Additionally, our scheme streamlines the process of verifying user addresses, minimizing computational costs. We present a prototype implementation for performance analysis.

Keywords Blockchain · One-time address · Anonymity · Traceability · Privilege escalation attack

1 Introduction

Monero [7] have been developed to offer true anonymity. However, these anonymity cryptocurrencies pose challenges in effective regulation, as they can potentially be employed for illegal activities such as terrorist funding and money laundering, given their difficulty of tracing. Consequently, the regulation has become crucial for these anonymity-enhanced cryptocurrencies. Zhao et al. [14] proposed a traceable one-time address scheme for digital museum assets, providing managers with the ability to trace the real identity behind one-time addresses.

T.-J. Chen · J.-C. Hsu · R. Tso (✉)
National Chengchi University, Taipei 11605, Taiwan
e-mail: raylin@cs.nccu.edu.tw

T.-J. Chen
e-mail: 110753119@nccu.edu.tw

J.-C. Hsu
e-mail: 105753501@nccu.edu.tw

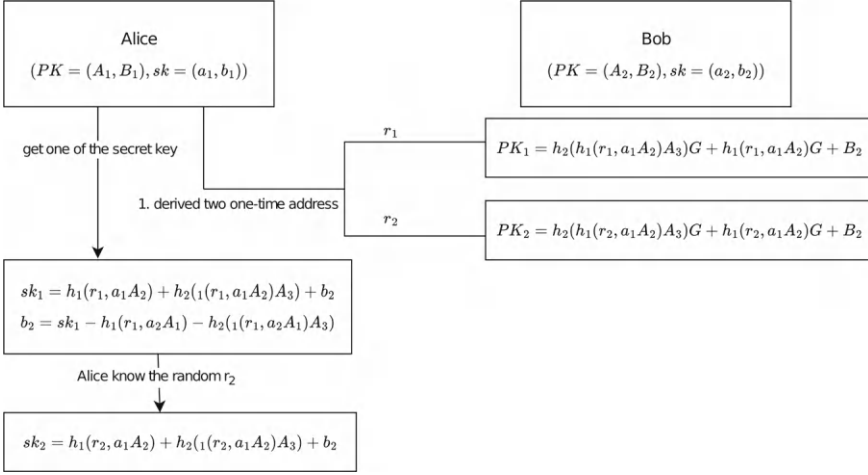


Fig. 1 Zhao et al. [14] implement is not secured against privilege escalation attack

However, this solution may be susceptible to other attacks, such as privilege escalation. A privilege escalation attack occurs when one derived secret key is compromised, allowing the attacker to compute the corresponding master secret key, rendering it no longer secure. Consequently, all the other derived keys may also be compromised. As a result, the coins belonging to the corresponding master key and all the derived keys could be stolen by the attacker. Figure 1 illustrates that the traceable one-time address scheme, as invented by Zhao et al. [14], is susceptible to this problem. For instance, suppose there are three users: the payer Alice, the receiver Bob, and the manager ($PK = A_3$). If the payer Alice wishes to send something to Bob twice, she derives two one-time addresses: PK_1 and PK_2 . Let's assume Alice obtains one of the secret keys sk_1 . With this key, she can compute Bob's secret key, b_2 . Furthermore, she can also calculate sk_2 , as she knows the random value r_2 . Additionally, Alice could collaborate with another payer who has transacted with Bob, thereby compromising all the derived keys. Consequently, all the one-time addresses associated with Bob would no longer be secure.

1.1 Our Contribution

We introduce a new traceable one-time address scheme that satisfy:

- **Secure against privilege escalation attacks:**

Our scheme can tolerate one time secret key leakage, meaning that if someone compromises our derived secret key once, they cannot compromise the corresponding secret key.

- **Eliminates the need for a secret channel:**

Our scheme eliminates the need for a secret channel to send the sender's address to the receiver, in contrast to the scheme proposed by Zhao et al. [14].

- **Expedite the process of verifying a one-time address:**

We have expedite the process of verifying whether a one-time address belongs to a particular user. This could reduce the computational power requirements and prevent users who lack computational power from relying on trusted third parties to identify their transactions.

2 Related Work

2.1 Techniques Related to Privacy Protection

Bitcoin [6] provides pseudonymity instead of true anonymity. Since all transactions are publicly available, certain analyses, such as transaction graph analysis [9], and address clustering [8], could potentially reveal the real identities of users.

Zerocash [10] hide both the participants and the amount, and support transactions of any denomination. Furthermore, it utilizes zero-knowledge succinct non-interactive arguments of knowledge (zk-SNARKs) to achieve small proof sizes.

Monero is based on the CryptoNote protocol [12]. It employs linkable ring signatures [3] to obscure the source in a transaction and one-time addresses to conceal the destination address. Furthermore, the transaction amount is hidden from public view through confidential transactions. However, while the anonymous set in the ring signature is small, attackers can still determine the relationship between transaction addresses and real identities. A larger anonymous set could strengthen anonymity, but it would require a significant increase in storage costs. In 2017, a more efficient RingCT 2.0 [11] was proposed, which utilizes accumulators with a one-way domain to significantly reduce storage space requirements. However, these privacy-enhanced cryptocurrencies are lack of regulation and auditing capability.

2.2 Traceability

Lin et al. proposed a decentralized conditional anonymous payment system (DCAP) [2], aimed at concealing the real identities of participants in transactions. The manager could trace back to the real identity of suspicious transactions. However, a significant limitation of DCAP is that the system's manager requires access to the real identity of each participant to validate transactions. Furthermore, the manager could reject any transaction if he want, giving them potentially excessive power. Zhao et al. introduced a traceable one-time address scheme [14], which modified the Monero one-time address scheme to add traceability. However, in every transaction,

the sender needs to securely send the first part of their public key to the recipient, which could create an extra burden. Additionally, the proposed scheme is not secure against privilege escalation attacks, as private key leakage could lead to significant problems.

2.3 Privilege Escalation Attack

Gus Gutoski and Douglas Stebila proposed a hierarchical deterministic wallet that tolerates key leakage [1]. It is secure against privilege escalation attacks by increasing the key size. However, if one wants to tolerate n times key leakage, the key size would need to be n times larger. This trade-off between security and key size can become prohibitive. Yin et al. proposed a Secure Hierarchical Deterministic Wallet Supporting Stealth Address system (HWDSA) [13], which is a secure wallet algorithm that combines the advantages of both hierarchical deterministic wallets and one-time addresses while remaining secure against privilege escalation attacks. Liu et al. introduced a Publicly Derived Public Key System (PDPKS) [4], which utilizes key insulation to resist privilege escalation attacks. However, none of the above papers provide traceability in their systems.”

3 Definitions of Partial Security Traceable One-Time Address Scheme Against Privilege Escalation Attack

3.1 System Architecture

System Description There are three entities in our proposed traceable one-time address scheme, that is user, viewer and manager. Entity details are as follows:

- *user*: Users have their own public keys and secret keys, enabling them to send and receive transactions using these keys. The secret key consists of two parts: the first part is used to find the one-time address belonging to the user, while the second part is used to generate the corresponding derived secret key for the one-time address.
- *viewer*: A viewer is a third party that assists users who lack of bandwidth or computational power to identify their one-time addresses.
- *manager*: Manager are special nodes on the system who are capable of tracing back to the real identity of the one-time address in the suspicious transactions.

The workflow of our scheme can be simplified as shown in the Fig. 2. To initiate a transaction, the sender generates a one-time address based on the receiver's and manager's public keys and sets the one-time address as the receiver in the transaction. The sender then broadcasts the transaction to the blockchain. The receiver observes all the new transactions on the blockchain and identifies the one-time address that

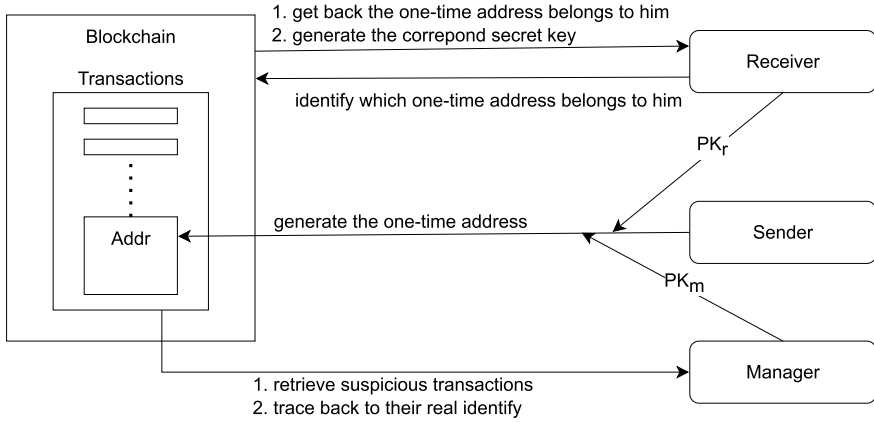


Fig. 2 The workflow of the proposed scheme

belongs to them using their secret key. Once the receiver finds the one-time address that belongs to them, they can use the secret key to generate the corresponding derived secret key. If there are any suspicious transactions, the manager can trace the one-time address and determine the real identity of the receiver.

System Model

- $Setup(\lambda) \rightarrow PP$
On inputting a security parameter λ , the algorithm output the public parameter PP .
- $KeyGen(1^\lambda) \rightarrow (PK_i, MSK_i)$
On inputting a security parameter, the user's generate the secret key $SK_i = (a_i, b_i)$, the algorithm output the public key $PK_i = (A_i, B_i)$ of the user.
- $OnetimeAddrGen(PK_r, PK_m) \rightarrow (Addr, R)$
On inputting the receiver's public key $PK_r = (A_r, B_r)$, manager's public key A_m . The algorithm output the one-time address $Addr$ and auxiliary information R .
- $OnetimeSKGen(Addr, R, SK_r) \rightarrow (dsk, r)$
On inputting the one-time address $Addr$, auxiliary information R and user's secret key $SK_r = (a_r, b_r)$. The algorithm output the derived secret key dsk correspond to the one-time address and auxiliary information r .
- $ReceiverGen(r, SK_r) \rightarrow (dsk, Addr)$
On inputting the auxiliary information r and the receiver's secret key SK_r . The algorithm output the $dsk, Addr$.
- $AddressBeloner(Addr, a_r) \rightarrow (1, 0)$
On inputting the user's first part secret key a_r to check whether the one-time address $Addr$ belongs to him. Output 1 if the $Addr$ belongs to the user, output 0 if the $Addr$ doesn't belong to the user.

- $IdentityTracing(Addr, a_m) \rightarrow B_r$

On inputting the one-time address $Addr$ and the manager's secret key a_m . The algorithm output the real identity B_r of the one-time address.

3.2 Security Model

In this section, we define the security model for our proposed one-time address scheme to capture the requirement of privacy and security for the users.

For privacy purposes, unlinkability is defined by the following Game_{UNL} . For all PPT adversaries $\mathcal{A}$, they are not able to identify the one-time address, out of two public keys, from which the one-time address was generated from. Therefore, the adversary cannot identify the real identity of the one-time address

Definition 1 A traceable one-time address scheme is public key unlinkable, if for all PPT adversaries $\mathcal{A}$, the probability of $\mathcal{A}$ winning the following Game_{UNL} is negligible.

- **Setup.** The challenger runs *Setup* and *KeyGen* algorithms and get the PP, PK, PK_m then send PP, PK, PK_m to $\mathcal{A}$.
- **Query Phase.** In this phase, the adversary $\mathcal{A}$ make hash query to the challenger and construct the one-time addresses.
- **Challenge.** The adversary $\mathcal{A}$ outputs two distinct public keys PK_1 and PK_2 to challenger. The challenge chooses a random bit $b \in \{0, 1\}$ and compute $OnetimeAddrGen(PK_b, PK_m) \rightarrow (Addr, R)$ and send $Addr$ back to $\mathcal{A}$.
- **Guess.** The adversary $\mathcal{A}$ outputs a bit $b' \in \{0, 1\}$ as its guess to b . $\mathcal{A}$ wins the game if $b' = b$.

For security purposes, our scheme is secure against privilege escalation attack one time, meaning it can tolerate key leakage once, as demonstrated in the following Game_2 . If an one-time address secret key is leaked, for any probabilistic polynomial-time (PPT) adversary $\mathcal{A}$, no matter they are the sender or another user, they cannot compute the corresponding secret key.

Definition 2 A traceable one-time address scheme is secure against one-time privilege escalation attack, if for all PPT adversaries $\mathcal{A}$, the probability of $\mathcal{A}$ winning the following Game_2 is negligible.

- **Setup.** The challenger runs *Setup* and *KeyGen* algorithms to get the PP, PK_r, PK_m . Then send PP, PK_r, PK_m to $\mathcal{A}$.
- **Challenge.** The adversary $\mathcal{A}$ set the PK_r as the recipient and generate two one-time address $Addr_0, Addr_1$ and then send $Addr_0, Addr_1$ back to the challenger. The challenger executes $OnetimeSKGen(Addr_0, MSK) \rightarrow (dsk_0, r)$ and sends the resulting dsk_0 back to the adversary $\mathcal{A}$.
- **Output Phase.** The adversary $\mathcal{A}$ outputs a forgery secret key (a'_r, b'_r) . $\mathcal{A}$ wins if $(a'_r, b'_r) = (a_r, b_r)$.

4 The Proposed Scheme

In this section, we first present some preliminaries, then we proposed our scheme.

4.1 Preliminaries

Bilinear Map Group: Let λ be a security parameter and p be a λ -bit prime number. Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be cyclic groups of order p where $\mathbb{G}_1$ is additive and $\mathbb{G}_2$ is multiplicative. G is the generator of $\mathbb{G}_1$. We call $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ a bilinear map if it satisfying the following properties:

1. **Bilinearity:** $\forall a, b \in \mathbb{Z}_p, e(aG, bG) = e(G, G)^{ab} = e(bG, aG)$
2. **Non-degeneracy:** $e(G, G) \neq 1$
3. **Computable:** $\forall a, b \in \mathbb{Z}_p, (aG, bG) \in (\mathbb{G}_1, \mathbb{G}_1), e(aG, bG)$ is efficiently computable.

4.2 Construction

Our scheme consist of seven algorithms, namely: **Setup**, **KeyGen**, **OnetimeAddrGen**, **OnetimeSKGen**, **ReceiverGen**, **AddressBeloner**, **IdentityTracing**. We do not adopt the signature mechanism because our scheme could directly adopt to the same signature scheme in Monero.

- $Setup(\lambda) \rightarrow PP$

On input a security parameter λ , the algorithm chooses bilinear map groups $(p, \mathbb{G}_1, \mathbb{G}_2, e, G)$ of prime order $n > 2^\lambda$. G is the generator of $\mathbb{G}_1$. $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2, e(G, G) = g$. Two hash functions $h_1 : G \rightarrow Z_n$ and $H : \mathbb{G}_2 \rightarrow Z_n$. The algorithm output the public parameter:

$$PP = ((n, \mathbb{G}_1, \mathbb{G}_2, e, G), h_1, H, Z_n)$$

- $KeyGen(1^\lambda) \rightarrow (PK_i, SK_i)$

There are three users of the algorithm. They are the sender, receiver and the manager. The user randomly choose $a_i, b_i \in Z_n$, and computes $A_i = a_i \cdot G, B_i = b_i \cdot G$. The secret key is $SK_i = (a_i, b_i)$, the public key of the user is $PK_i = (A_i, B_i)$. Then output $(PK_i, SK_i) i \in \{s, r, m\}$. s means the sender, r means the receiver and m means the manager.

- $OnetimeAddrGen(PK_r, PK_m) \rightarrow (Addr, R)$

The sender randomly choose a random $r_1 \in Z_n$, receiver's public key (A_r, B_r) and manager's public key A_m . Then the algorithm computes as follow:

1. $R_1 = r_1 G$
2. $r_2 = h_1(r_1 A_r)$
3. $R_2 = r_2 A_r$
4. $r_3 = H(e(R_2, A_m)^{r_1})$
5. $Addr = r_3 G + R_2 + B_r$

The output one-time address for the receiver $Addr$ and the auxiliary information $R = (R_1, R_2)$.

- $OnetimeSKGen(Addr, SK_r) \rightarrow (dsk, r)$

The receiver take its secret key $SK_r = (a_r, b_r)$, the manager's public key A_m and the one-time address $Addr$ as the input. Then the receiver first checks if the one-time address belong to himself. The receiver compute as follow:

1. $r_2 = h_1(a_r R_1)$
2. $R'_2 = r_2 A_r$

In the above algorithm, the receiver doesn't need to compute the $Addr$. He can just check the $R'_2 = R_2$. If the check pass, the receiver run the below algorithm:

1. $r_3 = H(e(R_1, A_m)^{r_2 a_r})$
2. $dsk = r_3 + r_2 a_r + b_r$

and output the derived one-time address secret key dsk and the auxiliary information $r = (r_2, r_3)$ which can be used for generating the derived secret key in the future.

- $ReceiverGen(r, SK_r) \rightarrow (dsk, Addr)$

If the user forget which one-time address belongs to him or the correspond derived secret key. He could use the auxiliary information $r = (r_2, r_3)$ and his secret key $SK_r = (a_r, b_r)$ to compute:

1. $dsk = r_3 + r_2 a_r + b_r$
2. $Addr = dsk \cdot G$.

Then output $(dsk, Addr)$. This can find the $Addr$ faster instead of go through all the blockchain to find the one-time address belongs to him.

- $AddressBeloner(Addr, R, a_r) \rightarrow (1, 0)$

If the user is lack of the computation power, he can provide the first part of his secret key a_r to the trust authority. The trust authority can then execute this algorithm. On inputting the user's a_r , the auxiliary information $R = (R_1, R_2)$ and $Addr$ to compute R'_2 as follow:

1. $r_2 = h_1(a_r R_1)$
2. $R'_2 = r_2 A_r$

If $R'_2 = R_2$, then the *Addr* belongs to the user and output 1. If not, it outputs 0 which means the *Addr* doesn't belong to the user.

- *IdentityTracing*(*Addr*, R , a_m) $\rightarrow B_r$

The manager take its secret key a_m , the one-time address *Addr* and the auxiliary information $R = (R_1, R_2)$ as the input. Then the manager run the below algorithm:

1. $r_3 = H(e(R_1, R_2)^{a_m})$
2. $B_r = PK - r_3G - R_2$

and outputs the real identity B_r of the one-time address. Therefore, the manager can trace back to the real identity.

Correctness. For any one-time address.

1. OnetimeSKGen: $e(R_1, A_m)^{r_2a_r} = e(G, G)^{r_1a_m r_2a_r} = e(r_2a_r G, a_m G)^{r_1} = e(R_2, A_m)^{r_1}$
2. IdentityTracing: $e(R_1, R_2)^{a_r} = e(G, G)^{r_1a_m r_2a_r} = e(r_2a_r G, a_m G)^{r_1} = e(R_2, A_m)^{r_1}$.

5 Security

Theorem 1 Suppose two hash functions h_1, H is a random oracle. Our traceable one-time address scheme is public key unlinkable, if for all PPT adversaries $\mathcal{A}$, the probability of $\mathcal{A}$ identifying the real identity of the one-time address is $1/n$. n is the order of Z_n .

Proof Assume that there exists a polynomial-time adversary $\mathcal{A}$ who can identify the real identity of the one-time address, meaning the adversary can compute the one-time address corresponding to a specific real identity. In the Game_{UNL} , the adversary can make q_h, q_H queries and the challenger simulates the oracles to response all queries for $\mathcal{A}$ as follow:

- **Setup.** The challenger execute the *Setup* and *KeyGen* to generate $PP = ((n, \mathbb{G}_1, \mathbb{G}_2, e, G), h_1, H, Z_n)$ and $PK_r = (A_r, B_r)$, $PK_m = (A_m, B_m)$. Then the challenger sends PP, PK_r, PK_m to the adversaries $\mathcal{A}$. Two empty hash lists are set as L_{h1}, L_H and initialized, each element of which will be (G_1, r_2) and (G_2, r_3) ($G_2 \in \mathbb{G}_1, G_2 \in \mathbb{G}_2, r_2, r_3 \in Z_n$). n is the order of Z_n .
- **Query Phase.** In this phase, the adversaries $\mathcal{A}$ make hash queries to the challenger. The challenger records all hash queries and their responses in L_{h1}, L_H . Upon receiving a hash query with G_{1i} , the challenger randomly choose $r_{2i} \in Z_n$ and sets $r_{2i} = h_1(G_{1i})$. Then response with r_{2i} and record (G_{1i}, r_{2i}) in L_{h1} . If receiving a hash query with G_{2i} , the challenger randomly choose $r_{3i} \in Z_n$ and sets $r_{3i} = H(G_{2i})$. Then response with r_{3i} and record (G_{2i}, r_{3i}) in L_H . If $\mathcal{A}$ makes a query that has been asked before, the challenger looks up the hash lists to find the corresponding entries and returns the value to $\mathcal{A}$. Then the adversary $\mathcal{A}$ can use the returned values of the hash queries to construct one-time addresses.

- **Challenge.** The adversary $\mathcal{A}$ send two distinct public key PK_0, PK_1 to the challenger to be challenged. The challenger randomly choose $b \in \{0, 1\}$, and $r_1 \in \mathbb{Z}_n$ compute as follow:

$$\begin{aligned}
 - R_b &= r_1 G \\
 - r_{2b} &= h_1(r_1 A_b) \\
 - R_{2b} &= r_{2b} A_b \\
 - r_{3b} &= H(e(R_{2b}, A_m)^{r_1}) \\
 - Addr &= r_{3b} G + R_{2b} + B_b
 \end{aligned}$$

Then set the $Addr$ as the challenge address and send it to the adversary $\mathcal{A}$.

- **Guess.** The adversary $\mathcal{A}$ outputs a guess b or $\perp$. $\mathcal{A}$ wins the Game_{UNL} if $b' = b$.

If the adversary $\mathcal{A}$ could win the Game_{UNL} , the adversary $\mathcal{A}$ could find which public key is used to generate the one-time address. However, the $Addr$ is generate by r_{3b} . The r_{3b} is a random number generate from the random oracle and the adversary could not identify the difference between the random and the r_{3b} . Therefore, the adversary $\mathcal{A}$ need the correct r_{3b} respond from the challenger to compute the $Addr$ to find the relation. This means that the adversary had query $G_{2j} (j \in \mathbb{Z}_{q_H})$ before to get the r_{3b} , and the $G_{2j} = e(R_{2b}, A_m)^{r_1}$. However to find the correct $G_{2j} = e(R_{2b}, A_m)^{r_1}$, the adversary $\mathcal{A}$ need the correct R_{2b} and r_1 . For getting the correct R_{2b} , the adversary need to query $G_{1k} (k \in \mathbb{Z}_{q_H})$ before and $G_{1k} = h_1(r_1 A_b)$. Hence, to find the correct G_{1k} need to find the correct r_1 . The probability to find the correct r_1 is $1/n$. Therefore, the probability that the adversary $\mathcal{A}$ to identify the real identity of the one-time address is $1/n$. The probability is negligible.

Theorem 2 *Our traceable one-time address scheme is secure against one time privilege escalation attack. For all PPT adversaries $\mathcal{A}$, the probability of $\mathcal{A}$ to forge out the correct secret key is $1/n$, n is the order of the $\mathbb{Z}_n$.*

Proof Assume that there exists a polynomial-time adversary $\mathcal{A}$ who can successfully implement a privilege escalation attack when only one derived one-time address secret key is compromised. This means that the adversary could successfully forge the secret key.

- **Setup.** The challenger execute the *Setup* and *KeyGen* to generate $PP = ((n, \mathbb{G}_1, \mathbb{G}_2, e, G), h_1, H, \mathbb{Z}_n)$ and $SK_r = (a_r, b_r)$, $PK_r = (A_r, B_r)$, $PK_m = (A_m, B_m)$. n is the order of the $\mathbb{Z}_n$. Then the challenger sends PP, PK_r, PK_m to the adversaries $\mathcal{A}$.
- **Challenge.** In this phase, the adversary $\mathcal{A}$ randomly choose $r_{1i} \in \mathbb{Z}_n (i \in \{0, 1\})$ and execute *OnetimeAddrGen*(PK_r, PK_m) as follows:

$$\begin{aligned}
 - R_i &= r_{1i} G \\
 - r_{2i} &= h_1(r_{1i} A_r) \\
 - R_{2i} &= r_{2i} A_r \\
 - r_{3i} &= H(e(R_{2i}, A_m)^{r_{1i}}) \\
 - Addr_i &= r_{3i} G + R_{2i} + B_r
 \end{aligned}$$

and record $r_{2i}, r_{3i} (i \in \{0, 1\})$ that use to generate the one-time address. The adversary $\mathcal{A}$ then send $Addr_0$ and $Addr_1$ to the challenger. The challenger executes $OnetimeSKGen(Addr_0, SK_r)$ to get (dsk_0, r_{20}, r_{30}) and send the dsk_0 back to the $\mathcal{A}$.

- **Output Phase.** $\mathcal{A}$ outputs a forgery secret key a'_r, b'_r . $\mathcal{A}$ wins the Game_2 if $(a'_r, b'_r) = (a_r, b_r)$.

The adversary only knows one derived secret key dsk_0 respond by the challenger, so the adversary $\mathcal{A}$ only has one polynomial equation $dsk_0 = r_{30} + r_{20}a_r + b_r$, but there are two unknown variables a_r, b_r . If the adversary want to find the correct answer of a_r, b_r . He need two equations. Therefore, adversary $\mathcal{A}$ could find n sets of possible answer but doesn't know which is the correct one. Hence, the only chance to win the game is to guess correctly, and the probability is $1/n$.

6 Comparison

This section discusses the properties of our traceable one-time address scheme and compares them with other related schemes. The properties of our proposed scheme are as follows.

- Privilege escalation attack resistant: In our scheme, we can resist privilege escalation attack one time.
- Traceability: True anonymity cryptocurrency like Zcash [10] and Monero [7] have often been used in illegal activities. Therefore, our scheme provide traceability, allowing us to trace back to the real identity of a one-time address to solve the problem of lack of regularity.
- Accelerated Search Time: In a one-time address system, users typically identify their respective addresses by searching the blockchain and computing all recorded one-time addresses. This process can be time-consuming. To enhance efficiency, we've optimized the verification step by computing only R_2 instead of the entire address $Addr$. This modification leads to an almost 50% reduction in computational complexity.
- Without secret channel: In our proposed scheme, the one-time address does not require information about the sender's address. As a result, compared to Zhao's approach [14], users do not need a secure channel to transmit their public key to the receiver when generating a one-time address.

We implemented our traceable one-time address scheme using the C language for efficiency analysis. The experiment was conducted on a laptop equipped with an Intel(R) i5-11300H CPU clocked at 3.1GHz and 24GB of system memory (Tables 1, 2 and 3). In comparison with CryptoNote V2.0 [12], our one-time address generation time and derived secret key generation time are approximately twice as long. However, we offer traceability and security against privilege escalation attacks. At the same time, the search time remains nearly unchanged.

Table 1 Computational complexity

Scheme	$AddrGen$	$sk^{onetime}Gen$	Search	$Trace$
CryptoNote V2.0 [12]	3exp	1exp	1exp	x
Zhao's [14] first instance	4exp	2exp	4exp	1exp
Our Construction	4exp+e	2exp+e	2exp	1exp+e

e: bilinear mapping

Search: check an one-time address whether belongs to the user

Table 2 The comparisons between others and our scheme

Scheme	Support one-time address	Traceability	Secure against privilege escalation
CryptoNote V2.0 [12]	✓	✗	✗
Zhao's [14] first instance	✓	✓	✗
Our construction	✓	✓	Partially

Table 3 Experimental result(ms)

Scheme	$AddrGen$	$sk^{onetime}Gen$	Search	$Trace$
CryptoNote V2.0 [12]	1.38	1.11	1.1	x
Zhao's [14] first instance	2.42	2.24	2.5	1.2
Our construction	2.68	2.64	1.15	1.36

In comparison with Zhao's first instance [14], our address generation time and derived secret key generation time are only slightly longer, ranging from 0.2 to 0.4ms. However, our scheme is secure against privilege escalation attacks. Additionally, we have significantly expedited the search process, reducing the required time consumption. We only need half the time compared to Zhao's approach [14] for searching.

In our opinion, the slightly longer experimentation time required by our scheme is acceptable due to the additional tracing functionality and enhanced security environment it provides. Furthermore, we have significantly accelerate the search process, substantially reducing the time required to determine if a one-time address belongs to the user. The experimentation time should be feasible for most use cases since all of our algorithms can be executed within a mere 3 ms.

7 Conclusion and Future Works

This paper proposes a new traceable one-time address scheme, secure against privilege escalation attacks and capable of tolerating derived key leakage once. Furthermore, our scheme provides traceability and accelerates search time. We prove its security in the random oracle model, and experimental results demonstrate that time and space consumption is feasible for most use cases.

This paper only provides resistance against privilege escalation attack one-time. We aim to explore, in future research, a method that can both trace and completely withstand privilege escalation attacks. We believe that focusing on the signature aspect might offer a promising avenue for such developments.

References

1. Gutoski G, Stebila D (2015) Hierarchical deterministic bitcoin wallets that tolerate key leakage. In: International conference on financial cryptography and data security. Springer, pp 497–504
2. Lin C, He D, Huang X, Khan MK, Choo KKR (2020) Dcap: a secure and efficient decentralized conditional anonymous payment system based on blockchain. *IEEE Trans Inf Forens Secur* 15:2440–2452
3. Liu JK, Wong DS (2015) Linkable ring signatures: security models and new schemes. In: Computational science and its applications–ICCSA 2005: international conference, Singapore, May 9–12, 2005, Proceedings, Part II 5. Springer, pp 614–623
4. Liu Z, Yang G, Wong DS, Nguyen K, Wang H, Ke X, Liu Y (2021) Secure deterministic wallet and stealth address: Key-insulated and privacy-preserving signature scheme with publicly derived public key. *IEEE Trans Depend Secure Comput* 19(5):2934–2951
5. Miers I, Garman C, Green M, Rubin AD (2013) Zerocoin: anonymous distributed e-cash from bitcoin. In: 2013 IEEE symposium on security and privacy. IEEE, pp 397–411
6. Nakamoto S (2008) Bitcoin: a peer-to-peer electronic cash system. *Decentral Busin Rev*
7. Noether S, Mackenzie A et al (2016) Ring confidential transactions. *Ledger* 1:1–18
8. Reid F, Harrigan M (2013) An analysis of anonymity in the bitcoin system. Springer
9. Ron D, Shamir A (2013) Quantitative analysis of the full bitcoin transaction graph. In: Financial cryptography and data security: 17th international conference, FC 2013, Okinawa, Japan, April 1–5, 2013, Revised selected papers 17. Springer, pp 6–24
10. Sasson EB, Chiesa A, Garman C, Green M, Miers I, Tromer E, Virza M (2014) Zerocash: decentralized anonymous payments from bitcoin. In: 2014 IEEE symposium on security and privacy. IEEE, pp 459–474
11. Sun SF, Au MH, Liu JK, Yuen TH (2017) Ringct 2.0: A compact accumulator-based (linkable ring signature) protocol for blockchain cryptocurrency monero. In: Computer Security–ESORICS 2017: 22nd European symposium on research in computer security, Oslo, Norway, September 11–15, 2017, Proceedings, Part II 22. Springer, pp 456–474
12. Van Saberhagen N (2013) Cryptonote v 2.0
13. Yin X, Liu Z, Yang G, Chen G, Zhu H (2022) Secure hierarchical deterministic wallet supporting stealth address. In: European symposium on research in computer security. Springer, pp 89–109
14. Zhao L, Zhong L, Zhang J (2023) Traceable one-time address solution to the interactive blockchain for digital museum assets. *Inf Sci* 625:157–174

An Object-Based Multi-level Authentication Framework for AR e-Book



Chia-Chen Lin, Aristophane Nshimiyimana, Wei-Liang Tai,
and Morteza Saberikamarposhti

Abstract In contrast to traditional electronic books, augmented reality books provide readers with the chance to explore supplementary information or multimedia content associated with the material they are reading. Unlike typical electronic books, augmented books encompass a wider range of content levels. This highlights the necessity for a specialized authentication framework designed to handle the diverse levels of content found in augmented books. Within this paper, we introduce an innovative, object-based, multi-level content authentication framework crafted to meet practical requirements. Its purpose is to aid lawful users in confirming the legitimacy of the augmented books they are perusing, effectively distinguishing between legally copyrighted editions and pirated reproductions. Finally, the conventional LSB substitution method is employed to demonstrate the functionality and applicability of our proposed object-oriented, multi-level content authentication framework.

Keywords Data hiding · Augmented reality · Multi-level content authentication · Object-based authentication

C.-C. Lin (✉) · A. Nshimiyimana
National Chin-Yi University of Technology, 411030 Taichung, Taiwan
e-mail: ally.cclin@ncut.edu.tw

A. Nshimiyimana
e-mail: aristophanenshimiyimana@gmail.com

W.-L. Tai
National Taichung University of Science and Technology, 404336 Taichung, Taiwan
e-mail: twl@nutc.edu.tw

M. Saberikamarposhti
Multimedia University, 63100 Cyberjaya, Selangor, Malaysia
e-mail: msaberyk@ieee.org

1 Introduction

Augmented reality, commonly referred to as AR, denotes an interactive encounter that blends computer-generated elements with the physical world. Any book that can activate 2D or 3D layers of content through a designated AR application, AR glasses, or a compatible device equipped with a camera qualifies as an AR book, regardless of whether it exists in printed or digital text format. Markers or triggers are placed strategically on certain printed pages or digital content of AR books. Once these embedded markers are recognized by the AR applications or the AR equipment mentioned above, the pre-designed digital content will be automatically projected in front of the readers. Usually, the projected content comprises 3D animations or videos, serving diverse objectives such as interactive games, quizzes, and other forms of educational and entertainment demonstrations.

Hence, an AR book can encompass not only multi-level content but also distinct 2D or 3D elements activated by embedded markers or triggers. Since these projected 2D or 3D objects often carry informative value, there is a risk of illicit duplication by malicious users. Without a robust copyright mechanism in place, these replicated 2D or 3D objects can be incorporated into other content, ultimately resulting in the creation of pirated AR books [1]. To provide an authentication framework and assist users in evaluating whether the AR book is legal or pirated, in this paper, we focus on AR e-books and an object-based multi-level content authentication framework is proposed. Finally, the conventional LSB substitution method is employed to demonstrate the function of our proposed object-oriented, multi-level content authentication framework.

The rest of this paper is organized as follows. Section 2 introduces background knowledge. Details of our proposed scheme are given in Sect. 3. Experiments and analyses are demonstrated in Sect. 4. Finally, Sect. 5 provides a conclusion.

2 Related Works

In order to ensure that our paper comprehensively covers the relevant background knowledge, we present the AR e-book framework and discuss various works pertaining to multi-level multimedia authentication and object-based authentication in Sects. 2.1 and 2.2, respectively. Through this exposition, we emphasize the essential nature of our proposed object-based multi-level authentication scheme.

2.1 Framework of AR e-Book

AR e-books are just a single category within the broader spectrum of AR applications. Among the various frameworks designed for different AR applications, the

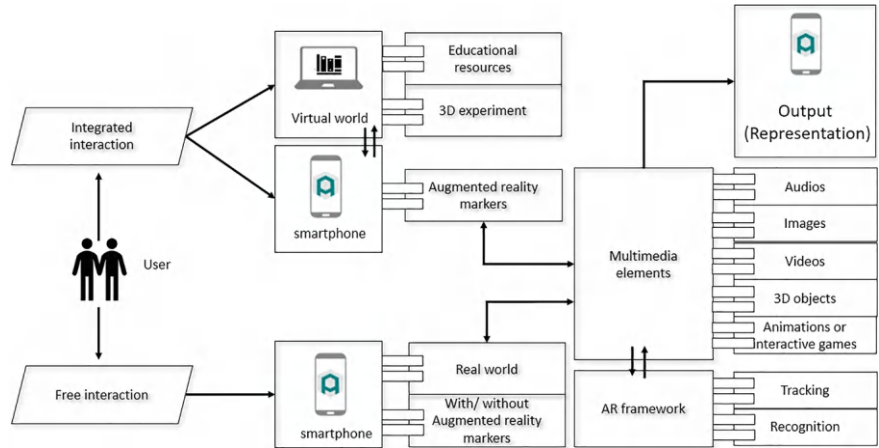


Fig. 1 Framework of AR applications, the AVATAR Project Book

AVATAR Project Book [2] framework as shown in Fig. 1 distinguishes itself as the most representative, primarily due to its integration of a wide range of multimedia elements, such as audios, images, videos, 3D objects, animations and simulations, and interactive scenarios. Here, the framework of AVATAR Project Book is used to demonstrate two possible interaction paths which have been frequently adopted in the AR book or AR e-book.

The first approach relies on personal computers to guide users in accessing the pre-built virtual lab. In this context, 3D expressions or 3D experiments can be incorporated, particularly when the content pertains to education. In contrast, the second approach, which involves mobile devices, does not offer 3D expressions or 3D experiments. This second approach can also be seen as AR books, where physical hardcopy books contain AR markers. Regardless of whether users choose the first or second path, they can access AR resources through mobile applications. Through the AR framework built up in mobile device applications, the embedded AR markers can be recognized and tracked.

In summary, as illustrated in Fig. 1, the structure of an AR e-book can be established based on the outlined framework. Subsequently, the script of the e-book can be defined for each sub-title, categorizing the corresponding content into various object types, including text, images, audio, video, animations, and interactive games. Moreover, it is also possible to establish connections among the specific objects belonging to each sub-title and their respective object types, as depicted in Fig. 2.

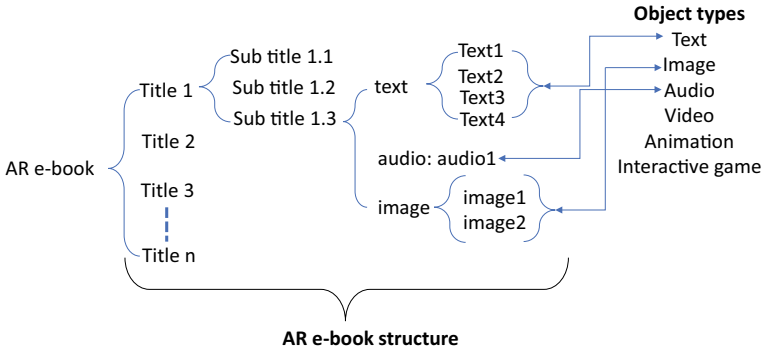


Fig. 2 Example of structure of AR e-book and linkage between the AR e-book and content objects

2.2 Multimedia Authentication

To authenticate the ownership or integrity of multimedia objects, multiple approaches have been designed, such as authentication, watermarking, hashing, or cryptographic techniques. In 2006, Lu et al. [3] designed a content-based image authentication scheme based on singular value decomposition. In their scheme, they consider the Green channel to be the most representative feature of a given image; therefore, they extract the image feature from the Green channel. First of all, they divided the Green channel of a color image into 8×8 non-overlapping blocks, and then they transformed the block into the coefficients of SVD. Finally, three features of an 8×8 block can be derived and then embedded into the corresponding RGB channels. Lu et al. generated block-based features for a color image and then embedded these features into the image itself so that the integrity of the color image can be verified. Experimental results confirm that their scheme can identify block-based attacks. In line with their concept, numerous image authentication schemes have been introduced. These authentication schemes can be categorized based on their detection scope into image-wide, block-based, and pixel-based [4, 5] methods, with some of them even incorporating recovery capabilities [6]. Either authentications or watermarking schemes, various schemes have been proposed for images, video [7], and audio [8].

To the best of our knowledge, in 2007, Lee et al. were the first to integrate invisible watermarking into the design of an invisible marker tracking system for AR applications [9]. Lee et al. opted for invisible watermarking with the goal of concealing markers without requiring additional equipment. In their proposed scheme, users perceive only real-world images, while the hidden invisible markers can only be interacted with by mobile device applications. Eventually, 3D objects are displayed once the invisible marker is detected and recognized. Lee et al.'s approach leveraged invisible watermarking to enhance the invisibility of concealed markers, rendering AR e-books more practical. However, they did not address measures to prevent

the unauthorized duplication of these concealed, invisible markers or establish an authentication framework to make it difficult to replicate authenticated AR objects.

3 Proposed Object-Based Multi-level Authentication Framework for AR: OML-AFAR

As we mentioned in Sect. 2.1, the structure of an AR e-book and the linkage between object type and object content designed for each sub-title can be built up as shown in Fig. 2. After integrated with AR technology, AR tag(s) are added into the AR e-book structure, as shown in Fig. 3a, to indicate which objects will be triggered by the markers. However, to correctly embed the invisible markers proposed by Lee et al. [9] and other approaches, the script describes the sequence of each involved object is required. Take Fig. 3 for example, according to the AR(s) tag defined in the AR e-book structure, it is known that object audio1 will be triggered by an AR tag, as for its possible position, it has two options depending on whether it is visible or invisible. If the AR tag related to audio1 is visible, it is easy to proceed. If the AR tag related to audio1 is invisible, the potential carrier will be located to the object listed before audio1, which will be text3 based on the script shown in Fig. 3b.

To ensure easy recognition of visible markers, one can employ visible logos or enlarged text to indicate their positions, as shown in visible marker 1 or visible marker 2 in Fig. 4. In contrast, for the invisible marker, the possible area is first two or first three lines in the text3 marked in the blue frame in Fig. 4 if audio1 is related to text3.

To establish an object-oriented multi-level authentication framework, abbreviated as ORML-AFAR in the subsequent sections, four distinct phases must be undertaken: script generation for an AR e-book phase, authentication code generation phase,

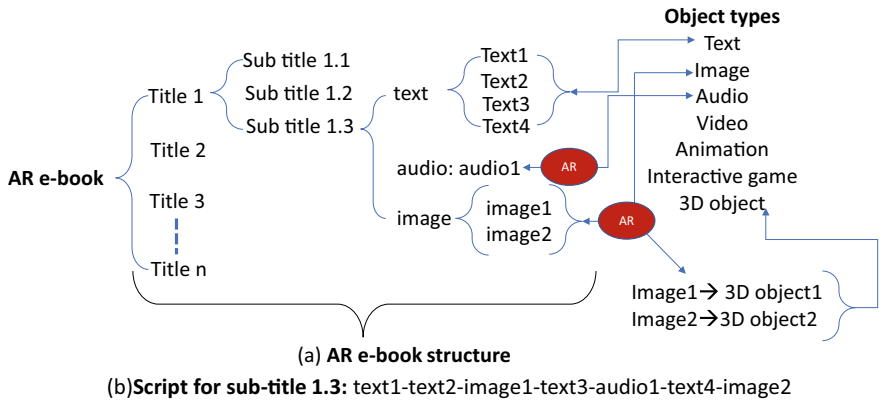


Fig. 3 Example of structure of AR e-book with AR tags, and the corresponding script

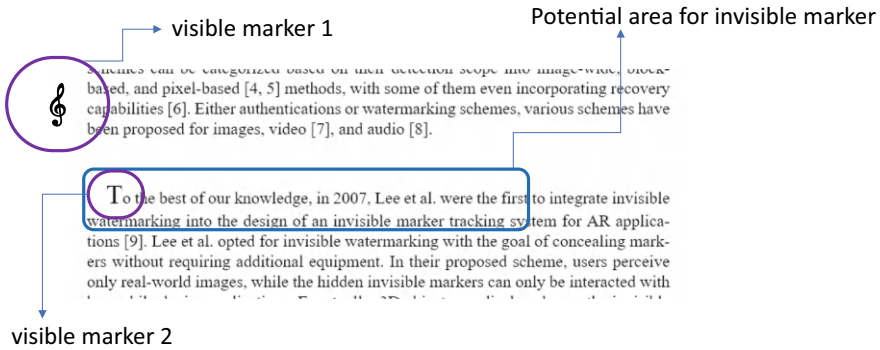


Fig. 4 Examples of potential positions of visible markers and invisible marker

authentication code embedding phase, and verification phase. Each of these phases will be elaborated upon in detail in the upcoming subsections.

3.1 Script Generation for an AR e-Book Phase

As we mentioned at the beginning of Sect. 3, an AR e-book may contain multiple objects belonging to different object types. Some objects will be triggered by embedded markers no matter those markers are visible or not. To link each object with the rest objects (i.e. texts, images, audio, videos, animations, and interactive games), the script for each sub-title must be built. Finally, a hierarchical structure must be pre-defined in advance so that an AR e-book's metadata can be derived.

3.2 Authentication Code Generation Phase

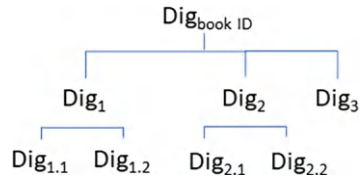
Once the hierarchical structure of an AR e-book is defined, the corresponding script for each sub-title along with the AR tags can be generated. Finally, SHA256 [10, 11] can be used to generate a digest for each subtitle called Dig for short. The Dig is served as the authentication code for the corresponding title (or sub-title)

$$\text{Dig}_{\text{title identifier}} = \text{SHA256}(\text{script of title identifier}), \quad (1)$$

where the script of the title identifier indicates the script along with the title number, such as script 1.1 means the script is mapped to sub-title 1.1 based on the example shown in Fig. 5, and the digest of script 1.1 is denoted as $\text{Dig}_{1.1}$ according to Eq. (1).

The same rule can be applied to generate an AR e-book's metadata digest as follows:

Fig. 5 Examples of authentication code (Dig)'s tree structure



$$\text{Dig}_{\text{book identifier}} = \text{Dig}_1 || \text{Dig}_2 || \dots || \text{Dig}_n, \quad (2)$$

Here, n represents the total number of titles within an AR e-book. If in a title, there are multiple sub-titles, then the Digs can form a tree structure as follows.

It is noted for an AR object, which means this object carries a marker to trigger a 3D object or the other type of object, the corresponding authentication code generated from the 3D object is also generated by Eq. (1) so that later this authentication can be used to authenticate the integrity of the triggered 3D object. Take Fig. 3 for example, image1 contains an AR tag, and 3D object1 will be triggered once the mark embedded in image1 has been detected and recognized. Therefore, 3D object 1's authentication code is also generated in this phase.

3.3 Authentication Code Embedding Phase

In our OML-AFAR, there is an authentication code for metadata for an AR e-book, an authentication code for either a sub-title or a title. Therefore, the authentication code will be embedded into the preface of an AR e-book, the embedding position for the next authentication code will be recorded after the hidden authentication code of metadata, as shown in Fig. 6.

It is noted that there are two scenarios for authentication code embedding, one is the general case as shown in Fig. 6a, where the digest for the metadata is located at the preface of an AR e-book, and the position for the next digest can be found after the



Fig. 6 Examples of Dig tracing path for an AR e-book

metadata's digest. The second scenario is specially designed for an image or text that contains a marker used to trigger a 3D object. In this case, the digest of 3D object has been generated by Eq. (1) in Sect. 3.2, and embedded into the corresponding image or text, as shown in Fig. 6b.

There are many data-hiding algorithms have been proposed in the last decade, such as LSB substitution-based data-hiding method [12], difference expansion [13], pairwise prediction-error expansion [14], and so on. Any of these algorithms can be considered a viable option for hiding a digest within digital content, provided that they adhere to the specific size constraints of the authentication code that needs to be concealed.

3.4 Verification Phase

After the first three phases have been completely conducted, the verification phase begins. Since the preface of an AR e-book contains the digest of the metadata, the verification phase is conducted from the preface page. It is noted that there are two scenarios for authentication code embedding, one is the general case as shown in Fig. 6a, where the digest for the metadata is located at the preface of an AR e-book, and the position for the next digest can be found after the metadata's digest. The second scenario is specially designed for an image or text that contains a marker used to trigger a 3D object. In this case, the digest of the 3D object should be generated by Eq. (1) in Sect. 3.2, and embedded into the corresponding image or text, as shown in Fig. 6b in the authentication code embedding phase as mentioned in Sect. 3.3.

For the first case, simplicity reigns when the mobile device application effectively retrieves the location containing the next authentication code. In such instances, Eq. (1) is once again utilized to generate the authentication code, using the script for the title or sub-title currently undergoing verification as a basis. For the second case, after the embedded marker is recognized by the mobile device application, the 3D object is retrieved and its authentication code is regenerated by Eq. (1) and then compared with the hidden authentication code. If they are equal, it indicates that the retrieved 3D object is related to the image which contains a marker to trigger the 3D object, and vice versa.

4 Experimental Results

In order to showcase the practicality and versatility of our OML-AFAR proposal, we have developed a prototype system using Python 3.9 and Unity with Vuforia. This prototype runs on a Mac OS-based PC equipped with a 2.9 GHz Dual-Core

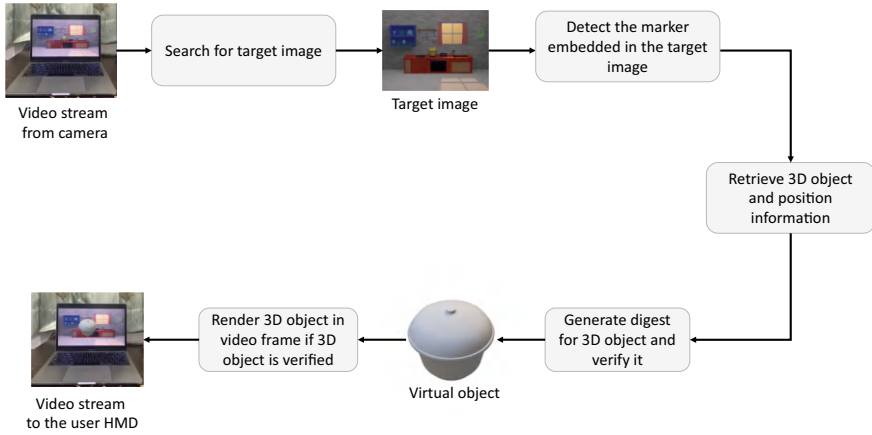


Fig. 7 Verification procedure for the second scenario presented in Fig. 6b

Intel® Core™ i5 CPU and is enhanced by an Intel Iris GPU for accelerated performance. With our implemented prototype system, the AR e-book system performs the verification procedure as shown in Fig. 7.

To test the image quality after concealing 3D object's digest, a frame of video is selected to calculate the corresponding *PSNR*.

$$\text{MSE} = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N (O_{i,j} - S_{i,j})^2, \quad (3)$$

$$\text{PSNR} = 10 \log_{10} \frac{(255)^2}{\text{MSE}} (\text{dB}), \quad (4)$$

where M and N are the height and width of the selected frame, respectively. O and S indicate the original frame and the stego frame, respectively (see Fig. 8).

Fig. 8 Example of an original frame selected from our AR e-book system



With the implemented AR e-book system, the performance of our proposed OML-AFAR is evaluated with two experiments. For the two scenarios mentioned in Sect. 3.4, the digest of the triggered 3D object must be derived by SHA256 in advance and then embedded into the corresponding image or text. Therefore, in the first experiment, we embed one 3D object’s digest into the corresponding image with LSB substitution and then we evaluate the image quality of the selected video frame. After concealing the digest, we can see the selected video frame still remains around 55 dB, which is above 30 dB and it indicates the human visual system is hard to distinguish the stego frame from the original selected frame.

In the second experiment, we substitute the image responsible for triggering a 3D object with a different image. This step allows us to evaluate the detection capabilities offered by OML-AFAR. To assess whether the authentication performance of OML-AFAR is influenced by various image processing techniques, we modify the test stego frame into a variant stego frame, as illustrated in Fig. 9a and b. The corresponding detection outcomes are presented in Table 1.

From Table 1, we can see that when the brightness is set as -40 and the contrast is set as -40 , the stego frame becomes darker compared with the stego image. In this particular scenario, the detection rate stands at 59%. However, when there is no modification, the detection ratio provided by our OML-AFAR is up to 100%.

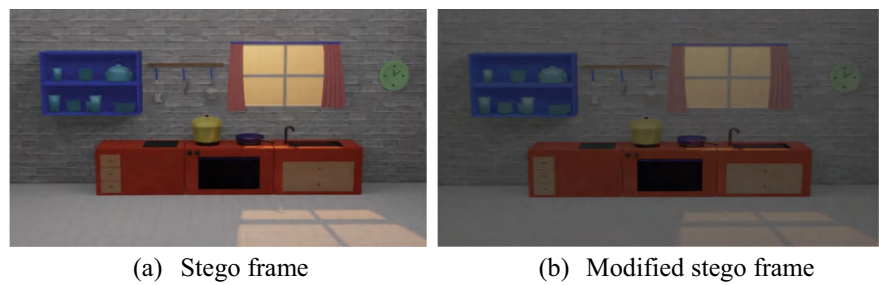


Fig. 9 Examples of **a** a stego frame and **b** a modified stego frame

Table 1 Detection results.
(B: brightness, C: contrast)

Frames	Parameters	Detection rates (%)
Frame 0	B: 0 C: 0	100
Frame 3	B: -40 C: -40	59

5 Conclusions

Augmented reality (AR) technology has reached a high level of maturity, finding applications in entertainment and education. To safeguard the integrity of informative AR objects or 3D objects and prevent unauthorized duplication, we introduce an object-based multi-level authentication framework for Augmented Reality, abbreviated as OML-AFAR, in this paper.

Our approach begins with script generation based on the hierarchical structure of an AR e-book. Subsequently, we design a digest derived from SHA256 for specified titles or sub-titles. Finally, the authentication code embedding phase is executed using any data-hiding method, provided its payload size is equal to or greater than that of the generated authentication code.

To enhance the protection of 3D objects triggered by hidden markers (visible or invisible), we introduce a second scenario where the digest of the 3D object can be directly embedded into the image or text carrying the markers. In our experiments, we evaluate not only the performance of our proposed OML-AFAR but also its detection capabilities under the second scenario. The results demonstrate that the detection rate reaches 100% when no modifications are made, but decreases to 59% when the stego frame is altered with a brightness setting of -40 and a contrast setting of -40 .

Based on these findings, we confirm the applicability of OML-AFAR. In the future, we intend to further test OML-AFAR with different data-hiding strategies to enhance detection rates under various attack scenarios.

References

1. Bogdanova EE (2020) Protection of intellectual property rights in the field of virtual and augmented reality. *Lex Russica* 73(7):86–96. (in Russ.) <https://doi.org/10.17803/1729-5920.2020.164.7.086-096>
2. AVATAR Project (2018) Official web page. Retrieved February, www.ufrgs.br/avatar
3. Lu T-C, Chang C-C, Liuc Y-L (2006) A content-based image authentication scheme based on singular value decomposition. *Pattern Recognit Image Anal* 16:506–522
4. Gul E, Ozturk S (2021) A novel pixel-wise authentication-based self-embedding fragile watermarking method. *Multimedia Syst* 27:531–545
5. Dahmane M (2023) A perceptual entanglement-based image authentication with tamper localisation. *Multimed Tools Appl*. <https://doi.org/10.1007/s11042-023-16791-y>
6. Chang Y-J, Wang R-Z, Lin J-C (2008) Sharing-based fragile watermarking method for authentication and self-recovery of image tampering. *EURASIP J Adv Signal Process* 17. <https://doi.org/10.1155/2008/846967>
7. Hariharalakshmi, Sivajothi M, Mohamed Sathik M (2023) Digital video watermarking in lower variance sub-bands of curvelet transform, vol 29. <http://cims-journal.com/index.php/CN/article/view/959>
8. Liu Z, Cao Y, Lin K (2023) A watermarking-based authentication and recovery scheme for encrypted audio. *Multimed Tools Appl*. <https://doi.org/10.1007/s11042-023-16032-2>
9. Lee HR, Shin JS, Hwang CJ (2007) Invisible marker tracking system using image watermarking for augmented reality. In: *Digest of technical papers international conference on consumer electronics*, Las Vegas, NV, USA. <https://doi.org/10.1109/ICCE.2007.341310>

10. Slavin O, Grin E (2022) Features of protection of intellectual property obtained using virtual and augmented reality technologies. *Society 5.0: Human-Centered Society Challenges and Solutions*, pp 103–113
11. Sobti R, Geetha G (2012) Cryptographic hash functions: a review. *IJCSI Int J Comput Sci Issues* 9(2):461–477
12. Wu M-N, Lin M-H, Chang C-C (2004) A LSB substitution oriented image hiding strategy using genetic algorithms. In: *AWCC 2004. Lecture notes in computer science*, vol 3309. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-30483-8_27
13. Tian J (2003) Reversible data embedding using a difference expansion. *IEEE Trans Circuits Syst Video Technol* 13(8):890–896
14. Ou B, Li X, Zhao Y, Ni R, Shi Y-Q (2013) Pairwise prediction-error expansion for efficient reversible data hiding. *IEEE Trans Image Process* 22(12):5010–5021
15. Wang R (2022) Application of augmented reality technology in children's picture books based on educational psychology. *Front Psychol* 13. <https://doi.org/10.3389/fpsyg.2022.782958>

A Door Lock System Based on Visual Cryptography



Chang-Yu Chao, Jen-Chun Chang, Ching-Cheng Wang, Chi-Sheng Chen,
Hao-Hsuan Chung, Kuan-Han Lu, and Guan-Ying Huang

Abstract A Visual Cryptography Scheme (VCS) is a unique data hiding technique aimed at decomposing secret information into multiple images, referred to as shares. These individual images, when viewed in isolation, are incapable of revealing the contents of the secret. The original secret information can only be reconstructed when multiple images are overlaid. The distinguishing feature of this method lies in the distribution of these shares. They individually appear as random noise. It ensures that access to any part of the secret information is only possible when all shares are combined. In this paper, we propose a Visual Cryptography Scheme enhanced with surface cover and apply it to door lock systems, effectively enabling efficient identity management and verification. Through this technology, the system can guarantee that access permissions are granted only to authorized individuals.

Keywords Visual cryptography · Data hiding · Door lock · Identity verification

1 Introduction

In modern life, door locks and security systems play a crucial role in protecting our property and privacy, Pavelić et al. [1] have conducted research on the potential risks associated with smart door locks in the Internet of Things. However, the traditional physical key unlocking method comes with a series of problems and challenges. These issues include the complexity of the unlocking process, the risk of losing keys, and concerns about security.

For many, carrying a bunch of keys can be cumbersome, with each door requiring a specific key. This leads to the time-consuming task of searching through a sea of keys, resulting in inconvenience.

Moreover, the risks and security issues of traditional keys cannot be ignored. For example, if keys are lost and accidentally fall into the hands of criminals, it might

C.-Y. Chao · J.-C. Chang (✉) · C.-C. Wang · C.-S. Chen · H.-H. Chung · K.-H. Lu · G.-Y. Huang
National Taipei University, New Taipei City, Taiwan
e-mail: qwasqwas8520@gmail.com

lead to loss in property and personal safety. The risk of keys being duplicated and data being stolen also exists.

To address the shortcomings of traditional keys, we need a more advanced and secure method of unlocking. Aswini et al. [2] proposed an RFID-based door locking system, while Mu et al. [3] proposed a technology that utilizes Bluetooth for controlling smart locks. Jahnavi and Nandini [4] propose the use of Raspberry Pi in combination with facial recognition for unlocking. In this paper, we propose an innovative door locking system called the Visual Cryptography door lock system. This system utilizes Visual Cryptography as its core, generating user share images based on the door lock Secret image, door lock share images, and the user's name. These user share images serve as keys, allowing for effortless door unlocking through a two-way scanning mechanism, eliminating the need for physical keys. Similarly, for identity verification using visual cryptography, Das and Ganguly [5] collect two distinct biometric features, merge them into a single image, and then split the image into two share images for subsequent identity verification.

Our door lock system offers numerous advantages. Firstly, it eliminates the complications tied to using physical keys, simplifying and speeding up the unlocking process. There is no need for users to carry physical keys, thereby drastically reducing the chances of losing them. Furthermore, the system comes with an automatic key finding feature, saving users from the trouble of searching for the right key. More importantly, it ensures that only authorized persons can unlock the door, bolstering security.

The first section is the introduction. The subsequent sections of the paper are organized as follows: In Sect. 2, we present the related works, delving into the methods of Original Visual Cryptography and Visual Cryptography with Cover. Section 3 offers a detailed description of our proposed algorithm. Section 4 introduced the design of the door lock system. The unlocking process is elaborated in Sect. 5. Applications are introduced in Sect. 5.1. Finally, Sect. 5.2 concludes the paper.

2 Related Works

Naor and Shamir [6] were pioneers in the field of visual cryptography, introducing both the original concept and the idea of steganographic visual cryptography. Hwang and Chang [7] proposed a scheme where two shares can be generated using any two distinct meaningful images and one secret image. In the realm of identity verification, Bhase and Mangrulkar [8] proposed an identity authentication scheme utilizing a visual cryptography framework, they propose a model that categorizes the system into various permission levels, granting distinct privileges to users. Huang [9] utilized one-dimensional codes for identity verification in banks and co-branded enterprises, utilizing the principles of visual cryptography. Oo [10] utilized QR codes for registration and identity authentication in online banking. Annappanavar and Chavan [11] used visual cryptography for fingerprint matching in voter identity verification. In

this paper, we primarily expand upon the research theories of Naor and others, as well as Chang, Hwang, for further discussion. Details will be elaborated below.

2.1 Original Visual Cryptography

The method introduced by Naor and Shamir [6] in 1994 is a data hiding technique that uses the visual attributes of images to safeguard confidential information. The main idea is to divide the secret information into several share images, each assigned to a participant. Each share image, when viewed individually, reveals no hidden secret; it only becomes meaningful when participants come together and overlay their share images to reconstruct the original secret information. The individual share images seem to be just random patterns, resembling noise to the naked eye, as shown in Fig. 1.

The advantages of visual cryptography lie in the fact that the shared images themselves do not contain any information about the secret, thus enhancing security. Even if one share is stolen, an attacker cannot gain any knowledge about the secret. Furthermore, visual cryptography allows for easy sharing and simple decryption, removing the need for complex cryptographic processes.

To achieve this, each pixel of the secret message is divided into 2×2 subpixels. Based on the principle of dithering, when the proportion of black subpixels exceeds white subpixels, the human eye perceives the image as darker. As shown in Fig. 2, if the pixel is white, Share1 is randomly selected with equal probability, while Share2 depends on the selection made in Share1, resulting in four possible choices. If the pixel is black, Share1 is also randomly selected, but Share2, based on Share1's choice, cannot have any overlap, leading to only one choice.

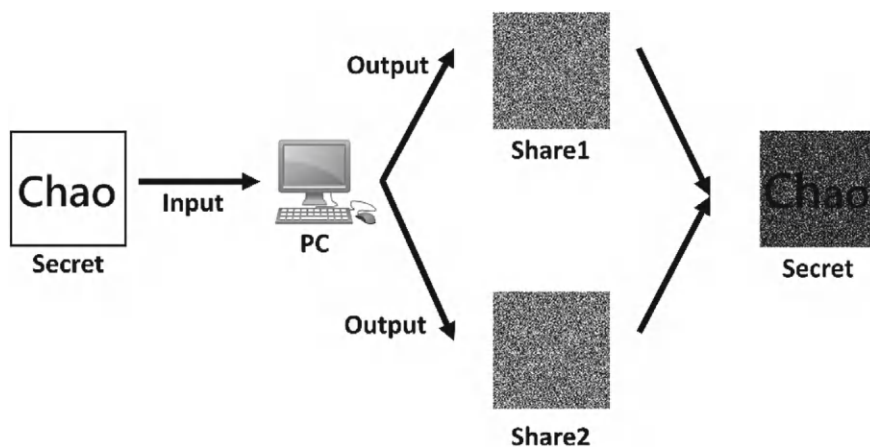


Fig. 1 Original visual cryptography scheme

Secret	Share1	Share2			
 White					
					
					
					
					
					
 Black					
					
					
					
					
					

Fig. 2 Pixel combination table

2.2 Visual Cryptography with Cover

Visual Cryptography with cover is an improvement over traditional Visual Cryptography, addressing the issue of share images appearing as meaningless noise and being difficult to distinguish among different users. Developed by Hwang and Chang [7], Visual Cryptography with cover enhances the share images by embedding concealed information within the noise.

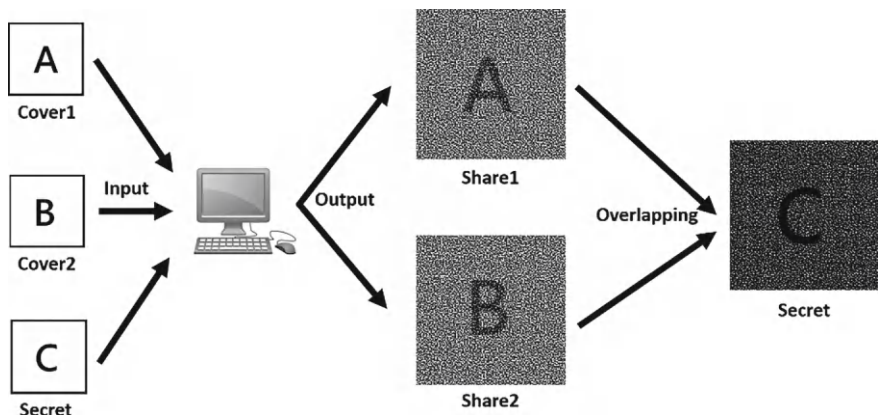


Fig. 3 Visual cryptography with cover schematic

The fundamental concept involves preparing a secret image (Secret) and two cover images (Cover1 and Cover2). These images are processed by a computer to generate two share images (Share1 and Share2). Share1 visually resembles Cover1, and Share2 similarly appears like Cover2. When the two share images are overlaid, the surface information vanishes and the hidden secret image reveals, as shown in Fig. 3.

To reveal a white pixel in the secret image after overlaying, it must have a result of $\frac{3}{4}$, and for a black pixel, it must be $\frac{4}{4}$. Therefore, the original arrangement before overlaying is set as $\frac{2}{4}$ for white pixels and $\frac{3}{4}$ for black pixels. As shown in Fig. 4, in this table, 0 represents white pixels and 1 represents black pixels, resulting in eight possible combinations to show the secret. For example, when we aim to reveal a black pixel by overlapping a white pixel with another white pixel, there are six choices available for Share1. The choice for Share2, however, will be determined based on the content of Share1, leaving only one option. Similarly, to reveal a white pixel by overlapping black with white, there are four options for creating Share1, with Share2 having three possible options based on the content of Share1.

Taking Fig. 4 as an example, when Secret, Cover1, and Cover2 are (1 , 1 , 1 ) respectively, Share1 will have the following four choices (, , , ). If Share1 chooses , then Share2 will have the following three choices (, , ).

Share 1 will offer varying alternatives contingent on Secret, Cover 1, and Cover 2, while Share 2's options will depend on the choices made in Share 1.

Secret	Cover1	Cover2	Share1	Share2
 0	 0	 0	  	  
 0	 0	 1	  	 
 0	 1	 0	  	  
 0	 1	 1	  	
 1	 0	 0	  	
 1	 0	 1	  	 
 1	 1	 0	  	  
 1	 1	 1	  	  

Fig. 4 Secret combination relationship table

3 Our Visual Secret Hiding Algorithm

In order to address the various inconveniences associated with traditional keys, we propose an algorithm based on this related work. This algorithm effectively achieves the objectives and reduces data storage space. The method is as follows.

3.1 Algorithm for Visual Cryptographic Door Lock System

In visual cryptography scheme, it is possible to randomly generate subpixels for either Share1 or Share2. The first generated Share (either Share1 or Share2) has more freedom in random selection, while the one generated later has less. However, in the research proposed in this paper, we aim to reduce the number of shares required for the door lock system to minimize the storage space needed for share images. We have modified and applied the visual cryptography scheme algorithm to the door lock system. In this system, we first randomly select subpixels from the door lock share image and then generate subpixels for the user share image based on the door lock share image. The order of these two selections cannot be exchanged, so the door lock share image subpixels will have a higher “degree of freedom” in the resulting shares.

In the door lock system algorithm, we divide the secret message into two share images, one for the door lock and one for the user. Both the door lock share image and the user share image appear to contain information on the surface. In the setup process, we first establish the door lock system, then add users associated with that particular door lock. When setting up the door lock, we first create a unique share image and a secret message for the door lock. When adding a user, we generate the user’s Share image based on the door lock Share image and the door lock’s secret message, as shown in Fig. 5.

Taking Fig. 5 as an example, when Secret, Door Cover, and User1 Cover are (0, 0, 0) respectively, there are six possible choices for Door Share (, , , , ,). If Door Share is chosen as , User1 Share will have four possible choices (, , ,).

Users can generate multiple user share images. However, storing each user share image along with the door share image would consume a significant amount of

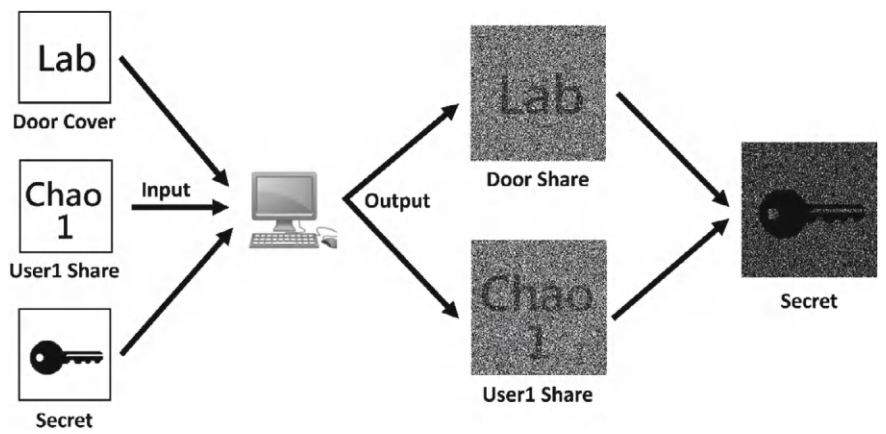


Fig. 5 Door lock system generates share relationship diagram

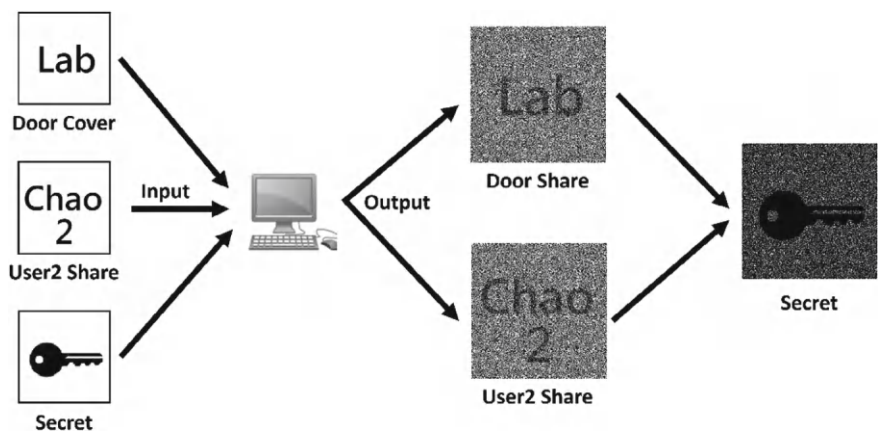


Fig. 6 Simply by changing the user cover

storage resources. As mentioned above, door share image has a higher degree of freedom, while user share images have a lower degree of freedom. Still, it doesn't mean there is no freedom; other subpixel combinations are still possible. As shown in Fig. 6, if the user needs to add a second, third, or more user shares, in our approach, it is only necessary to keep the door lock fixed and simply change the user cover image. In the end, the secret can be revealed through overlapping.

In the example from Fig. 6, when adding user2, there's no need to create new secret, door cover, or door share, as they remain the same as those in Fig. 5. Only the user2 cover needs to be changed. When Secret, Door Cover, and User2 Cover are set to $(0 \blacksquare, 0 \blacksquare, 1 \blacksquare)$, Door Share has the same six options as before $(\blacksquare, \blacksquare, \blacksquare, \blacksquare, \blacksquare, \blacksquare)$. If Door Share chooses option $\blacksquare$, User2 Share will have two options $(\blacksquare, \blacksquare)$.

In the Visual Cryptography door lock system, three $n \times n$ images are required as inputs: a secret image, a user cover image, and a door lock (DL) cover image. Using the algorithm, two new images are generated. The original $n \times n$ image undergoes pixel expansion, resulting in a $2n \times 2n$ user share and a $2n \times 2n$ door lock share. The process is further explained in the pseudocode presented in Fig. 4.

Algorithm 1 A share image generation algorithm**Input:** $n \times n S, n \times n C^1, n \times n C^2$.**Output:** $2n \times 2n H^1, 2n \times 2n H^2$.**for** $i=1$ to n^2 **do** Get S_i from Secret image; Get C_i^1 from User cover image; Get C_i^2 from DL cover image; **if** $(S_i, C_i^1, C_i^2)=(0,0,0)$ **then** $H_i^1 \in_R \{0011, 0101, 0110, 1001, 1010, 1100\}$ Based on the choice made by H_i^1 , H_i^2 will have 4 different options. **else if** $(S_i, C_i^1, C_i^2)=(0,0,1)$ **then** $H_i^1 \in_R \{0011, 0101, 0110, 1001, 1010, 1100\}$ Based on the choice made by H_i^1 , H_i^2 will have 2 different options. **else if** $(S_i, C_i^1, C_i^2)=(0,1,0)$ **then** $H_i^1 \in_R \{0111, 1011, 1101, 1110\}$ Based on the choice made by H_i^1 , H_i^2 will have 3 different options. **else if** $(S_i, C_i^1, C_i^2)=(0,1,1)$ **then** $H_i^1 \in_R \{0111, 1011, 1101, 1110\}$ Based on the choice made by H_i^1 , H_i^2 will have 1 different options. **else if** $(S_i, C_i^1, C_i^2)=(1,0,0)$ **then** $H_i^1 \in_R \{0011, 0101, 0110, 1001, 1010, 1100\}$ Based on the choice made by H_i^1 , H_i^2 will have 1 different options. **else if** $(S_i, C_i^1, C_i^2)=(1,0,1)$ **then** $H_i^1 \in_R \{0011, 0101, 0110, 1001, 1010, 1100\}$ Based on the choice made by H_i^1 , H_i^2 will have 2 different options. **else if** $(S_i, C_i^1, C_i^2)=(1,1,0)$ **then** $H_i^1 \in_R \{0111, 1011, 1101, 1110\}$ Based on the choice made by H_i^1 , H_i^2 will have 3 different options. **else if** $(S_i, C_i^1, C_i^2)=(1,1,1)$ **then** $H_i^1 \in_R \{0111, 1011, 1101, 1110\}$ Based on the choice made by H_i^1 , H_i^2 will have 3 different options. **end if****end for**

Here, the symbols in the pseudocode represent the following: S represents the Secret image, S_i represents the i -th pixel in the Secret image, C^1 represents the User Cover image, C^2 represents the Door Lock Cover image, C_i^1 represents the i -th pixel in the User Cover image, C_i^2 represents the i -th pixel in the Door Lock Cover image, H^1 represents the User Share image, H_i^1 represents the i -th pixel in the User Share, $\in_R$ represents random selection from a set where each element has equal probability, H^2 represents the Door Lock Share image, H_i^2 represents the i -th pixel in the Door Lock Share image.

4 Our Proposed Door Lock System

4.1 System Architecture Diagram

The system architecture, as shown in Fig. 7, consists of four main components: the door lock, the user, the server, and the administrator. Each door lock contains a unique share image. Users can have multiple share images to unlock different doors.

When a user wants to add a key to unlock another door, they must submit a key request to the server. The server then sends this request to the administrator for approval. Once approved, the server creates a new user share image and sends it to the user. The user can save this user share image on their device. Subsequently, they can unlock the door using this image without an internet connection.

This system architecture enables an authorization mechanism between users and door locks. Users can unlock various doors by requesting keys from the server and obtaining the corresponding user share images. Additionally, the system can manage and record relevant data in the background, such as maintaining a blacklist of unauthorized users.

5 Unlocking Process

The unlocking process involves two parties: the user and the door lock. We designed the unlocking process as a “bidirectional scan,” which involves two main steps: first, the user scans the door lock, and then the door lock scans the user.

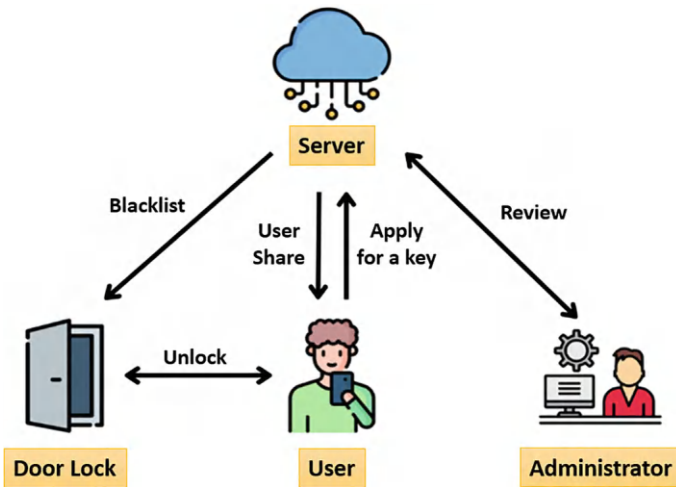


Fig. 7 System architecture diagram

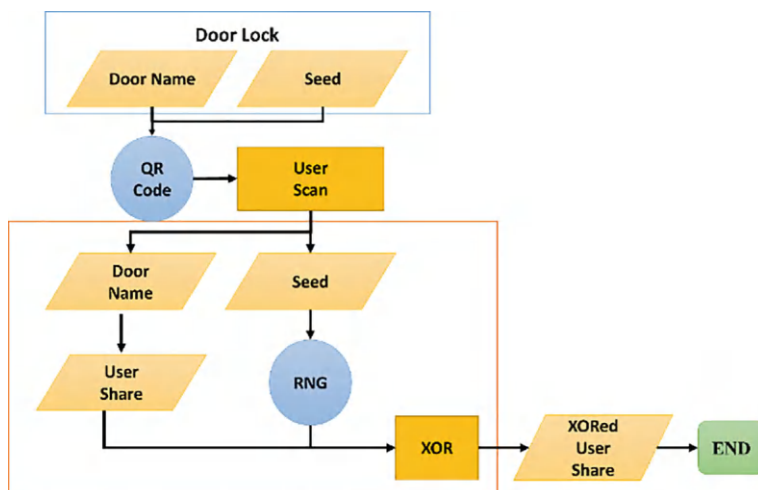


Fig. 8 Scanning the door lock share illustration

5.1 The User Scanning the Door Lock Share

Figure 8 shows the flowchart for the first step, where the user scanning results in obtaining two pieces of information. The first is the door QR code, which is used to automatically identify the corresponding user key. The second is a time-sensitive random seed, denoted as “Seed,” which is XOR-encrypted with the user’s Share image to enhance key replication difficulty.

5.2 The Door Lock Scanning User Share

Figure 9 illustrates the process of the second step. After scanning, the door lock obtains a XORed user share image, which we decrypt using a random seed to obtain the user share image. First, we check whether this user Share image is in the blacklist. If it is not in the blacklist, we overlay the user share image with the door lock share image. If the overlaid image is the secret image, the unlocking is successful; otherwise, it fails.

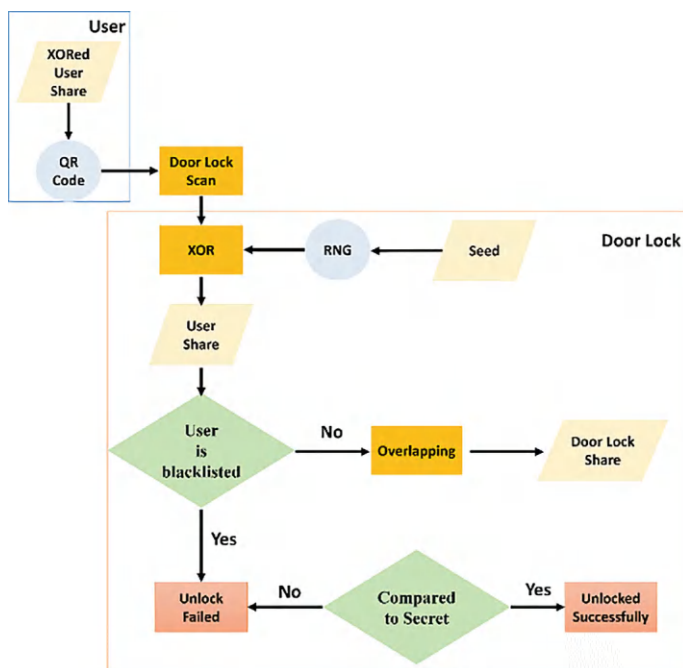


Fig. 9 Scanning the user share illustration

Acknowledgements This research is supported by National Science and Technology Council, Taiwan under contract 110-2221-E-305-005-MY2.

References

1. Pavelić M, Lončarić Z, Vuković M, Kušek M (2018) Internet of Things cyber security: smart door lock system. In: International conference on smart systems and technologies (SST)
2. Aswini D, Rohindh R, Manoj Ragavendhara KS, Mridula CS (2021) Smart door locking system. In: International conference on advancements in electrical, electronics, communication, computing and automation (ICAECA)
3. Mu Z, Li W, Lou C, Liu M (2020) Investigation and application of smart door locks based on Bluetooth control technology. In: Asia-Pacific conference on image processing, electronics and computers (IPEC)
4. Jahnvi S, Nandini C (2019) Smart anti-theft door locking system. In: 1st international conference on advanced technologies in intelligent control, environment, computing & communication engineering (ICATIECE)
5. Das A, Ganguly N (2022) An efficient visual cryptography approach for intermodal biometric verification. In: International conference on smart generation computing, communication and networking (SMART GENCON)
6. Naor M, Shamir A (1995) Visual cryptography. In: De Santis A (ed) Advances in cryptology—EUROCRYPT’94. Lecture notes in computer science. Springer, Berlin, Heidelberg

7. Hwang R-J, Chang C-C (2001) Hiding a picture in two pictures. *Opt Eng*
8. Bhas G, Mangrulkar RS (2018) An access control system using visual cryptography and steganography. In: Fourteenth international conference on information processing (ICINPRO)
9. Huang Y-C (2022) Design of a secure bar code system for banks and co-branded enterprises based on visual cryptography
10. Oo LL (2023) Development of online banking system based on secure captcha image using visual cryptography. In: IEEE conference on computer applications (ICCA)
11. Annappanavar S, Chavan P (2022) (2,2) visual cryptography based biometric authentication mechanism for online elections. In: 2nd Asian conference on innovation in technology (ASIANCON)

Automatic APT Attack Reconstruction Supporting Lateral Movement



Chun-I Fan, Wei-Chen Wu, Cheng-Han Shie, Hsin-Nan Kuo, and Bo-Yi Lee

Abstract This work proposes a framework for generating datasets that allows users to adjust the APT attack techniques within it. The framework utilizes the MITRE ATT&CK framework to label the attack traffic based on the Tactics, Techniques, and Procedures (TTP), which facilitating researchers in using the datasets for model training. Meanwhile, this work uses the framework to reconstruct the APT29 attack process and collect a dataset to validate the framework's feasibility.

Keywords Advanced Persistent Threat (APT) · Threat intelligence · Dataset generation · The APT29 attack

1 Introduction

To address the limitations of traditional antivirus software in combating APT attacks, researchers are increasingly turning to machine learning as a potential solution. Machine learning relies on substantial amounts of relevant data to train models capable of accurately identifying anomalies.

C.-I. Fan · W.-C. Wu · C.-H. Shie · H.-N. Kuo (✉)

Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan

e-mail: bluedunk@gmail.com

C.-I. Fan

e-mail: cifan@mail.cse.nsysu.edu.tw

C.-H. Shie

e-mail: hanhan3927@g-mail.nsysu.edu.tw

B.-Y. Lee

National Center for High-Performance Computing, Hsinchu, Taiwan

e-mail: boyi@narlabs.org.tw

C.-I. Fan

Information Security Research Center, National Sun Yat-sen University, Kaohsiung, Taiwan

While there are several publicly available network or host log datasets, currently, only one known publicly available dataset is designed for collecting APT attacks. However, it is important to note that this dataset is not derived from real-world environments. This limitation arises from the fact that APT attack groups primarily target government organizations and large enterprises with high-value and sensitive information. Consequently, the disclosure of attack paths and related information to the public becomes challenging.

As a result, most researchers resort to simulating a limited number of APT attacks on their own and collecting data from the simulated victim hosts to train their detection models. This process consumes considerable time and effort to generate data that serves a singular purpose and is difficult to effectively apply in real-world environments due to its limited size.

Furthermore, APT attacks frequently update their techniques and tools within short timeframes. This dynamic nature makes it challenging to capture the entirety of APT attacks using a single dataset alone. As a result, researchers face difficulties in keeping up with the rapid evolution of APT attack methods and staying up-to-date with the latest attack vectors and strategies.

To address the current challenges, this work proposes a framework for generating datasets. The challenges in this context encompass two main aspects. Firstly, there is a lack of sufficient datasets available for training APT attack detection models. Secondly, existing datasets suffer from the issue of containing outdated APT attack techniques, primarily due to limited disclosure of relevant information.

The framework aims to overcome these challenges by providing a solution for generating comprehensive and up-to-date datasets. This framework allows users to customize attack techniques to construct complete attack chains and simulate multiple Windows devices and Active Directory (AD) environments as victim hosts. After collecting system logs from these compromised hosts, the attack traffic is labeled with TTPs from the MITRE ATT&CK framework.

This framework enables researchers to utilize the generated dataset for training APT attack detection models, thus enhancing the effectiveness and accuracy of their detection capabilities.

2 Related Works

Datasets play a critical role in machine learning and data analysis, serving as the foundation for training models when appropriately labeled and organized. In the field of intrusion detection and APT detection, datasets play a crucial role by providing valuable past experiences that enable informed judgments to be made. By utilizing historical data, analysts and researchers can develop effective models and algorithms that capture patterns and behaviors indicative of intrusions and advanced persistent threats.

These datasets provide valuable information, enabling the identification and classification of potential threats and facilitating the development of robust detection and mitigation strategies. As a result, datasets are heavily relied upon in these domains to enhance the accuracy and efficacy of intrusion and APT detection systems.

Here are a few introductions to some famous datasets.

2.1 Network Intrusion Detection Datasets

– ISCX [1]:

ISCX is a network intrusion detection dataset generated in a laboratory environment in 2012. The dataset was constructed to closely reflect the proportion of various packet types observed in real-world environments. The ISCX dataset covers attack types such as DoS, R2L, U2R, and Probe. It also includes some forms of injection attacks and backdoor implantation techniques.

Additionally, the dataset has multiple subdatasets, including ISCX-VPN2016 and ISCX-URL2016. These subdatasets respectively collect data related to virtual private network (VPN) usage and malicious URLs and phishing attacks.

The ISCX dataset can be utilized to train general intrusion detection models. However, it lacks the techniques commonly employed by modern APT organizations, making it less suitable for training APT attack detection models.

– UNSW-NB15 [2]:

UNSW-NB15 is a network intrusion detection dataset. It collects network traffic data generated in a laboratory environment from June to November 2015, including both normal and various types of attack traffic. The attack types covered in the dataset include DoS, R2L, U2R, and Probe, the four main attack categories. The main advantage of the UNSW-NB15 dataset is its inclusion of diverse attack traffic and normal traffic, enabling more realistic and comprehensive training and evaluation of intrusion detection models. The dataset is of a significant scale, containing millions of records, providing ample samples for training and evaluating intrusion detection models. Similar to the previous dataset mentioned, the UNSW-NB15 dataset is primarily used for training models for intrusion detection and network security analysis, but it differs from recent APT attack techniques and is not suitable for training APT attack detection models.

2.2 Datasets for APT Attacks

• CICIDS2017 [3]:

CICIDS2017 is a real network dataset that includes both network intrusion and

normal traffic. It consists of various types of attacks, including DoS, DDoS, botnet, worm, scanning, malware, etc. It also includes normal traffic and traffic data from different protocols and services. The dataset contains 80 network traffic features, including features based on flow statistics, Transmission Control Protocol (TCP), Internet Protocol (IP), and other protocols. Additionally, the dataset provides labels that can be used for model training and evaluation. However, it should be noted that the CICIDS2017 dataset contains relatively few APT attacks compared to other types of attacks. APT attacks represent only a small portion of the dataset, making it less suitable for training APT attack detection models.

- DAPT2020 [4]:
DAPT2020 is a dataset created specifically for APT attacks, encompassing various types of APT attack, such as external penetration, data exfiltration, and persistence. The dataset also simulates more hosts to achieve a more realistic environment. This dataset collected both network traffic packets and local system logs, which facilitate tracking the complete path of APT attacks. However, the simulated attack techniques in this dataset primarily come from APT attack reports between 2014 and 2018, which may not cover the continuously evolving forms of APT attacks. This dataset may not provide sufficient data for model training when confronted with more novel attack techniques.
- DARPA TC [5]
The DARPA Transparent Computing (TC) dataset was created specifically for APT attacks. It emphasizes that many organizations' closed environments provide favorable conditions for APT attack organizations to operate covertly. The project aims to construct a dataset to train models in identifying the correlations between behaviors across systems, enabling the tracing of attack origins. Multiple datasets with different scenarios are generated in the project, but each scenario mainly focuses on a series of related attack techniques. In this dataset, the detection targets are limited, and detailed labeling data for attack events is lacking, making it less accessible for other researchers to utilize.

Most of the datasets mentioned above have limited APT attacks, posing challenges for training relevant models. Several APT attack datasets created in specific scenarios lack proper labeling and have low reproducibility, limiting subsequent applications. The proposed dataset generation framework allows researchers to adjust the types and quantities of attack techniques according to their needs, enabling more effective utilization by researchers. Additionally, this work utilizes the framework to execute APT29 adversary simulations, resulting in a dataset that encompasses the complete APT life cycle and includes proper labeling.

3 The System Architecture

This chapter describes the architecture of the proposed system.

3.1 System Design

The proposed system architecture is shown in Fig. 1. The system will establish a simulated environment consisting of multiple hosts based on Windows 10 and a domain controller on Windows Server 2016 to simulate the victim organization. Additionally, an attack server based on Kali Linux will be set up for Caldera to write and orchestrate attack scripts. These attack scripts will be executed on compromised victim hosts, which will maintain normal user operations to ensure that the system logs on these hosts contain both malicious and normal activities. Subsequently, the system logs from the victim hosts will be collected, and the malicious activities in the log files will be labeled before being stored back in the database. Finally, a labeled dataset of APT attacks will be obtained. In the following sections, a more detailed description of the content will be provided.

3.2 Environment Overview

This section provides an overview of the simulated environment and the tools used in the research. The environment is hosted on a personal computer and includes multiple virtual machines and a domain controller to simulate a more realistic enterprise network configuration.

Vagrant is used to build the virtual environment required for the entire infrastructure, which includes the following four components:

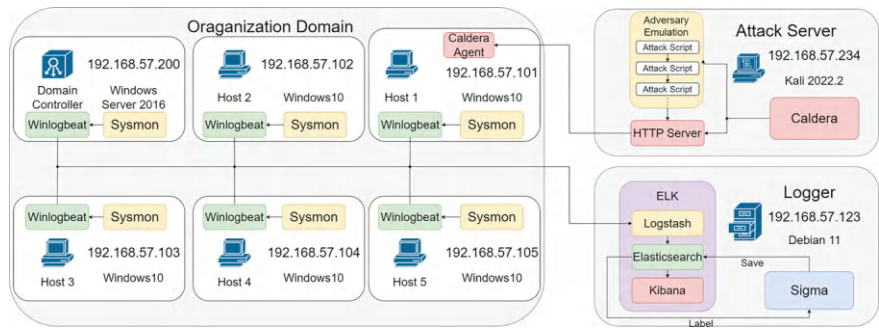


Fig. 1 The system architecture

- Host
- Domain Controller
- Attacker Server
- Logger.

In this experiment, the system sets up 5 hosts and 1 Domain Controller, which serves as the central management for the entire AD domain, to simulate the target organization. The configuration of multiple hosts helps us simulate the lateral movement of APT attacks within the organization. To facilitate the collection of system log files, the system also sets up a log collector, referred to as the logger, based on a Debian system within the same domain. This system is responsible for receiving and storing all system logs from the target organization. The collected logs are then labeled using the Sigma tool. In addition, an attack server based on the Kali Linux system is independently configured to simulate the attacks. This server is utilized to orchestrate the attack scripts and initiate simulated attacks on the target organization.

3.3 Data Generation

In an APT attack detection model, the dataset should include both malicious and benign data. It is because the model needs to learn the patterns of normal system behavior in order to differentiate abnormal behavior and potential attack activities. Benign data helps the model establish a baseline for normal behavior, thereby improving the accuracy of detecting abnormal behavior. Relevant studies have mentioned that maintaining a balanced ratio between these two types of data during model training is more effective [6]. Therefore, during this stage, the focus should be on generating both attack data and benign data.

Regarding the simulated attacks, the proposed method assumes that a host within the organization has already been compromised. This initial compromise could occur through methods such as phishing or backdoor implantation. It provides a foothold for external attackers to further penetrate the network and carry out their malicious activities. For generating attack data, the reliance is on the MITRE ATT&CK framework's automated penetration testing platform called Caldera [7]. It provides the capability to pre-implant backdoor programs into assumed victim hosts and monitor them through the platform's GUI interface. This work achieves a comprehensive attack simulation by developing attack scripts to perform specific techniques, such as data exfiltration from the victim host, tool transmission, and reconnaissance of the victim environment. Then, the Caldera platform is used to orchestrate the entire attack campaign, automating the process of passing the results of one action as inputs to the next, effectively simulating a complete attack chain.

Within the MITRE-developed automated penetration testing platform, there are legitimate commands available for system privilege management, compliance checks, and other operations. Therefore, we also utilize the platform to simulate some benign data, providing a means to automatically reproduce benign activities

within the dataset. However, since the proportion of benign data may be smaller in this simulation, and repeatedly simulating similar operations to compensate for data scarcity may not accurately reflect real-world usage. Therefore, in this work, we attempt to increase the realism of the environment by personally performing routine activities on the virtual machine, such as web browsing, installing software, and creating files.

3.4 Data Collection

The Sysmon system monitoring tool was installed on each host in the simulation environment to capture general user operations and external attack behaviors. And the tool Winlogbeat [8] was used to send the system logs collected on the hosts to the designated storage database which names Logger.

Winlogbeat is an open-source log collection tool developed by Elastic. Winlogbeat offers several advantages, including lightweightsness and high efficiency. This tool can be easily configured on each host with minimal resource consumption. It collects and transmits data with low latency, providing high real-time capabilities. As a result, Winlogbeat serves as a critical foundation for anomaly detection in scenarios that require real-time analysis. In this work, Winlogbeat is utilized for log collection and centralized log file management. Its primary function is to forward Sysmon log data from Windows systems to the ELK stack, where the data is processed and analyzed.

Logger is the host the system sets up as the database to store system logs from other hosts. On the Logger host, ELK and Kafka [9] were installed to assist in log file management and analysis. ELK provides various functions such as data storage, search, and data visualization, effectively assisting in organizing and analyzing the data.

3.5 Labelling

Labels serve as important criteria for training anomaly detection models and are essential components of the dataset. Therefore, after simulating and simulating attacks, the collected data underwent a labeling process. The following outlines the process and provides relevant descriptions of labeling in our system.

1. File format conversion and storage

Winlogbeat converts the Sysmon log files collected on the victim host from the EVTX format to the JSON format and sends them to the ELK system for storage.

2. Sigma rule creation

Sigma rules are used to describe specific behaviors and define them as threat scenarios. They help identify specific patterns of attack techniques.

3. Deploy Logstash

Logstash is an important component of ELK used for receiving and reading log files. It is employed to monitor the directory where Winlogbeat sends logs and establish a connection with Sigma. When a series of related log files is received, the transformer is triggered to perform labeling.

4. Labeling system

The labeling system identifies the behavior of log files based on YAML files describing threats, which are sigma rules. If the operations in the log files match the descriptions in the sigma rules, a label attribute is added to the log file. The label description is marked according to the relevant names defined in ATT&CK.

5. File storage

After labeling, the log files are returned to Elasticsearch for storage. The labeled log files can be extracted as a dataset or visualized using Kibana for further analysis.

4 Experimental Results

This chapter will describe the relevant details of creating the dataset. The dataset consists of a total of 9305 events, and the malicious operation logs among them are labeled based on the TTPs defined by MITRE ATT&CK.

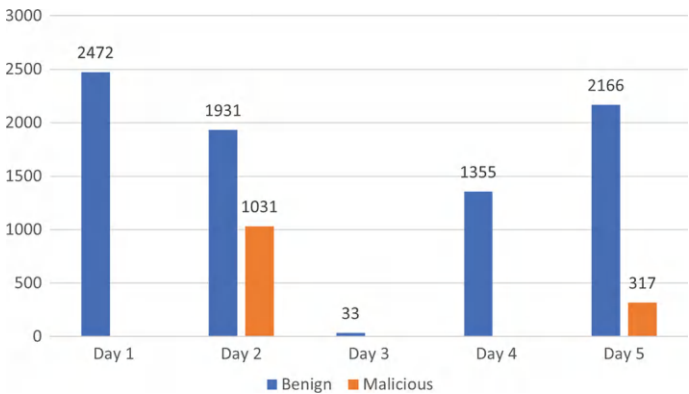


Fig. 2 Log files collected from day1 to day5

4.1 *Distribution of Log Quantities*

Figure 2 presents the system logs collected over a period of five days, with attacks conducted by the attack server against the victim host on the second and fifth days. On the second day, the main objective is to perform the “smash-and-grab” operation of APT29, which includes rapid collection and exfiltration data while attempting to remain covert. Subsequently, after deploying tools and erasing footprints, the attackers will remain dormant until the fifth day, at which point they will resume data collection.

On the third day, the victim host was idle, with all hosts maintaining network connectivity but only running the operating system and some necessary processes. In real-world scenarios, employees of organizations such as enterprises and governments do not work every day. Therefore, this work attempted to collect the activities that the host may generate during idle periods, such as scheduled tasks.

4.2 *Distribution of TTPs*

According to the content presented in Table 1, the simulated APT29 dataset created in this work includes 29 labeled techniques. These techniques are distributed across 10 tactics in the MITRE ATT&CK framework, excluding the initial steps related to reconnaissance and initial compromise, as well as the impact tactic. The exclusion of the impact tactic is because the primary objective of this simulated attack is not to cause service disruption or system damage.

It can be observed from the table that there are fewer techniques related to network operations. It is because host logs usually only record process creation during network operations, making it difficult to identify the specific purpose of such operations. Additionally, some discovery techniques are challenging to identify, resulting in a small number of occurrences. It is because there are many activities in the discovery phase that resemble normal system operations or administrative activities. Setting the identification criteria of Sigma rules too leniently could lead to a significant number of false positives in subsequent model training.

The aforementioned 29 techniques are classified into different tactics within the MITRE ATT&CK framework, as shown in Table 2. Some techniques in the table exhibit multiple characteristics and are simultaneously associated with different tactics. The distribution of techniques among various tactics is illustrated in Fig. 3.

Multiple techniques are used in the Discovery and Defense Evasion tactics. It is because these two tactics have finer-grained categorizations. For example, in the case of Discovery, the simulated APT29 dataset created in this work includes multiple techniques for probing host software, processes, network configurations, and more. In

Table 1 Technique label distribution in dataset

No.	Technique	IDs	Count	Percentage (%)
1	OS credential dumping	T1003	65	4.8
2	Query Registry	T1012	8	0.6
3	System network configuration discovery	T1016	21	1.6
4	Remote system discovery	T1018	59	4.4
5	Remote services	T1021	26	1.9
6	System owner/user discovery	T1033	41	3.0
7	Service execution	T1035	12	0.9
8	Masquerading: match legitimate name or location	T1036.005	8	0.6
9	Exfiltration over C2 channel	T1041	21	1.6
10	Process Injection	T1055	255	18.9
11	Input capture: keylogging	T1056.001	15	1.1
12	Process discovery	T1057	98	7.3
13	Command and scripting interpreter: PowerShell	T1059.001	207	15.4
14	Permission groups discovery	T1069	78	5.8
15	Indicator removal: file deletion	T1070.004	47	3.5
16	Indicator removal: timestomp	T1070.006	13	1.0
17	System information discovery	T1082	102	7.6
18	Ingress tool transfer	T1105	8	0.6
19	Modify registry	T1112	18	1.3
20	Screen capture	T1113	32	2.4
21	Email collection: local email collection	T1114.001	14	1.0
22	Clipboard Data	T1115	9	0.7
23	Automated collection	T1119	26	1.9
24	Access token manipulation: create process with token	T1134.002	16	1.2
25	System binary proxy execution: rundll32	T1218.011	11	0.8
26	Software discovery	T1518	18	1.3
27	Software discovery: security software discovery	T1518.001	13	1.0
28	Event triggered execution: windows management instrumentation event subscription	T1546.003	17	1.3
29	Unsecured credentials: private keys	T1552.004	10	0.7
	Total		1348	

Table 2 Mapping of tactics to techniques

No.	Tactic	Technique IDs
1	Execution	T1035, T1059.001
2	Persistence	T1546.003
3	Privilege escalation	T1055, T1134.002, T1546.003
4	Defense evasion	T1036.005, T1055, T1070.004, T1070.006, T1112, T1134.002, T1218.011
5	Credential access	T1003, T1056.001, T1552.004
6	Discovery	T1012, T1016, T1018, T1033, T1057, T1069, T1082, T1518, T1518.001
7	Lateral movement	T1021
8	Collection	T1056.001, T1113, T1114.001, T1115, T1119
9	Command and control	T1105
10	Exfiltration	T1041

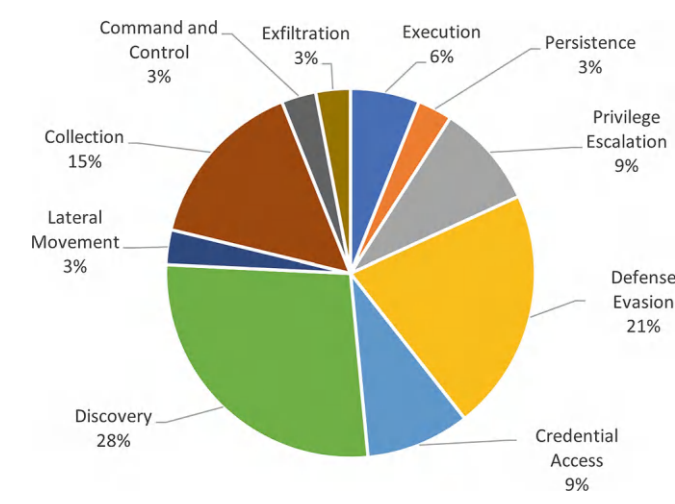


Fig. 3 Distribution of techniques across tactics

contrast, only one technique is classified under Lateral Movement. The main reason for this result is that lateral movement often requires the use of multiple tactics to achieve its objectives. In this study, for example, achieving lateral movement involves first conducting domain reconnaissance using T1018 (classified under Discovery). Once the target host is identified, T1021 (classified under Lateral Movement) is used to establish a remote PowerShell session on that host, and then T1105 (classified under Command and Control) is executed through that session to transfer the attack tools to the compromised host.

Table 3 Phase of APT life cycle covered in published datasets

Dataset	Life cycle phase	Establish foothold	Escalate privileges	Internal reconnaissance	Move laterally	Maintain presence	Complete mission
Friedberg et al. [10]		✗	✗	✗	✓	✗	✓
Bohara et al. [11]		✓	–	✓	✗	✗	✗
Giura and Wang [12]		✓	✓	✗	✗	✗	✗
Zhang et al. [13]		✓	✗	✗	✗	✗	✗
Yuan [14]		✓	✗	✗	✗	–	✗
Razaq et al. [15]		✓	✗	✗	✗	✗	✗
DAPT2020 [16]		✓	✓	✓	✓	✗	✓
The proposed scheme		✓	✓	✓	✓	✓	✓

4.3 Comparison and Discussion

The generation of datasets is often closely related to research on intrusion detection and intrusion prevention. The datasets mentioned in Table 3 are mostly generated to validate their anomaly detection mechanisms. The collection time for these datasets is often relatively short. For example, the dataset established by Zhang et al. took two weeks to collect, while the others were completed within a few hours. This makes it difficult for the datasets to truly reflect the characteristics of APT attacks. Additionally, the short collection time also results in a small amount of data, which is disadvantageous for training detection models. Therefore, all datasets employ synthetic data to increase their data volume.

Referring to Table 3, it becomes evident that the majority of datasets have a narrow scope, and it primarily targets specific stages of the attack lifecycle. Although lateral movement within the compromised organization is an important means for APT attacks to gain access to sensitive information, these datasets rarely reconstruct this step. The APT29 dataset created in this work provides a more complete reconstruction of the APT life cycle. The DAPT2020 dataset also reconstructs most stages, and a more detailed comparison is shown in Table 4.

Despite the fact that DAPT2020 dataset has reconstructed the majority of APT life cycle stages, it still has some limitations. In terms of reproducibility, due to the dataset not being open source and the attack simulations being conducted by

Table 4 Comparison between the proposed dataset and DAPT2020

	Reproducibility	Label	AD	Hosts	Duration	Stages of life cycle
The proposed dataset	High	✓	✓	5	5 d	6
DAPT2020	Low	✗	✗	2	5 d	5

external red team personnel, it becomes difficult for other researchers to reproduce the same conditions. Regarding simulating the target organization, the dataset lacks the construction of an AD, and the simulated victim hosts are limited to only two, which reduces the realism of the simulation. Furthermore, this dataset does not provide labels for the attack data, which imposes limitations on the utilization of the dataset in subsequent research.

5 Conclusions

The advanced persistent threat attacks often lurk in victim systems for extended periods, attempting to disrupt or steal sensitive information. During this process, attackers frequently utilize legitimate commands or advanced attack techniques to achieve their goals. MITRE has documented and developed a set of general procedures for defenders to reference in response to these actions. Machine learning has emerged as an effective means of identifying and defending against advanced persistent threat attacks. This has led to an increasing number of related research studies focusing on anomaly detection. However, model training relies heavily on a complete dataset.

Modern operating systems such as Windows 10 generate and manage logs, which allows for the recording of tools or legitimate commands executed by attackers, enabling the discovery of their intrusion and presence. APT attacks often target high-value confidential information, making it difficult to publicly disclose logs from real victim hosts and limiting related research. Many detection models are trained using incomplete or limited datasets that only cover a small number of attack techniques, resulting in limited research outcomes.

This work proposes a viable framework for generating datasets and labeling data. Within this framework, it utilizes the concatenation of attack scripts to simulate APT attacks and configure Sysmon on each simulated victim host to detect system activities and generate logs. In terms of log collection, it uses Winlogbeat to forward log files from multiple hosts to a storage host and deploy ELK on the storage host for log management and analysis. To label the dataset, Sigma rules are customized and authored to enable the detection of log files. When the content of a log file matches the detection rules in Sigma, the corresponding label field is added to the log file.

In this work, a simulated enterprise environment was deployed as the victim, and APT29 simulated adversary was simulated by orchestrating scripts. The logs from the compromised environment were subsequently collected and labeled to validate the feasibility of the framework.

For training APT attack detection models, a dataset should not only contain labeled attack data but also include benign data for the model to distinguish between the two. The APT29 dataset created through the framework in this work adopts a manual operation approach to simulate normal user behavior. While this approach reduces the reproducibility of the experiments, it enhances the diversity of the dataset and brings it closer to real-world environments. Similarly, due to the limited time available for manually generating benign data, the proportion of malicious operation logs is higher compared to the real environment.

In summary, this thesis has constructed a framework for generating labeled datasets and validated its feasibility. The simulated adversary log dataset generated through this framework includes APT29 attack techniques investigated by MITRE, and it covers six stages of the APT life cycle. Furthermore, the framework employs rule matching to identify 29 techniques within the simulated attacks and achieves approximately 94% labeling coverage of the attack data. These techniques span across 10 tactics, encompassing most of the post-compromise steps defined by MITRE for APT attacks. In comparison to datasets constructed in other detection models, this dataset retains more characteristics of APT attacks and is more comprehensive, providing attack data labels that enhance the effectiveness of machine learning training.

Acknowledgements This work was partially supported by the National Science and Technology Council (NSTC) of Taiwan under grant 112-2634-F-110-001-MBK. It also was partially supported by the Information Security Research Center at National Sun Yat-sen University in Taiwan and National Center for High-performance Computing in Taiwan.

References

1. Shiravi A, Shiravi H, Tavallaee M, Ghorbani AA (2012) Toward developing a systematic approach to generate benchmark datasets for intrusion detection. *Comput Secur* 31(3):357–374
2. Moustafa N, Slay J (2015) Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set). In: 2015 military communications and information systems conference (MilCIS). IEEE, pp 1–6
3. Stiawan D, Idris MY, Bamhdi AM, Budiarto R et al (2020) Cicans-2017datasetfeature analysis with information gain for anomaly detection. *IEEE Access* 8:132911–132921
4. Myneni S, Chowdhary A, Sabur A, Sengupta S, Agrawal G, Huang D, Kang M (2020) Dapt 2020-constructing a benchmark dataset for advanced persistent threats. In: Deployable machine learning for security defense: first international workshop, MLHat 2020, San Diego, CA, USA, August 24, 2020, Proceedings, vol 1. Springer, pp 138–163
5. Griffith J, Kong D, Caro A, Benyo B, Khoury J, Upthegrove T, Christovich T, Ponomorov S, Sydney A, Saini A et al (2020) Scalable transparency architecture for research collaboration (starc)-darpa transparent computing (tc) program, technical report. Raytheon BBN Technologies Corp, Cambridge United States

6. Zipperle M et al (2022) Provenance-based intrusion detection systems: a survey. *ACM Comput Surv* 55(7):1–36
7. CALDERA, <https://caldera.mitre.org/>
8. Winlogbeat, <https://www.elastic.co/downloads/beats/winlogbeat>
9. Kafka, <https://kafka.apache.org/>
10. Friedberg I et al (2015) Combating advanced persistent threats: from network event correlation to incident detection. *Comput Secur* 48:35–57
11. Bohara A, Thakore U, Sanders WH (2016) Intrusion detection in enterprise systems by combining and clustering diverse monitor data. In: *Proceedings of the symposium and bootcamp on the science of security*
12. Giura P, Wang W (2012) A context-based detection framework for advanced persistent threats. In: *2012 international conference on cyber security*. IEEE
13. Zhang T, Liao Q, Shi L (2014) Bridging the gap of network management and anomaly detection through interactive visualization. In: *2014 IEEE pacific visualization symposium*. IEEE
14. Yuan X (2017) Phd forum: Deep learning-based real-time malware detection with multi-stage analysis. In: *2017 IEEE international conference on smart computing (SMARTCOMP)*. IEEE
15. Razaq A, Tianfield H, Barrie P (2016) A big data analytics based approach to anomaly detection. *Proceedings of the 3rd IEEE/ACM international conference on big data computing, applications and technologies*
16. Myneni S et al (2020) DAPT 2020-constructing a benchmark dataset for advanced persistent threats. In: *Deployable machine learning for security defense: first international workshop, MLHat 2020, San Diego, CA, USA, August 24, 2020, Proceedings 1*. Springer International Publishing

Using Open-Source Intelligence to Archive Criminal Organizations



Peter Heusch and Patrick S. Chen

Abstract Gangs are closely knit groups that pose a significant threat to law and social security. They are able to do this because of the strong bonds that exist between their members, as well as the internal and external factors that constrain them. In the past, most academic research on gangs has relied on sociological methods. However, this paper proposes a new approach that uses mathematical models and information technology (IT). This approach is based on the following four core functions of an archival system: 1. Data acquisition, 2. Archive construction, 3. Data maintenance, 4. Application. The proposed approach has several advantages over traditional sociological methods. First, it is more dynamic and can be used to track the changing structure and activities of gangs. Second, it is more comprehensive and can be used to collect data from a wider range of sources. Third, it is more accurate and can be used to identify patterns and trends in gang activity. The paper uses a case study of a criminal gang in Taiwan to illustrate the application of the proposed approach. The case study shows how the gang's organizational structure, leadership, and activities have changed over time. It also shows how the gang has been able to expand its geographical reach and influence. The proposed approach is a new and innovative way to study gangs. It has the potential to provide law enforcement agencies with a more comprehensive and accurate understanding of gang activity. This information can then be used to develop more effective strategies for combating gangs and protecting society.

Keywords Archive · OSINT · Criminal organization · Social network analysis

P. Heusch (✉)
HFT Stuttgart, Schellingstraße 24, 70174 Stuttgart, Germany
e-mail: peter.heusch@hft-stuttgart.de

P. S. Chen
Tatung University, Zhongshan N Road, Taipei City 10491, Taiwan

1 Introduction

Archives are essential in a variety of settings, including government, military, society, economy, and business. After the 9/11 attacks, terrorism and organized crime have become major threats to public security. To monitor and control the activities of criminal groups, it is effective to establish archives for them. This allows us to better understand their organizations. Traditionally, archival records preserve data that is not intended to be altered. However, in the digital age, all data are subject to frequent creation, updating, modification, or deletion. This is also true of digital archival systems. To address this challenge, open-source intelligence (OSINT) provides a new perspective for digital archival systems.

Open-source intelligence (OSINT) is a methodology of search, collection, analysis, and use of data from publicly available sources, as well as its related techniques and software tools. It emerged from military applications and found useful applications in national security and law enforcement, even in business. To completely record the organizations and activities, OSINT provides the timely information for the archival system.

The core functions of an archival system are data acquisition, archival construction, archival application, and archival maintenance. In this study, we use social network analysis (SNA) as the theoretical framework to implement the system. SNA is a method of exploring social organizations, relationships, and interactions among their members, which can help us understand the structures and activities of criminal gangs.

For data acquisition, open-source intelligence (OSINT) can be used to provide a large quantity of materials as evidence to the official documents. For archival construction, various digital repository techniques can be applied. In this study, we use graph theory as the theoretical framework to organize the data.

For archival maintenance, operations of update, insertion, and deletion based on data structure are performed. For archival application, not only data search techniques but also artificial intelligence (AI) techniques can be used. For example, Bright et al. [1] showed a crime analysis based on big data.

The development of criminal gangs has always been a threat to social security. Their existence is a serious hidden danger to society. The formation, existence, and development of criminal gangs are often closely related to their leaders. This can be explained by the concept of centrality in social network analysis (SNA) [8, Chaps. 7–8].

Therefore, this study intends to use SNA to deeply understand the organization, operation mechanism, and development strategy of criminal gangs. This will help us to propose suppression measures to eradicate hidden social dangers, prevent the formation of new criminal gangs, and maintain national social security.

As a case study, a notorious criminal gang based on Taiwan has attracted worldwide attention due to its expansion and activities. After six or seven decades of development, with the advancement of globalization and economic development, the

gang's activities have extended to many countries and regions, becoming a criminal group that worries the world.

Since organized criminal groups engage in various illegal activities, such as drug trafficking, gambling, extortion, and corruption, which pose a serious threat to social order and public safety, the study of the development of organized criminal groups is an important topic. The proposal of this methodology is a new attempt to improve research methodology.

The main purpose of this research is to use information technology, especially archival science and social network analysis, to study organized crime groups. The advantages of this research methodology are mainly manifested in the following:

- Complete and effective records of the members, locations, organizational structures, and activities of criminal gangs, with timely updates of information.
- Flexible and accurate definition of individual members, groups, and subgroups, as well as their relationships, including peripheral personnel and organizations.
- Identification of core members and organizations (centrality) to provide references for government security decision-making or criminal investigation.
- Dynamically organizable events, including time, place, participants, as well as previous events and subsequent developments.
- Support for big data analysis to sort out the overall picture of the organization or to sort out the context or symptoms of illegal activities, especially to support criminal investigations.
- Visualization of the results to present them briefly to support security decisions.

2 Research Model and Design

There are many research methods for social network analysis, including the graphic-theoretic approach, the sociometric approach, and the algebraic approach. This study will discuss the graphic-theoretic approach.

2.1 Graphic Models

A graph consists of nodes of a collection (Node, N) and links to a collection (Link, L):

$$G = (N, L)$$

where $N = \{n_1, n_2, \dots, n_s\}$ is a set containing s nodes;

$L = \{l_1, l_2, \dots, l_t\}$ is a set containing t links.

In constructing a model, G can represent a criminal organization; N is a set of nodes, which may be a branch or a sub-organization; n_j is a gang member; L is the

collection of links, whose element lj can be affiliation, support, proximity, and so on. In this way, the organizational structure of a gang can be found by analyzing the network of links between members of a gang to understand its hierarchical structure, power core, and information flow patterns. The structure can support data analysis and help to gain a deeper understanding of the operating mechanisms and tactics of a gang and its negative impact on society.

2.2 Organizational Structure Center

In a graph, there are two types of links between nodes: bidirectional links (such as proximity) and unidirectional links (such as affiliation). The direction of the arrow represents the direction of the link.

The **in-degree** of a node is the number of links pointing to that node. For example, in Fig. 1, the nodes A, B, C, and D all have an in-degree greater than 1; node A has an in-degree of 8.

A **path** is a sequence of nodes such that there is a link between each consecutive pair of nodes. For example, K-D-C is a path in Fig. 1.

We say that node n_i is **reachable** from node n_j if there is a path from n_i to n_j .

In the context of gang organizations, the nodes with a high in-degree are likely to be branch leaders. In Fig. 1, all nodes have a path to node C, which suggests that node C is the center of the organization.

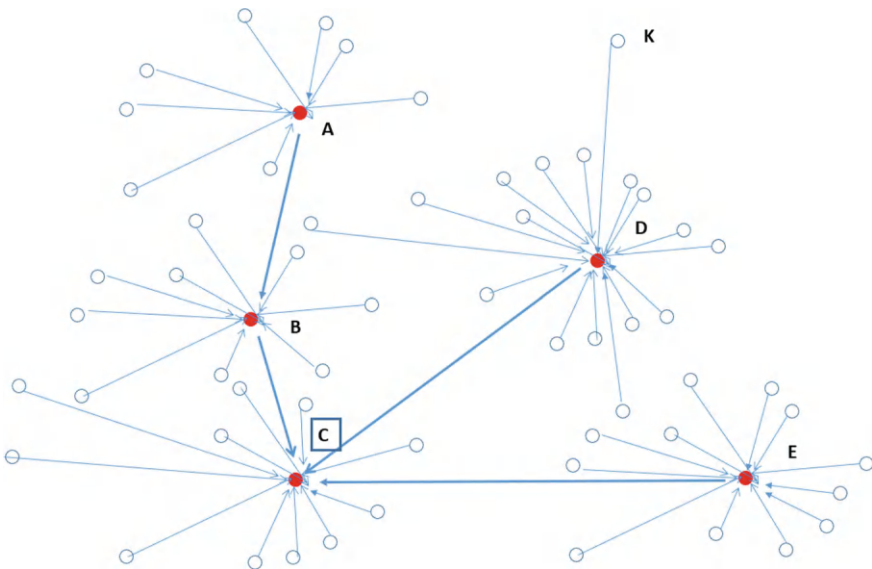


Fig. 1 Graphical representation of the organization center [7]

2.3 *Activities of Criminal Organization*

To study the change of the organizational center of a criminal gang, we will be carried out according to the following steps:

1. Organize information from existing files

Normally, the law enforcement has accumulated a lot of files about criminal organizations. They are data collected in the daily operations. These files should be reorganized based on certain mathematical foundation, for example, graph theory in this study. A theoretically well-founded structure in archival system or any digital repository will better support for future applications.

2. Collect data from publicly available sources

After having laid the basis of an archive, continuous data collection should start soon. An effective methodology is to adopt Open-Source Intelligence as the following step. Publicly sourced intelligence is used by law enforcement agencies as a data collection channel because it presents a certain degree of truth [2]. Through OSINT, researchers can obtain relevant information about a certain gang and fill the vacancy left by existing files. Specifically, OSINT's value to police agencies lies in observing continuous development of the gang, understanding astute intelligence data, supporting decision-making in dealing with organized crime, and establishing a complete dossier of organizations or incidents. The steps for OSINT collection are as follows [2]:

- I. Identify research objectives: here organized criminal groups;
- II. Collect from various OSINT sources;
- III. Trace leads;
- IV. Collate information;
- V. Certify the alertness of the information;
- VI. Evaluate data and follow-up.

3. Sort out major events of a gang

These major events and activities of a gang may include murder, drug trafficking, smuggling, robbery, and more, and by collecting and collating information about these events, researchers can gain insight into the gang's patterns, strategies, and influence. Collating major events can also help researchers identify the scope and possible areas of action of the gang.

4. Observe the shift of the gang center

Center shifting refers to the change of key person or main activities in different times or locations. By observing the activities and influence of gang over different time periods, the researchers can observe the patterns and causes of the gang center shifts. This comparative analysis can reveal the evolution of the gang and its possible tactical adjustments.

While studying centrality, we may need a variety of scales, commonly used:

- In-degree: calculation of the total number of links pointing to a node.
- Closedness: the measure of distance or intimacy between two nodes.
- Betweenness: The intermediary is often the one who has more resources and may become the center of substance.
- Concentration: People who grasp information and make good use of it often become the center;
- Rank: In the class structure, those who occupy the upper position will inevitably become the center, and the distribution of finances, power, etc. will become the center more naturally.

The above criteria are measures for the center of an organizational structure that can be derived from observation or data analysis techniques. When combined with artificial intelligence (AI) technologies, for example, K-Nearest Neighbors (KNN), it can help confirm outstanding properties of a network because it takes into account various factors to obtain the maximum cohesion within the group, but the minimum coupling among different groups. The detailed algorithm of KNN and its application is not the focus of this article, please refer to Shi et al. [5].

5. Analyze the relationship between events and center shifts

By deeply analyzing the timeline, location, and other relevant factors of major events and center shifts, researchers can assess the extent to which these factors affect bad gang organizations. This analysis can reveal the power structure within bad gang organizations, the strategic adjustments of leadership, and the impact of changes in the external environment on the shift of centers. Researchers can explore the role of different events in triggering center shifts, such as an event that leads to a decrease in the activity of gangs in a specific area, which in turn affects the central location of the organization. In addition, analyzing the relationship between incidents and center shifts can also help researchers understand the operation mode and strategy changes of gang organizations to propose corresponding preventive measures and countermeasures.

Center shift is essentially a function of time. In this study, we will observe the leader, location, and main activity on a timeline of a criminal organization. The changing function is formally described as follows, where x_i , y_j , z_i , are parameters such as economic conditions, legal constraints, social development, international events, etc.

$$X = f(\text{time, leader, } x_1, x_2, x_3, \dots)$$

$$Y = f(\text{time, location, } y_1, y_2, y_3, \dots)$$

$$Z = f(\text{time, activity, } z_1, z_2, z_3, \dots)$$

There are many factors impacting the development of criminal gangs; most of them were stopped by the law enforcement in their infancy. However, when it grows and thrives, it becomes a tight symbiotic organization. The organization is usually hierarchical and tightly interconnected. Those who are at the top have absolute control over their subordinates. Those who betrayed the organization will be brutally lynched

so that the will of the leader could be carried out. The hierarchical structure is a specific kind of graph, generally expressed as a tree. The tree structure has a root that can represent the gang leader. The subordinate nodes are the branch master, and the bottom leaves are the members (Fig. 2). Comparing with Fig. 1, it is not difficult to find that the gang leader is C while B, D, and E are the branch masters. A is the head of a division, and K is a member.

The data storage in the tree structure can be done by the following program code (Fig. 3). The root node is number 0, denoting the leader. The previous node of all branch masters is the leader; “data” is instance of ordinary “Person” (gang member). With the program code, we may record all gang members and their relevant data.

In our case study, the gang has taken advantage of globalization to seek new development opportunities in foreign countries and to evade the crackdown of Taiwan’s law enforcement agencies. For example, in recent years their subordinate branches can be found in Hong Kong, Macao, Southeast Asian countries, and China.

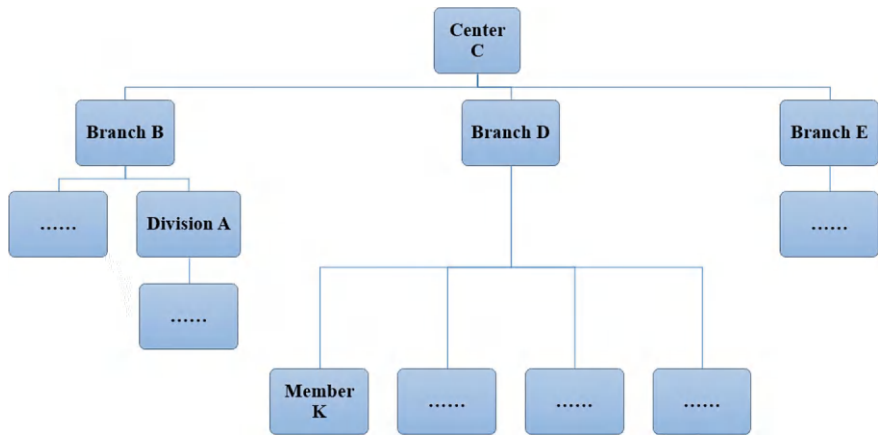


Fig. 2 Hierarchical organizational structure of gangs (refer to Wang et al. [7])

Fig. 3 Tree structure (reference to Goodrich [3])

```
/* Defining a tree node */
# define MAX_TREE_SIZE 2000
typedef struct {
    Person data;
    int parent; /* Parent node */
} PNode;

typedef struct {
    PNode nodes[MAX_TREE_SIZE];
    int n; /* Number of the node */
} PTree;
```

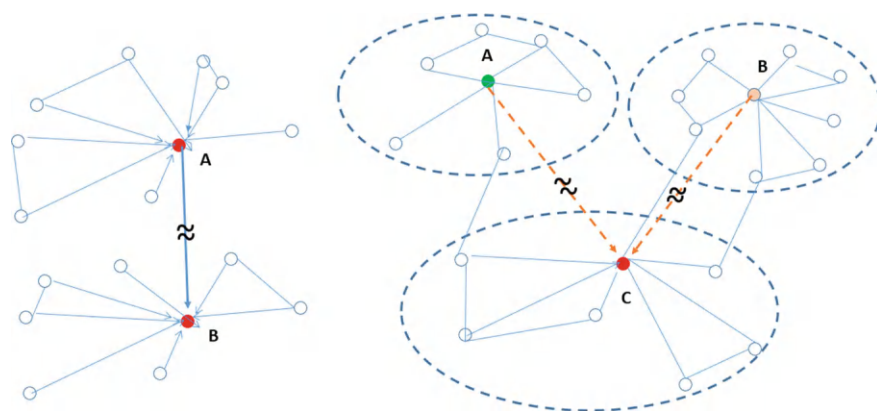


Fig. 4 Bridge versus hypergraph (refer to Wang et al. [7])

Traditionally, the activities of the gang include drug trafficking, gambling, extortion, embezzlement, etc. In order to evade government pursuit, the gang turned to operate various special businesses to disguise illegality, causing the change of their activities. When businesses are successful, the gang becomes legitimate in the society. In particular, they intervene in elections to support specific candidates. Once elected, the gang gets a shelter. It is more difficult for the law enforcement to control and monitor their activities.

In graph theory, the phenomenon of “dragons without head” in a gang can be described by the phenomenon of partition or centrifuge of graphs.

Figure 4 depicts the phenomenon of segmentation and centrifugation. In graph theory, a bridge is a connection between two sets of secondary structures. If a bridge is broken, the two secondary structures become independent. This phenomenon can be used to describe the separation of gang organizations.

Another concept is the hypergraph, which is a system with several strong clusters. The operation of the system depends on the connections between the cluster centers. If these connections are weakened, centrifugation occurs. As a result, as shown in Fig. 4 (right), the original center C is no longer able to maintain its position as the center of the entire gang organization.

As mentioned previously, the leader plays a pivotal role in a gang organization. The factors that determine who can become the leader of a gang are varied. The founders are naturally good candidates. However, after the first founders, all successors typically engage in a fierce power struggle to become leaders. This painful process is known as the center shift.

In social behavior theory, symbionts lose their centrality and cause the breakup of the gang if they fail to cultivate strong personal relationships and get along well with others. This includes assisting in tasks to gain acceptance and integration into the group [6].

2.4 *Geographical Center Proliferation*

Crackdowns by law enforcement can force key members of a gang to flee overseas. The departure of the central leader can sometimes lead to the displacement of the gang's headquarters, similar to how a queen bee that swarms will cause her colony to relocate.

In addition, globalization and economic development in recent years have expanded the scope of activities of many gangs. They take advantage of the convenience of globalization and the opportunities for transnational economic exchange to seek new opportunities for growth while also evading the crackdown of Taiwanese law enforcement. Gangs often form alliances with local gangs to develop their operations abroad, as it is difficult to establish illegal organizations in foreign countries. This poses a serious threat to social security in other countries.

Geographic Information Systems (GIS) can be used to understand the geographical distribution and activity characteristics of gangs. This can help to identify activity hotspots and understand the gang's spread.

A study used GIS to analyze the spatial distribution of organized crime in Taiwan. Their findings provided insights into the government's crime prevention and local policing strategies.

Hsu and Ko [4] also used GIS to study the cooperative relationship between criminal gangs and other international criminal groups. They found that these relationships further strengthen the position of gangs in international criminal networks.

It is worth noting that the gang in question was originally concentrated in northern Taiwan, with Taipei as its headquarters. However, the gang's leader is now not in Taiwan, and its operations have expanded to include the Chinese mainland, Hong Kong, Macau, Southeast Asia, and the United States. Despite this, the gang's command center is still in Taipei.

2.5 *Changes in the Focus of Activities*

The evolution of criminal organizations is a major concern for law enforcement. The following timeline shows the different focus of activities of the study case in several periods:

- Early days of development: The gang was founded by a group of middle school students who engaged in fighting, extortion, and other criminal activities.
- After several key members were arrested: The remaining members were released from prison with a grudge against society. They lacked professional skills, so they turned to operating illegal businesses, such as drug trafficking and smuggling.
- Rise of globalization: The gang took advantage of globalization to develop legitimate businesses that masked their illegal activities. They became more established in society and gained political influence.

- **Change in main activities:** The gang has stepped into politics, participating in local elections and gaining political status. They have also strengthened their self-protection by using religious activities and foreign political forces to drive public opinion in Taiwan. In some cases, they have even established political parties or groups to directly participate in politics. This has strengthened the gang's position in society and will have a major impact on the future.

Information technology can be used to monitor and predict the trend of gang activity. By collecting data and analyzing it using big data techniques, law enforcement can identify emerging trends and take steps to prevent them from becoming more serious.

3 Problem Discussion

As one of the most influential triad organizations in Taiwan, the distribution and activities of the gang have always been the focus of social and academic attention. By establishing an archive system for it, we can gain insight into the organizational development in Taiwan and around the world, and the evolution of the nature of their activities.

3.1 *Transfer of Gang Leadership Centers*

Tasselli et al. [6] identified the following reasons for organizational centralization transfer in their research:

- **The change of rank in the department:** This can lead to a change in the organizational center, as the new person in charge may have different priorities and goals.
- **The change of the leader and the branch master:** This can also lead to a change in the structural center, as the new leader may have a different vision for the organization.
- **Individuals with higher symbiotic motivation tend to associate with “people with high centrality scores.”** This means that people who are motivated to form close relationships with others are more likely to be connected to people who are central to the organization.

Therefore, gang members who identify with a “person with high centrality scores” may cause center shifts. In summary, symbiotic motivation in social connections can be used to predict network-centric movements of individuals over time.

3.2 Possibility for Suppression of Gangs in the Early Days of Their Birth

The case gang was originally a group of violent and aggressive middle school students. The leader of the gang was frequently involved in fights and trouble, and he was arrested and imprisoned several times. However, he eventually emerged from prison as the leader of the gang, and he continued to recruit more young people to join. Over time, the gang grew in size and power.

This case highlights the importance of early intervention in preventing the formation of gangs. The government should pay close attention to changes in the family, economic condition, and social environment of adolescents. By providing support and resources to these young people, the government can help to prevent them from joining gangs and becoming involved in crime.

3.3 Mastering the Communication Situation of Nodes to Monitor the Gang Structure and Center Transfer

Using big data analysis on the archival system of the gang, we can monitor their structural change, territory expansion, and activities to prevent problems before they occur. The characteristic of “illegality under legal cover” is a common practice of gangs, and running a business is an important source of their financial support.

3.4 Prevention of the Expansion of Bad Gangs

The case gang is now one of the largest and most powerful criminal organizations in Taiwan. With a large number of members and a wide range of influence, the gang has diversified its operations to include legitimate businesses, religious activities, and political interference. It has also established ties with foreign gangs and political forces. This has made the gang a global threat, and it is likely to continue to grow and pose a serious threat to national and international security.

4 Conclusion

This paper proposes a novel application of archival systems in public security. Conventional archival systems are designed for data that are not subject to change, but the modern digital environment requires systems that can be updated in real time. We propose to use open-source intelligence (OSINT) to construct archival systems that are capable of handling dynamic data.

A sound theoretical structure is essential for the success of archive construction. In this paper, we apply graph theory to provide a foundation for the system. We also discuss the data that should be included in the archive, how it should be structured, and how it can be used.

Social network analysis is a powerful tool that can be used to monitor criminal organizations. Law enforcement agencies can use this tool to track the movement of leaders, changes in activities, and expansion of territory. By understanding these trends, law enforcement can take steps to prevent criminal activity before it happens.

To combat organized crime, we recommend that law enforcement agencies establish an archival system for each gang. The system should be updated in real time, and big data analysis should be used to predict future activities. This will help law enforcement to prevent harm before it occurs, and to stop gangs from becoming involved in international terrorism.

The proposed system can also be used to make better decisions about security measures. By combining archival data with artificial intelligence, law enforcement can develop more effective strategies to maintain security and stability in society.

This study is a new attempt to research methodology. By using a formal model to understand criminal organizations, we hope to help law enforcement officers focus their crackdowns and develop more effective measures to maintain social order and public safety.

Acknowledgements The authors thank Zihong Wang, Kunhong Chen, Minrong Chiu-Huang, and Hongyuan Yen for their help in preparing the manuscript.

References

1. Bright D, Brewer R, Morselli C (2021) Using social network analysis to study crime: navigating the challenges of criminal justice records. *Soc Netw* 66(7):50–64
2. Evangelista JRG, Sassi RJ, Romero M, Napolitano D (2021) Systematic literature review to investigate the application of open source intelligence (OSINT) with artificial intelligence. *J Appl Secur Res* 16(3):345–369
3. Goodrich MT (2001) Data structures and algorithms. Wiley, Chap. 3
4. Hsu C, Ko S (2018) An analysis of triad activities in Taiwan using social network analysis. *J Criminol* 36(2):123–145
5. Shi B, Han L, Yan H (2018) Adaptive clustering algorithm based on KNN and density. *Pattern Recogn Lett* 104:37–44
6. Tasselli S, Neray B, Lomi A (2023) A network centrality bias: Central individuals in workplace networks have more supportive coworkers. *Soc Netw* 73:30–41
7. Wang CH, Chen KH, Chiu-Huang MR, Yen HY, Chen PS (2023) Study on center shift in a social network. To be published in HCLF Magazine (in Chinese). <https://www.hclf.org.tw/>
8. Wasserman S, Faust K (1994) Social network analysis: methods and applications (Structural analysis in the social sciences, series number 8), 1st ed. Cambridge University Press, New York

FIDO-Based Access Control Mechanism in Named Data Networking



Yeong-Sheng Chen and Wei-Cheng Chien

Abstract In contrast to the conventional IP-based architecture, Named Data Networking (NDN) provides a content-centric, secure, efficient, and scalable network approach, making it suitable for various applications such as content delivery and data dissemination. However, in NDN, the information carried in the Interest packets includes only essential data names, which enhances privacy and security but complicates the identification of the client requester. This study introduces an access control mechanism incorporating FIDO U2F devices for user authentication and replacing the original client names with IDs to enhance privacy protection. The proposed framework empowers the Access Controller to verify the source of Interest packets, thereby determining whether the client has the authorization to access the data. We compare and analyze the authentication mechanisms of the proposed architecture with other access control architectures. Our findings show that the proposed architecture has more flexible permission settings and can effectively protect clients' identity privacy.

Keywords Named data networking · Name-based access control · Fast Identity Online (FIDO)

1 Introduction

Named Data Networking (NDN) [1] is an emerging network architecture aimed at redesigning and improving the traditional Internet Protocol (IP)-based internet framework. NDN seeks to transition the architecture of the internet from location-based naming to content-based naming. The core concept of NDN is to be content-centric. When a user initiates a request on NDN, routers and other nodes in the network search for stored data based on the request's name. If the data exists in the

Y.-S. Chen (✉) · W.-C. Chien

Department of Computer Science, National Taipei University of Education, Taipei, Taiwan
e-mail: yschen@tea.ntue.edu.tw

W.-C. Chien

e-mail: welsongn00@gmail.com

network, it is returned to the requester without needing the actual location of the resource. This eliminates the host-to-host communication model commonly used in IP networks and offers a more efficient and flexible content distribution method.

To ensure resource security, privacy protection, and authorization management, many researchers have begun studying issues related to NDN verification and protection [2–9]. Among these, Name-Based Access Control (NAC) [10] is a name-based access control mechanism used to determine access rights to specific resources or services. Traditional access control methods often use identity-based authorization, where users are granted permissions based on their identity, such as usernames or roles. In contrast, NAC focuses on the name of the resource the user attempts to access. Compared to traditional access control, it aligns well with the “content-centric” concept of NDN.

As zero trust gains prominence, identity verification and access control become increasingly crucial. While NAC can protect and control data based on names by encrypting each data with a different Key-Encryption Key (KEK) for access control via Content Key (CK), it has some drawbacks. One is that its names are public, and understanding who requests a Key-Decryption Key (KDK) requires examining the Name on an Interest Packet. If someone else requests a similar packet, they can discern who is requesting what kind of KDK. Moreover, since Interest Packets cannot be verified, they are susceptible to forgery. Therefore, this study aims to combine NAC with the FIDO U2F protocol, using U2F Devices for identity authentication. By using IDs instead of the original Client Name, it aims to enhance client privacy protection.

The rest of this paper is organized as follows: Sect. 2 describes the background and related work. The proposed mechanism is presented in Sect. 3, and comparisons with related work are discussed in Sect. 4. Finally, Sect. 5 concludes the paper.

2 Background and Related Work

2.1 *Named Data Networking*

Named Data Networking (NDN) is a content-centric network architecture designed to address issues stemming from the current IP network framework. Unlike the IP network architecture, NDN shifts network communication from being source-centric to content-centric. This is achieved by naming each data packet rather than relying on endpoint addresses for data exchange. In essence, NDN identifies and retrieves data by incorporating the content’s name within each data packet instead of using IP addresses.

NDN aims for more efficient utilization of network resources, enhancing data exchange efficiency and security. By searching for data based on content names rather than addresses, it eliminates many routing and addressing issues inherent in IP network architectures. NDN also offers functionalities like point-to-point data

Fig. 1 Interest packets and data packets

Interest Packet	Data Packet
Name	Name
Selector	MetaInfo
Nonce	Content
Guiders	Signature

transmission, multicast transmission, and multi-node communication, which are challenging to achieve in IP network frameworks.

In NDN, communication primarily occurs through two different types of packets as depicted in Fig. 1: Interest Packets and Data Packets.

Interest Packets primarily consist of Name (the data name being sought), Selector (setting the data scope), Nonce (preventing loop formation from duplicate packets), and Guiders (defining characteristics). When a data consumer seeks data, they write the data’s name into the Interest Packet and transmit it into the NDN network. Routers, based on the Name on the packet, forward the Interest Packet to routers caching the data or to data producers.

Data Packets primarily consist of Name (data name), MetaInfo (time-to-live), Content (data content), and Signature (digital signature). When routers caching data or data producers receive an Interest Packet, they transmit the Data Packet back along the same path to the data consumer.

Apart from packets, the architecture of routers in NDN differs from conventional networks. To achieve Interest Packet and Data Packet forwarding functionalities, routers primarily consist of three distinct data structures: Content Store (CS), Pending Interest Table (PIT), and Forwarding Information Base (FIB).

CS is responsible for locally caching requested data for swift responses. When a router receives a data request, it first checks its local cache. If it finds the corresponding data, it can respond directly, reducing network latency.

PIT maintains a mapping table of Names to exit interfaces. When a data packet arrives at a router, it uses the forwarding table to find the correct exit interface.

FIB is responsible for forwarding data packets to the correct exit interface. It forwards packets based on information in the forwarding table.

Figure 2 depicts the scenario when an Interest Packet reaches a router. Upon arrival, the router first checks the CS for corresponding data. If data is found, it responds with the respective data from the CS and discards the Interest Packet. If not, it forwards the Interest Packet to the PIT to check for prior similar requests. Afterward, the Interest Packet is handled accordingly based on the PIT’s findings, either by forwarding or discarding the packet.

Figure 3 illustrates the scenario when a Data Packet reaches a router. Upon arrival, the router checks the PIT to determine if anyone requires the data. If no corresponding entry is found, the data is discarded; otherwise, it is transmitted based on the corresponding exit interface in the PIT and cached in the CS.

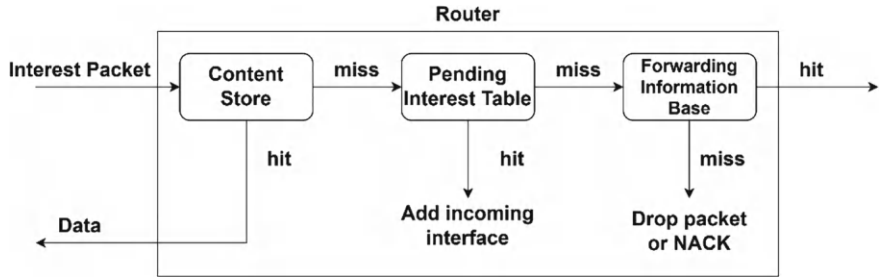


Fig. 2 Processing of interest packets

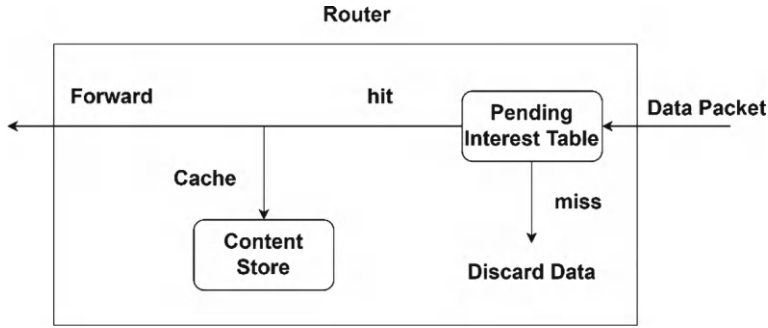


Fig. 3 Processing of data packets

2.2 Name-Based Access Control in NDN

Name-Based Access Control (NAC) [10] is a name-based access control method used to manage user access permissions to named data. Within the NDN architecture, NAC serves as a crucial security mechanism by applying security tags to names, thereby achieving data protection and control.

In NAC, an Access Controller generates a pair of keys known as KEK and KDK, where KEK and KDK represent the public and private keys, respectively. Data producers utilize a symmetric key generated by them to encrypt the content. As depicted in Fig. 4, data owners implement read access control by controlling the transmission of content keys. All authorized data consumers receive the private key, while data producers retrieve the public key and use it to encrypt CK. The data owner explicitly specifies the permissions for consumption certificates, indicating the data sets that consumers are authorized to read, allowing data producers to determine which content keys should be encrypted with specific KEK.

Figure 5 illustrates the naming scheme for encrypted keys in Interest Packets and Data Packets, enabling the retrieval of keys required to decrypt the data. Initially, the Consumer obtains the content they desire. However, the Consumer lacks the ability to decipher the content's CK. Thus, the Consumer uses the information regarding

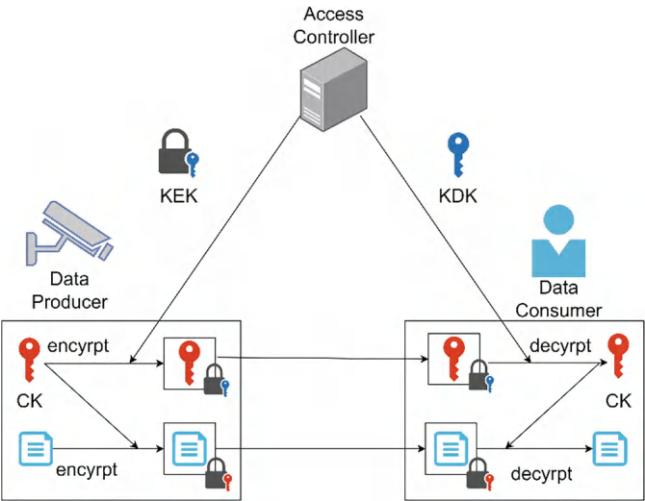


Fig. 4 Named-based access control

Content Data	<i>/<ContentName></i>
CK Interest	<i>/<ContentName>/CK/<CK-ID></i>
CK Data	<i>/<ContentName>/CK/<CK-ID>/ENCRYPTED-BY/<KEKName>/KEY/<Key-id></i>
KDK Interest	<i>/<KDKName>/KDK/<Key-id>/ENCRYPTED-BY/<ConsumerName>/KEY/<Key-id></i>
KDK Data	<i>/<KDKName>/KDK/<Key-id>/ENCRYPTED-BY/<ConsumerName>/KEY/<Key-id></i>

Fig. 5 Packet naming principles for NAC

the content’s Name and the encrypted keys in the data packet to request the CK. Upon acquiring the CK, if the Consumer has not requested the KDK, they use the information in the CK packet’s Name to request the KDK, specifying to the Access Controller whose key to use for encrypting and returning the KDK. Upon receiving the request, the Access Controller verifies the permissions associated with the Name. If permissions exist, the Access Controller responds to the Consumer’s request.

The advantage of NAC lies in its implementation of data protection and control based on names rather than user addresses or identity information. This approach facilitates more flexible, secure, and reliable data control, better coping with the continually evolving users and dynamic data within the network.

2.3 FIDO U2F

Fast Identity Online Universal Second Factor (FIDO U2F) [11] is an advanced identity authentication technology that has gained widespread application in numerous network applications and services [12, 13]. Widely regarded as a significant milestone in enhancing online account security, FIDO U2F addresses many weaknesses of traditional password authentication.

This technology operates based on the principle of public/private key encryption, achieving multi-factor authentication by providing physical security keys. Here are some key aspects of FIDO U2F:

- **Password Elimination:** FIDO U2F entirely removes the need for using passwords in the authentication process. This means that even the strongest passwords are invulnerable to theft or guessing because this technology does not rely on memorized information.
- **Physical Security Keys:** A prominent feature of FIDO U2F is the use of physical security keys that can only be utilized when in possession of the person physically holding them. This hardware-based security prevents numerous network attacks such as phishing and password cracking.

FIDO U2F represents a modern identity authentication solution with significant security and usability advantages. It eliminates weaknesses associated with passwords, providing higher security while simplifying user experience, thereby making online accounts more secure and manageable.

3 The Proposed Mechanism

3.1 System Structure

Figure 6 illustrates the overall architecture of the study, which is modified from the mechanism developed by Zhang et al. [10]. Our focus lies primarily on the relationship between the Client and the Access Controller, which contains two phases: the Registration Phase and the KDK Request Phase. In the Registration Phase, the client registers his/her FIDO U2F device to the access controller. In the KDK Request Phase, the client requests the KDK from the access controller, and the access controller grants the request only after it verifies the identity of the client with the corresponding FIDO U2F device.

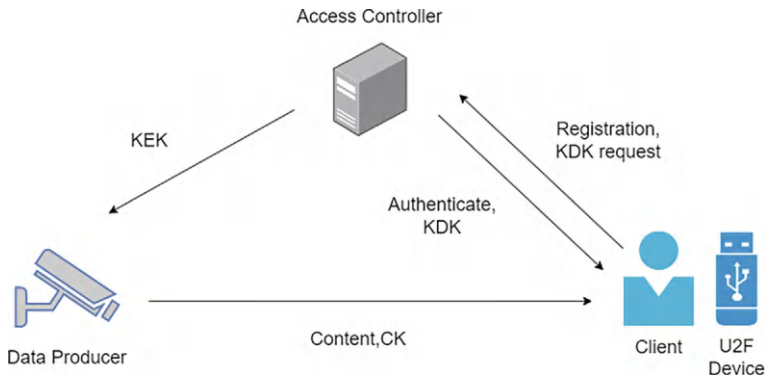


Fig. 6 System structure

Table 1 Example of access control list

U2F_ID	Name	Attributes	Public key	Registration	Expiration time
c4a6	Alice	Student, Woman	(Public Key A)	True	202312310000
1a2b	Bob	Student, Man	(Public Key B)	True	202312310000
9cda	Ken	Teacher, Man	(One Time Password)	False	202312310000

3.2 Access Control List

The Access Control List stores Client data requested by the Access Controller. Within this list, we store the U2F ID, Name, attributes, the public key used for U2F authentication, registration status, and the expiration time. Table 1 provides an example of the Access Control List. In this example, Alice and Bob are individuals who have registered U2F devices, while Ken has not registered. Therefore, in Ken’s Public Key field, instead of storing the public key used for U2F authentication, it contains a One-Time Password used for registration purposes.

3.3 Registration

Figure 7 depicts the process flow of the Registration Phase. Here, we assume that the client has already submitted their data to the Access Controller through a certain method. Hence, the Access Controller possesses some client-related data that can be used for one-time identity verification, such as a password. Initially, the client sends a registration request in the form of an interest packet to the Access Controller. Upon

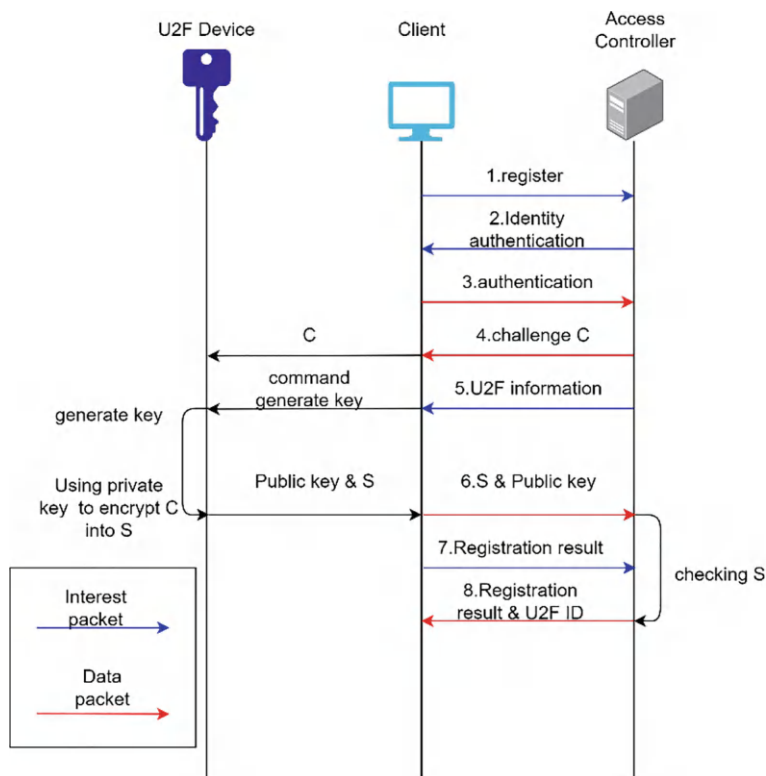


Fig. 7 Registration phase

receiving this, the Access Controller responds with an interest packet requesting identity authentication. Subsequently, the Client transmits prior data that can verify its identity back to the Access Controller. Upon receiving this data, the Access Controller generates a challenge and sends it back to the Client for U2F Device registration. Simultaneously, it sends an interest packet requesting the Client to return the challenge and the subsequent authentication public key. The Client, upon receiving the challenge, forwards it to the U2F Device, which generates a public-private key pair. The U2F Device then uses its private key to encrypt the challenge and returns the encrypted challenge and public key to the Client, with the private key securely stored in the U2F Device. The Client transmits the data received from the U2F Device back to the Access Controller. Upon receiving the encrypted challenge and public key, the Access Controller verifies their correctness. If the verification succeeds, the Access Controller sends a U2F ID back to the Client and stores the public key.

The packet naming convention and the Name used during the Registration Phase are described as follows. Initially, in the first step, the packet's Name is `"<AccessControllerName>/REG_BY/<ClientDeviceName>"`. Here, `"<AccessControllerName>"` indicates the packet's request directed towards

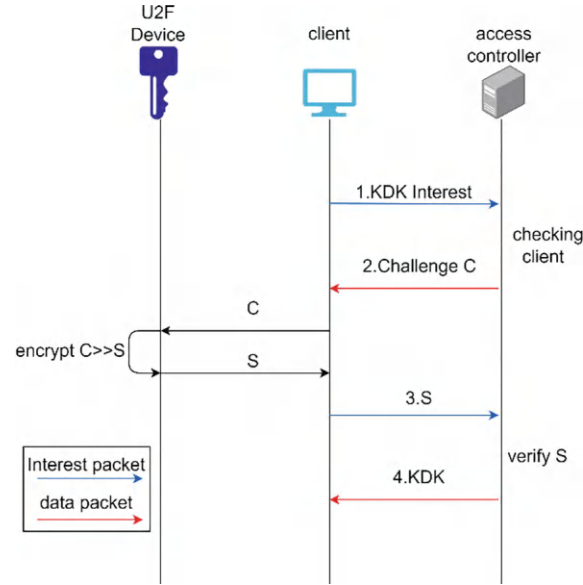
the Access Controller, while “<ClientDeviceName>” signifies the origin device. In the subsequent steps, “<ClientDeviceName>” is utilized for transmitting interest packets. In the second step, the packet’s Name is “<ClientDeviceName>/Ath”, where “<ClientDeviceName>” is extracted from the first packet. The third step’s packet, being the data packet corresponding to the second step, directly adopts the Name from the second step’s packet. Similarly, the fourth step’s packet also utilizes the Name from the first step’s packet and “Challenge” is appended to the packet’s Name. The fifth step’s packet describes that the Access Controller requests information from the Client’s U2F Device, and hence the Name is set as “/<ClientDeviceName>/FIDO/info”. The sixth step’s packet corresponds to the fifth step’s packet, and the same Name is used for the return. In the seventh step, the Client requesting results from the Access Controller. Hence, the Name uses “/<AccessControllerName>/RESULT” to denote this request, followed by “<ClientDeviceName>” for Access Controller confirmation. The eighth step’s packet corresponds to the seventh step’s packet, thus directly adopting the Name from the seventh step.

3.4 KDK Request

Figure 8 illustrates the process of a Client requesting KDK from the Access Controller. Initially, the Client sends an interest packet requesting the KDK to the Access Controller. Upon receiving this request, the Access Controller checks for the presence of the U2F ID in the interest packet. If it exists, the Access Controller generates a challenge string and sends it back to the Client in a data packet. Upon receiving the challenge, the Client encrypts it using its U2F Device. As the interest packet does not carry any data, the encrypted challenge is attached to the Name of the interest packet and returned to the Access Controller. Subsequently, the Access Controller verifies the encrypted challenge using its corresponding public key. If the verification is successful, the Access Controller provides the KDK required by the Client.

The packet naming convention during the KDK request phase is described as follows. In the initial step, the packet represents the Client’s request to the Access Controller for the KDK. The Name is structured as “/<AccessControllerName>/KDK/<Key-id>/REQUEST-BY/<U2F-ID>”. The segment “/<AccessControllerName>/KDK/<Key-id >” indicates the specific KDK requested by the Client. The trailing “<U2F-ID>” in the Name confirms the Client’s identity to the Access Controller. The second step packet is a response to the data packet in the first step, and it adopts the Name from the initial step’s packet. The third step packet contains the returned encrypted challenge from the second step. Thus, the Name is set as “/<AccessControllerName>/CHALLENGE-BY/<U2F-ID>/S”. Here, the “S” at the end signifies the challenge string encrypted by the U2F device. Since the interest packet doesn’t carry any data, it is attached to the Name and returned. The fourth step packet is a response to the data packet from the third step, and it follows the Name structure of the third step’s packet.

Fig. 8 KDK request phase



4 Comparisons with Related Work

We conducted a functional comparison between our access control design and three other related access control approaches: [10, 14, 15]. As illustrated in Table 2, the proposed framework can achieve identity confirmation, identity authentication, identity protection, and individual permission revocation.

Firstly, both our design and data-based access control perform identity confirmation, whereas NAC and data-based access control perform identity authentication. NAC-ABE does not engage in identity confirmation as it utilizes Attribute-Based Encryption (ABE).

Table 2 Comparisons with related work

Access control	Identity confirmation	Identity authentication	Identity protection	Individual permission revocation
The proposed mechanism	✓	✓	✓	✓
NAC [10]	✓	✗	✗	✓
NAC-ABE [14]	✗	✗	✓	✗
Data-based access control [15]	✓	✓	✗	✓

Regarding identity authentication, our design and data-based access control are involved. We utilize FIDO U2F for authentication, whereas data-based access control proposes password-based verification.

For identity protection, our design and NAC-ABE are applicable. Our method uses a predefined U2F ID between our U2F device and the Controller, while NAC-ABE does not need to disclose their identity.

Lastly, regarding individual revocation of permissions, our design and NAC, along with data-based access control, are capable. NAC-ABE uses Attribute-Based Encryption, necessitating the redistribution of attributes to all clients.

5 Conclusions

This study addresses an authentication challenge associated with access control in NDN, where verifying the source of interest packets is challenging. We propose a framework integrating FIDO U2F devices into the NDN architecture to allow the access controller to authenticate the client's identity while preserving the client's anonymity.

References

1. Zhang L et al (2014) Named data networking. *ACM SIGCOMM Comput Commun Rev* 44:66–73
2. Ullah SS, Hussain S, Gumaei A, AlSalman H (2021) A secure NDN framework for Internet of Things enabled healthcare. *Comput Mater Contin* 67(1):223–240
3. Meirovitch D, Zhang L (2021) NSC–named service calls, or a remote procedure call for NDN. Technical report NDN-0074, Revision 1, NDN
4. Ramani SK, Tourani R, Torres G, Misra S, Afanasyev A (2019) Ndn-abs: attribute-based signature scheme for named data networking. In: *Proceedings of the 6th ACM conference on information-centric networking*, pp 123–133
5. Shang W, Yu Y, Liang T, Zhang B, Zhang L (2015) NDN-ACE: access control for constrained environments over named data networking. NDN Project, Technical report NDN-0036, Revision 1
6. Zhang Z, Wong SY, Shi J, Pesavento D, Afanasyev A, Zhang L (2020) On certificate management in named data networking. [arXiv:2009.09339](https://arxiv.org/abs/2009.09339)
7. Pesavento D, Shi J, McKay K, Benmohamed L (2022) PION: password-based IoT on boarding over named data networking. In: *ICC 2022-IEEE international conference on communications*, pp 1070–1075
8. Ullah A, Hussain SS, Alroobaea R, Ali I (2021) Securing NDN-based internet of health things through cost-effective signcryption scheme. *Wirel Commun Mob Comput*, 1–13
9. Tehrani PF, Osterweil E, Schmidt TC, Wählisch M (2022) SoK: public key and namespace management in NDN. In: *Proceedings of the 9th ACM conference on information-centric networking*, pp 67–79
10. Zhang Z, Yu Y, Afanasyev A, Burke J, Zhang L (Sep 2017) NAC: name-based access control in named data networking. In: *Proceedings of the 4th ACM conference on information-centric networking*, pp 186–187

11. Srinivas S, Balfanz D, Tiffany E, Czeskis A, Alliance F (2015) Universal 2nd factor (U2F) overview. FIDO Alliance Proposed Standard
12. Chifor BC, Bica I, Patriciu VV, Pop F (2018) A security authorization scheme for smart home Internet of Things devices. *Futur Gener Comput Syst* 86:740–749
13. Togan M, Chifor BC, Florea I, Gugulea G (Jun 2017) A smart-phone based privacy-preserving security framework for IoT devices. In: 2017 9th international conference on electronics, computers and artificial intelligence (ECAI), pp 1–7
14. Zhang Z, Yu Y, Ramani SK, Afanasyev A, Zhang L (Oct 2018) NAC: automating access control via named data. In: MILCOM 2018–2018 IEEE military communications conference (MILCOM), pp 626–633
15. Hamdane B, Msahli M, Serhrouchni A, El Fatmi SG (Oct 2013) Data-based access control in named data networking. In: 9th IEEE international conference on collaborative computing: networking, applications and worksharing, pp 531–536

Artificial Intelligence (II)

Empirical Approach to a Fine-Tuning Using Forgetting in Large Language Models



Kaito Taguchi, Yujie Gu, and Kouichi Sakurai

Abstract Recent language models continue to improve their performance as the number of parameters is increased. On the other hand, it is known that large language models tend to store whole training samples. It is called memorization. If the training data contains private information, data privacy protection becomes an issue. Deep learning models are also prone to forgetting previously learned data, and large language models have a lower bound on the amount of forgetting. Currently, methods exist to mitigate memorization, such as de-duplication and the method using a learning algorithm that guarantees differential privacy. However, de-duplication does not completely prevent memorization. In this paper, we propose a method that splits a dataset and fine-tunes the pre-trained model in two steps. The method aims to reduce the memorization of the dataset used the first time by taking advantage of the nature of forgetting in large language models. Experimental evaluation results showed that fine-tuning, which updates all parameters, decreases the storage of the first data set by about 2%, although the perplexity decreases by about 0.1, and also decreases the overall storage by about 0.2%. The same trend was not observed for the Adapter with fine-tuning by adding new parameters, and additional verification is considered necessary.

Keywords Large language models · Fine-tuning · Memorization · Forgetting · Privacy

1 Introduction

In recent years, Large Language Models (LLMs) have seen an increase in the number of parameters as well as a significant improvement in performance [2]. On the other hand, LLMs are known to memorize sentences and words contained in the training

K. Taguchi (✉) · Y. Gu · K. Sakurai
Kyushu University, Fukuoka, Japan
e-mail: taguchi.kaito.834@s.kyushu-u.ac.jp

data, a property called memorization [4–10]. Training data may contain information about individuals, and protecting the privacy of this information is a challenge.

A concept that contrasts with memorization is forgetting. Learning by neural networks is known to have the property of catastrophic forgetting [22]. Catastrophic forgetting makes it difficult for a single model to solve tasks in multiple domains through continual learning. Even LLMs have been shown to have forgetting properties [6, 11]. However, a lower limit to forgetting, known as the forgetting baseline, has been empirically confirmed, and it can be assumed that some textual information remains [6].

Fine-tuning of LLMs is performed using supervised learning, and methods that update all parameters or only additional parameters are known. Mireshgallah et al. have shown that memorization exists for these fine-tuning methods as well [5]. Currently known methods for memorization reduction are de-duplication [19, 20] and differential privacy [15, 16].

De-duplication is the best approach currently known. Removing duplicates in the training data reduces perplexity and also reduces the amount of memorization. However, it is noted that it is not a method that can completely eliminate memorization [9]. Differential privacy is a concept that mathematically protects the privacy of data in the adjacency database. Differential privacy in deep learning is achieved by random sampling, adding Gaussian noise to the gradient, and gradient clipping when computing the gradient during training [16].

Although it is known to reduce memorization due to membership inference attacks, extraction attacks, etc., there is a tradeoff between privacy protection and model accuracy, and the performance of the model is degraded compared to that of normal training [17, 18]. If the text of the training data can be prevented from being stored in its entirety, the leakage of private data, even if it is included in the training data, can be controlled. On the other hand, currently, there is no known method to prevent memorization completely while maintaining accuracy.

In this study, we propose a fine-tuning method that takes advantage of the forgetting nature of LLMs and aims to reduce memorization related to training data without compromising accuracy.

2 Preliminary

2.1 Language Model

A model of the generation probability of words and sentences is called a language model. It can be formulated as follows. For a word sequence $x_1, x_2, \dots, x_n (x_i \in V)$, the following formula (1) is called a language model. V is the vocabulary.

$$p(x_1, x_2, \dots, x_n) = \prod_{t=1}^n p(x_t | x_1, \dots, x_{t-1}) \quad (1)$$

The language model in this study is a neural language model composed of a neural network. When a language model is used to generate words from a certain context, the likelihood of a word is used to determine the next word. This property can be used to calculate the likelihood of an entire sentence.

2.2 Perplexity

One of the evaluation metrics for language models is perplexity (PPL). Perplexity is defined by Eq. 2 and represents the difficulty with which a language model produces a given word sequence.

$$PPL = p(x_1, x_2, \dots, x_n)^{-1/n} \quad (2)$$

The lower the perplexity, the easier it is to generate a given word sequence.

2.3 Training Methods in LLMs

Recent large-scale language models have two learning phases. They are pre-training and fine-tuning.

In pre-training, self-supervised learning is performed using a huge amount of textual data. For example, in the GPT series [2], next word prediction, which predicts the next word from the context of k words, is pre-training. The goal of pre-learning is to acquire general natural language understanding. Fine-tuning, on the other hand, is learning to improve the accuracy of responses to the task to be solved using the language model. This study uses a supervised learning approach with additional training data.

3 Memorization and Forgetting in LLMs

3.1 Memorization in LLMs

Memorization in LLMs refers to the state in which the language model remembers the text contained in the training data as it is.

Currently known properties of memorization include the following.

1. Data with duplicates within training data are easily memorized [9, 17, 19, 20].
2. Models with a larger number of parameters have more to memorize and require fewer epochs to memorize [6, 9].

3. Memorization in fine-tuning has three phases [5].
4. Excluding vulnerable samples makes others vulnerable [10].

For 1 and 2, the meaning is as written. Regarding 3, the first phase is mainly a fitting and memorization phase in which the perplexity of the test data decreases; the second phase is mainly a memorization-only phase in which the amount of memorization increases; the final phase is an overlearning phase, in which the perplexity rises and the memorization approaches 1.

For 4, for a language model M trained on a certain dataset D , the memorization is measured by the membership inference attack described in the next chapter, and D' is created by removing samples that are determined to be memorized. Next, a new language model M' was trained using D' , and it was experimentally shown that the same amount of memorization was obtained. This phenomenon is described as another sample being more vulnerable to attack and is called the onion effect, analogous to the peeling of an onion.

3.2 Measuring Memorization

Memorization in LLMs is defined indirectly by attacking the target model and using the results. The following two main attack techniques are used. Membership inference attack [12] and extraction attack [8].

A membership inference attack is an attack that, given a sample x , guesses whether x is included in the training data of the target model M_θ . Extraction attack is a method to extract sentences included in the training data by repeatedly generating sentences with various prefixes given to the target model.

While the membership inference attack uses the statistical tendency to learn the training data as the memorization criterion, the extraction attack uses the possibility of generating the same text as the training data as the memorization criterion. Using the results of these attacks, memorization is defined as how many percent of the training data the target model remembers using the formula 3.

$$\text{Memorization} = \frac{1}{|D|} \sum_{i=1}^{|D|} \mathbf{1}_A(s_i \in D) \quad (3)$$

Here $\mathbf{1}_A(s)$ is a function that returns 1 if it remembers some sample s and 0 otherwise when using attack A . The computational cost of the extraction attack is higher than that of the membership inference attack because it involves language generation. Therefore, in this study, the membership inference attack is used to evaluate memorization.

3.3 Membership Inference Attack

For the attack, we use an approach based on the likelihood ratio test [4, 13, 14]. The assumption is that the attacker has a reference model M_{θ_R} trained on different training data than M_θ and can measure the likelihood of the sample using M_θ . The detail is as follows.

For training examples D sampled from a population p , we assume that a sample $x \in p$ is $x \notin D$, and we can show that it is not. Namely,

Null Hypothesis (H_{out}): $x \notin D$ and $x \in p$

Alternative Hypothesis (H_{in}): $x \in D$

In other words, it is equivalent to rejecting H_{out} when the above hypothesis is made. The attacker gives x for M_θ and M_{θ_R} and obtains the likelihood $p(x; \theta)$ and $p(x; \theta_R)$ for each model. Next, the following statistics are calculated using these likelihoods.

$$LR(x) = p(x; \theta_R) / p(x; \theta)$$

Compare $LR(x)$ with some threshold t and reject H_{out} if $LR(x) \leq t$. If it is smaller than the threshold, it means that $p(x; \theta_R)$ is larger relative to $p(x; \theta)$, so we can assume that only the target model has learned x .

The threshold value used during the test is calculated using a dataset different from the training dataset D . In this study, we follow related studies and compute it using the validation dataset D_v [4, 5].

Calculate $LR(y_i)$ for all data $y_i \in D_v$ in the validation dataset D_v and create a histogram. The threshold value is set to $LR(y_i)$ at the point where the cumulative frequency is 10% of the total. This means that a false positive rate of 10% is determined on the validation dataset, and since the training data is expected to have a smaller LR , a detection with a false positive rate of at least 10% is possible. Note that the value of 10% is the value empirically demonstrated in Miresghallah et al. [4].

3.4 Forgetting in LLMs

In neural networks, it is known as catastrophic forgetting in the context of continual learning of multiple domains [22]. In parallel with the research on memorization that has been described so far, research on forgetting is being conducted [6, 11].

Forgetting in LLMs has been verified by the following experiments. In pre-training with training data D , D' is used in addition to D only at a certain epoch t , and then only D is used again at the subsequent epochs to observe the forgetting of D' [6]. This experiment empirically showed that there is a lower limit, called the forgetting baseline, to the forgetting of stored text, and that not all samples are forgotten.

Based on these results, we hypothesized that the performance of the language model could be maintained while reducing the amount of memorization since some samples lost information due to forgetting but some samples remained.

4 Our Proposal

We propose a fine-tuning method that divides a dataset into two parts and fine-tunes a model in two steps to reduce the overall memorization by forgetting the memorization of the first dataset. We show the overview of our proposal in Fig. 1. First, a dataset D is divided into two parts, D_1 and D_2 . Next, we train the pre-trained model M_{PT} with D_1 , then we get the M_1 using criteria such as perplexity and amount of memorization. Further, we train M_1 with D_2 and get M_2 as well as M_1 .

5 Experiment

The purpose of this experiment is to experimentally demonstrate the impact of the proposed fine-tuning method on the overall memorization of the training data. GPT-2 small is used as the target model [2]. As a comparison, we use the model M_{12} trained by the conventional method without splitting D . And the GPU used in the experiments is RTX4090.

5.1 Dataset

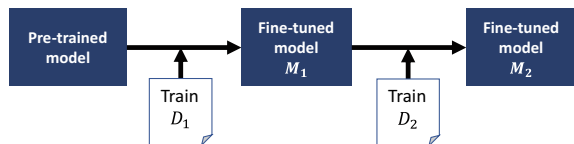
In this experiment, the dataset is a sampling of the Enron mail dataset [23]. The sampling method is based on Mireshghallash et al. When splitting D into D_1 and D_2 , we split them in half so that the number of letters in both halves is equal.

5.2 Fine-Tuning

Two fine-tuning methods are used: Full and Adapter [3, 21].

1. **Full:** A method to update all parameters.
2. **Adapter:** A method that adds an Adapter layer after the feedforward layer in a total of 12 decoder blocks and updates the Adapter layer and the last linear layer, the Head.

Fig. 1 The Overview of Our Proposal



The Adapter layer consists of a Down projection layer that compresses the number of parameters and an Up projection layer that restores the number of parameters to their original size. The number of parameters of the latent expression when compressing is determined by the expression (4).

$$reduction\_factor = d_{hidden}/d_{bottleneck} \quad (4)$$

The d_{hidden} refers to the number of output neurons in the feedforward layer and $d_{bottleneck}$ refers to the number of neurons when compressed.

In this experiment, $reduction\_factor = 16$ is used with reference to the experiment of Mireshghallah et al. [5].

5.3 Setting of Membership Inference Attacks

The reference model used in the membership inference attack is GPT-2, which is only pre-trained. In addition, the dataset used for threshold determination is the validation dataset used [5]. We use D_{1_v} to measure the memorization of D_1 and D_{2_v} to measure the memorization of D_2 for threshold determination. Both D_{1_v} and D_{2_v} are used to measure the combined D_1 and D_2 memorization.

5.4 Hyperparameters for M_1 and M_2

In this experiment, we trained with a batch size of 4, epoch 20, and a learning rate (LR) of $\{0.001, 0.0001, 2 * 10^{-5}\}$, recorded the memorization and perplexity at the end of each epoch and selected a model as shown in Table 1. The model was selected based on the point of best perplexity as shown in Table 1.

5.5 Result

First, for memorization, the measurement results for D_1 , D_2 and both are shown in Tables 2 and 3 is shown. The first column denotes the fine-tuning model used and the rest of the columns denote the percentage of memorization.

Compared to M_{12} , D_1 is remembered more, but D is remembered less by the proposed method. In the Adapter, we can see that memorization of D_2 decreases after the second fine-tuning.

Next, the perplexity is shown in Tables 4 and 5. The first column denotes the fine-tuning model used and the rest of the columns denote the perplexity.

Table 1 Hyperparameters used to train each model

Model	Epoch	LR
M_1 (Full)	5	0.0001
M_2 (Full)	5	$2 * 10^{-5}$
M_{12} (Full)	4	0.0001
M_1 (Adapter)	20	0.001
M_2 (Adapter)	20	0.001
M_{12} (Adapter)	20	0.001

Table 2 Memorization of each model (full)

Model	D_1 [%]	D_2 [%]	D [%]
M_1	5.4	1.7	3.4
M_2	3.6	4.0	3.6
M_{12}	3.0	4.7	4.3

Table 3 Memorization of each model (adapter)

Model	D_1 [%]	D_2 [%]	D [%]
M_1	4.2	2.0	3.2
M_2	3.9	1.4	2.8
M_{12}	3.3	2.6	3.0

Table 4 Perplexity of each model (full)

Model	D_1	D_2	D
M_1	7.23	6.27	6.72
M_2	6.81	5.88	6.31
M_{12}	6.70	5.79	6.21

Table 5 Perplexity of each model (adapter)

Model	D_1	D_2	D
M_1	7.72	6.71	7.18
M_2	7.71	6.71	7.18
M_{12}	7.32	6.31	6.78

It can be seen that the proposed method has higher perplexity than the conventional method. In addition, as can be inferred from the memorization results in the Adapter, there is no change in the perplexity with respect to D_2 through two rounds of fine-tuning.

Table 6 Hyperparameters in additional experiment 1

Model	Epoch	LR
M_1 (Full)	5	0.0001
M_2 (Full)	9	$2 * 10^{-5}$
M_1 (Adapter)	18	0.001
M_2 (Adapter)	20	0.001

Table 7 Memorization for each model in additional experiment 1 (full)

Model	D_2 [%]	D_1 [%]	D [%]
M_1	7.9	2.2	3.3
M_2	5.1	2.7	3.7

Table 8 Memorization for each model in additional experiment 1 (adapter)

Model	D_2 [%]	D_1 [%]	D [%]
M_1	4.8	2.5	3.0
M_2	2.2	3.9	3.3

Table 9 Perplexity in additional experiment 1 (full)

Model	D_2	D_1	D
M_1	6.30	7.34	6.78
M_2	5.89	6.82	6.32

5.6 Additional Experiment 1: Reordering of Datasets

Next, an additional experiment will be conducted to investigate memorization trends. First, we will conduct an experiment to see if there is a difference in memorization propensity depending on the order in which the split data sets were trained. That is, D_2 and D_1 were trained in that order. The hyperparameters are as shown in Table 6. Memorization is shown in Tables 7 and 8 and perplexity is shown in Tables 9 and 10.

Memorization increased and perplexity decreased compared to the previous experiment. This time, the memorization is higher than M_{12} and the perplexity is also higher, and the overall forgetting of the data set is not confirmed.

Table 10 Perplexity in additional experiment 1 (adapter)

Model	D_2	D_1	D
M_1	6.78	7.92	7.31
M_2	6.51	7.50	6.97

5.7 Additional Experiment 2: Forgetting Effect from High Memorization

Next, we will verify what kind of forgetting curve will be drawn by the second fine-tuning from the state in which memorization was made close to 100% in the first training. Since it was not possible to make memorization close to 100% with the Adapter with a reduction_factor of 16, the experiment will be conducted in Full only.

In this experiment, learning is performed with a learning rate of 0.001 and epoch 10 to create M_1 with memorization of 94.0% and perplexity of 24.2, and further learning is performed using D_2 to check how D_1 is forgotten. At this time, M_1 is in the overfitting phase, and this parameter is used to maintain perplexity.

In this experiment, three learning rates of {0.001, 0.0001, 2×10^{-5} } were used to create M_2 . We show the result for memorization in Fig. 2. Basically, as the epoch progresses, the memorization of D_1 decreases, and the memorization of D_2 increases.

We also show the results for the perplexity of the test data in Fig. 3. It can be seen that forgetting varies with the learning rate. When the forgetting progresses to the 10% range, the perplexity is not maintained. This is due to the overfitting of D_2 before the forgetting of D_1 . By setting a smaller learning rate, such as 2×10^{-5} , we are able to maintain perplexity, although the forgetting is about 20%.

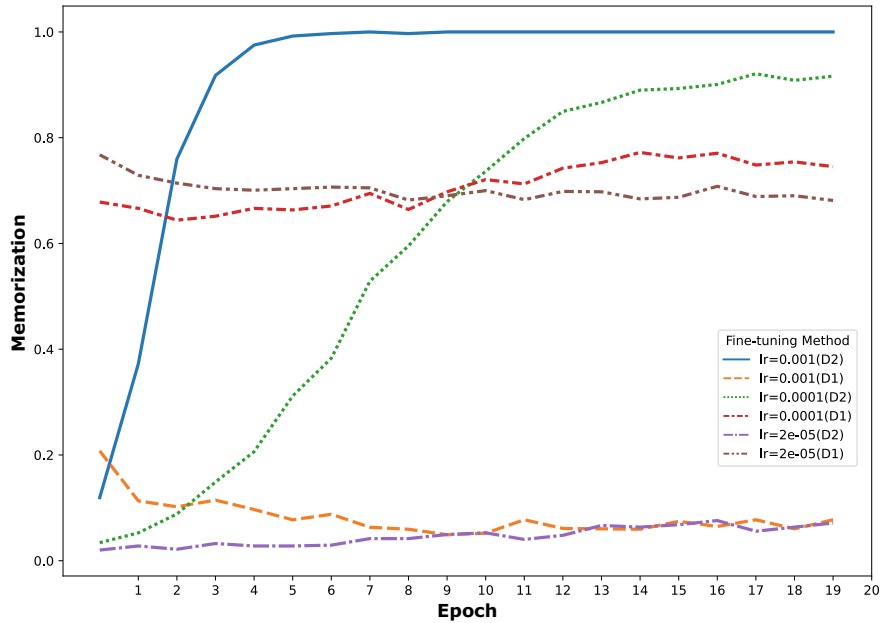


Fig. 2 Additional Experiment 2: Transition of Memorization for D_1 and D_2

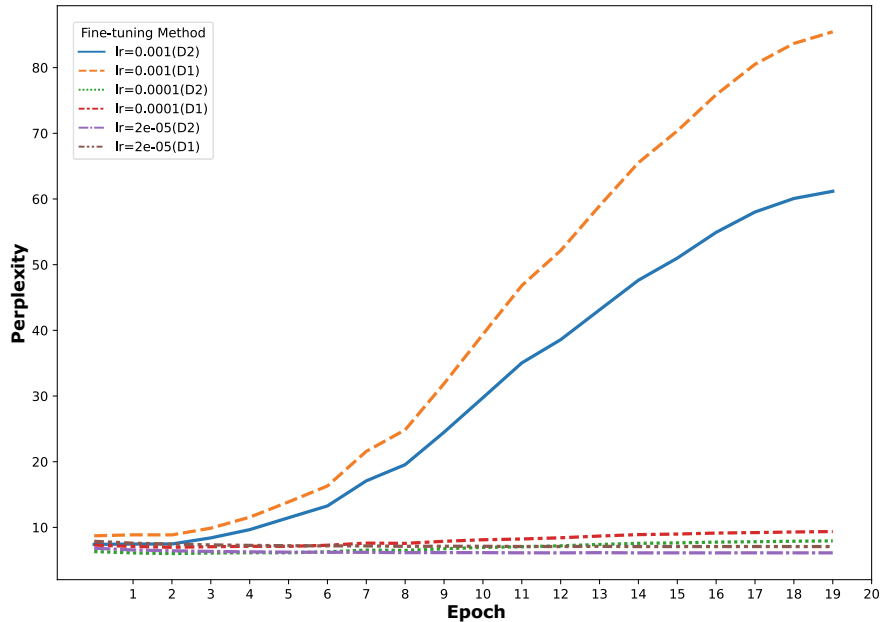


Fig. 3 Additional Experiment 2: Transition of Perplexity for D_{1_v} and D_{2_v}

Table 11 Hyperparameters in additional experiment 3

Model	Epoch	LR
M_1 (Adapter)	18	0.0001
M_2 (Adapter)	14	0.0001

Table 12 Memorization for each model in additional experiment 3 (adapter)

Model	D_1 [%]	D_2 [%]	D [%]
M_1	4.6	2.2	3.5
M_2	3.3	4.7	3.6

5.8 Additional Experiment 3: Adapter Reduction Factor Set to 2

Finally, the experiment was conducted with an increased number of parameters in the Adapter layer. The hyperparameters are as shown in Table 11.

Memorization is shown in Table 12 and perplexity is shown in Table 13.

Table 13 Perplexity in additional experiment 3 (adapter)

Model	D_1	D_2	D
M_1	7.75	6.72	6.91
M_2	7.36	6.31	6.80

These results confirm that increasing the parameters facilitates data memorization. On the other hand, we could not confirm forgetting while maintaining perplexity.

6 Conclusion

In this study, we experimentally tested whether a two-step fine-tuning can reduce memorization in the entire data set. As a result, although forgetting was confirmed under some conditions, the perplexity was higher than existing methods, making it difficult to say that it is an effective method. In addition, the forgetting effect itself could not be observed under different conditions, so it is not a method that can be used universally.

Acknowledgements This research was funded by JSPS international scientific exchanges between Japan and India, Bilateral Program DTS-JSP, grant number JPJSBP120227718.

References

1. Radford A, Narasimhan K (2018) Improving language understanding by generative pre-training. Accessed August 23, 2023, from https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf
2. Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I (2019) Language models are unsupervised multitask learners. Accessed August 23, 2023, from https://d4mucfpksyv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf
3. Houshy N, Giurghi A, Jastrzebski S, Morrone B, De Laroussilhe Q, Gesmundo A, Gelly S (2019) Parameter-efficient transfer learning for NLP. In: International conference on machine learning. PMLR, pp 2790–2799
4. Mireshghallah F, Goyal K, Uniyal A, Berg-Kirkpatrick T, Shokri R (2022) Quantifying privacy risks of masked language models using membership inference attacks. arXiv preprint [arXiv:2203.03929](https://arxiv.org/abs/2203.03929)
5. Mireshghallah F, Uniyal A, Wang T, Evans D, Berg-Kirkpatrick TS (2022) Memorization in NLP fine-tuning methods. arXiv preprint [arXiv:2205.12506](https://arxiv.org/abs/2205.12506)
6. Tirumala K, Markosyan A, Zettlemoyer L, Aghajanyan A (2022) Memorization without overfitting: analyzing the training dynamics of large language models. Adv Neural Inf Process Syst 35:38274–38290
7. Carlini N, Liu C, Erlingsson Ú, Kos J, Song D (2019) The secret sharer: evaluating and testing unintended memorization in neural networks. In: Proceedings of the 28th USENIX conference on security symposium (SEC’19)
8. Carlini N, Tramèr F, Wallace E, Jagielski M, Herbert-Voss A, Lee K, Roberts A, Brown TB, Song DX, Erlingsson Ú, Oprea A, Raffel C (2020) Extracting training data from large language models. In: USENIX security symposium
9. Carlini N, Ippolito D, Jagielski M, Lee K, Tramèr F, Zhang C (2022) Quantifying memorization across neural language models. arXiv preprint [arXiv:2202.07646](https://arxiv.org/abs/2202.07646)
10. Carlini N, Jagielski M, Zhang C, Papernot N, Terzis A, Tramèr F (2022) The privacy onion effect: memorization is relative. Adv Neural Inf Process Syst 35:13263–13276
11. Jagielski M, Thakkar O, Tramèr F, Ippolito D, Lee K, Carlini N, Wallace E, Song S, Thakurta A, Papernot N Others (2022) Measuring forgetting of memorized training examples. arXiv preprint [arXiv:2207.00099](https://arxiv.org/abs/2207.00099)

12. Shokri R, Stronati M, Song C, Shmatikov V (2017) Membership inference attacks against machine learning models. In: IEEE symposium on security and privacy (SP)
13. Murakonda SK, Shokri R, Theodorakopoulos G (2021) Quantifying the privacy risks of learning high-dimensional graphical models
14. Ye J, Maddi A, Murakonda SK, Bindschaedler V, Shokri R (2022) Enhanced membership inference attacks against machine learning models. In: Proceedings of the 2022 ACM SIGSAC conference on computer and communications security (CCS '22). Association for Computing Machinery, pp 3093–3106
15. Dwork C (2006) Differential privacy. In: Bugliesi M, Preneel B, Sassone V, Wegener I (eds) Automata, languages and programming. ICALP 2006. Lecture notes in computer science, vol 4052. Springer, Berlin, pp 1–12
16. Abadi M, Chu A, Goodfellow I, McMahan BH, Mironov I, Talwar K, Zhang L (2016) Deep learning with differential privacy. In: Proceedings of the 2016 ACM SIGSAC conference on computer and communications security (CCS '16)
17. Lukas N, Salem A, Sim R, Tople S, Wutschitz L, Zanella-Béguelin S (2023) Analyzing leakage of personally identifiable information in language models. arXiv preprint [arXiv:2302.00539](https://arxiv.org/abs/2302.00539)
18. Diera A, Lell N, Garifullina A, Scherp A (2022) A study on extracting named entities from fine-tuned vs. differentially private fine-tuned BERT models. arXiv preprint [arXiv:2212.03749](https://arxiv.org/abs/2212.03749)
19. Kandpal N, Wallace E, Raffel C (2022) Deduplicating training data mitigates privacy risks in language models. In: International conference on machine learning. PMLR, pp 10697–10707
20. Lee K, Ippolito D, Nystrom A, Zhang C, Eck D, Callison-Burch C, Carlini N (2021) Deduplicating training data makes language models better. arXiv preprint [arXiv:2107.06499](https://arxiv.org/abs/2107.06499)
21. Pfeiffer J, Rücklé A, Poth C, Kamath A, Vulić I, Ruder S, Cho K, Gurevych I (2020) Adapterhub: a framework for adapting transformers. arXiv preprint [arXiv:2007.07779](https://arxiv.org/abs/2007.07779)
22. Goodfellow IJ, Mirza M, Xiao D, Courville A, Bengio Y (2013) An empirical investigation of catastrophic forgetting in gradient-based neural networks. arXiv preprint [arXiv:1312.6211](https://arxiv.org/abs/1312.6211)
23. Klimt B, Yang Y (2004) The enron corpus: a new dataset for email classification research. In: European conference on machine learning

Automatic Wound Segmentation with Deep Convolutional Neural Networks



Jasmin T. Ngoc Nguyen, Dun-Hao Chang, and Chien-Lung Chan

Abstract Accurate wound segmentation is a pivotal task in medical image analysis, aiding clinicians in efficient diagnosis and treatment outcome monitoring. In this paper, we introduce an innovative approach for fully automatic wound segmentation utilizing a hybrid architecture combining two U-Net architectures. The proposed model takes advantage on the strengths of original U-Net architecture to achieve enhanced accuracy in segmenting wound regions. Our proposed method achieved 97% accuracy, 0.97 precision, 0.93 recall, 0.90 IoU score, and 0.95 dice coefficient similarity score. We built an annotated wound image dataset consisting of 365 images from 110 patients to train and test our model. We demonstrated the effectiveness and mobility of our method by conducting comprehensive experiments and analyses on various segmentation neural networks. Tested on diversified wound types, our model offers robustness and adaptability for varying wound appearances. The result promises to strengthen clinical decision-making and treatment monitoring through improved wound assessment.

Keywords Pressure ulcers · Wound segmentation · Medical image analysis · Deep learning · Clinical decision-making

J. T. Ngoc Nguyen · D.-H. Chang · C.-L. Chan (✉)

Department of Information Management, Yuan Ze University, Taoyuan 320, Taiwan

e-mail: clchan@saturn.yzu.edu.tw

D.-H. Chang

Department of Plastic and Reconstructive Surgery, Far Eastern Memorial Hospital, New Taipei City 220, Taiwan

School of Medicine, National Yang Ming Chiao Tung University, Taipei 112, Taiwan

C.-L. Chan

Innovation Center for Big Data and Digital Convergence (InnoBic), Yuan Ze University, Taoyuan 320, Taiwan

ZDT Group - Yuan Ze University Joint R&D Center for Big Data, Taoyuan 320, Taiwan

Abbreviations

FCN	Fully Convolutional Networks
CNN	Convolutional Neural Networks
AI	Artificial Intelligence
AZH	Advancing the Zenith of Healthcare
FEMH	Far Eastern Memorial Hospital
mPA/PA	Mean pixel accuracy/pixel accuracy
mPS/PS	Mean pixel sensitivity/pixel sensitivity
mPP/PP	Mean pixel precision/pixel precision
mIoU	Mean intersection over union
mDSC	Mean dice similarity coefficient
ReLU	Rectified Linear Unit

1 Introduction

Most recent advances in the healthcare field have been made possible by integrating deep learning and medical imaging. The segmentation of wounds, especially pressure ulcers, is one area that has substantially benefited from these discoveries. Pressure ulcers, commonly referred to as bedsores or decubitus ulcers, present a serious healthcare concern, especially for individuals with restricted mobility or those who spend a lot of time in bed. These wounds can lead to severe complications, such as infections and prolonged hospital stays, posing both medical and economic challenges [1].

Wound image segmentation, involving the separation of the wound from background and other parts of the body, is a crucial task for conducting further wound analyses like tissue classification, wound measurement, 3D reconstruction, or healing evaluation. The main objective of segmentation is to categorize the image into two classes, namely, wound, and non-wound [2]. Wound image segmentation has traditionally been done manually by healthcare personnel, which is not only time-consuming but also vulnerable to inter-observer variability. This diversity can result in uneven wound assessments, compromising patient treatment results.

Deep learning techniques, notably convolutional neural networks (CNNs) and semantic segmentation models have revolutionized medical image analysis, with the ability for automated and enhanced wound image segmentation operations. Deep learning algorithms have the capability to discern intricate patterns and features from extensive datasets, enabling them to accurately delineate the boundaries of pressure ulcers in medical imaging including photographs or scans. By automating this segmentation process, healthcare professionals can accomplish more consistent and reliable wound assessments, resulting in better patient care and outcomes.

However, applying deep learning to wound segmentation efficiently is not without hurdles. Because of changes in illumination, patient location, and imaging modalities, medical images have intrinsic unpredictability. Furthermore, gathering annotated medical image datasets for training can be both time-consuming and resource intensive. To deal with these issues, significant consideration must be given to data preprocessing techniques, model architecture selection, and optimization methods.

The purpose of this study is to investigate and demonstrate the efficacy of deep learning in segmenting wounds from medical images. We hope to construct a robust and accurate model that can assist healthcare practitioners in automating the wound segmentation process by leveraging cutting-edge convolutional neural networks and semantic segmentation techniques. The findings of this study hold the potential to enhance patient care by alleviating manual workload and argument the overall efficiency of wound assessment and treatment planning.

The remainder of this paper is organized as follows: Sect. 2 provides an overview of the related works on wound segmentation, Sect. 3 justifies the methodology, including data gathering and preprocessing, model architecture, and training procedures. Section 4 addresses the results from the proposed method on our dataset followed by comprehensive analysis. Finally, Sect. 5 concludes the study by summarizing key findings and highlighting the implications of our work in the context of wound segmentation and its practical implications.

2 Related Works

The landscape of wound image analysis has been revolutionized by the integration of image processing techniques and artificial intelligence (AI) advancements. Recent research in the fields of wound segmentation and wound condition analysis has witnessed remarkable progress, ushering in a new era of efficient and accurate wound assessment.

Pioneering the application of deep learning to wound segmentation and analysis, Wang et al. introduced a groundbreaking approach that harnessed the power of deep convolutional neural networks (CNNs) in 2015 [3]. Their innovative framework combined automated wound segmentation with subsequent analysis, offering a comprehensive solution. By leveraging a multi-scale CNN architecture and introducing a novel contour refinement strategy, the authors achieved impressive levels of accuracy in wound segmentation. This work not only constitutes a significant leap forward in medical image analysis but also aligns with the broader trend of employing deep learning techniques for intricate medical image segmentation tasks.

In a subsequent study, Cui et al. [4] presented a method for automated wound region segmentation from skin images of diabetic patients, utilizing convolutional neural networks (CNNs). Their approach encompassed a series of key steps, including pre-processing to enhance image quality and reduce the impact of varying illuminations and noise. The method employed two distinct CNN models to generate

a probability map, guiding wound area determination, followed by rigorous post-processing involving thresholding, connected component analysis, and rule-based refinement techniques. Comparative analysis against baseline methods highlighted the superiority of their approach in terms of both segmentation quality and efficiency.

Extending the scope of wound segmentation research, a study by Wang et al. [5] employed the Advancing the Zenith of Healthcare (AZH) Wound and Vascular Center dataset, a valuable resource with a substantial image volume exceeding 1200 images. The authors introduced a depth-wise separable convolutional neural network [6] with MobileNetV2 [7] as the backbone, bolstered by a post-processing mechanism to address segmentation imperfections. Their proposed method demonstrated exceptional accuracy, surpassing other state-of-the-art models in wound segmentation on both their dataset and the Medetec Wound dataset. Importantly, the approach's efficiency and adaptability to resource-limited environments, such as mobile devices, were underscored, making it a promising tool for real-world applications.

In a different vein, Scebbba et al. [8] introduced the Detect-and-Segment (DS) methodology, a three-step process encompassing wound bed detection, detection post-processing, and wound bed segmentation. This approach's performance was evaluated across multiple datasets [5, 9–11], demonstrating its versatility and robustness across various types of wounds. Notably, DS exhibited superior generalization capabilities, reducing the need for extensive training data.

Collectively, these studies represent significant advancements in the field of wound image analysis, offering diverse approaches that leverage AI and deep learning techniques for accurate segmentation and assessment. The convergence of innovation and practicality in these methods signals a promising direction for future research and applications in the field of medical image analysis and wound care.

3 Methodology

3.1 Data Preparation

The data preparation stage includes data collection, data pre-processing and data reading and splitting data into training, validating, and testing set.

Our own dataset was collected from Far Eastern Memorial Hospital (FEMH), Taiwan. Clinical wound images of Pressure Ulcers were retrieved from the digital database of the Plastic Surgery Department at the FEMH. Most of the images were taken by our nursing specialists, using a digital camera (Panasonic DMC-LX10 4 K). Some wound pictures were taken by various brands of cell phone or cameras. The usual environments of taking photos include operation room, wards and clinics. Some of the images were under- or over-exposed, blood or betadine-stained, obscured, and need to be excluded from analysis (as Fig. 1).

Images were collected more than 2 years (2019–2021) from 101 patients. Then, the doctor will do the annotation for wound area based on original collected images

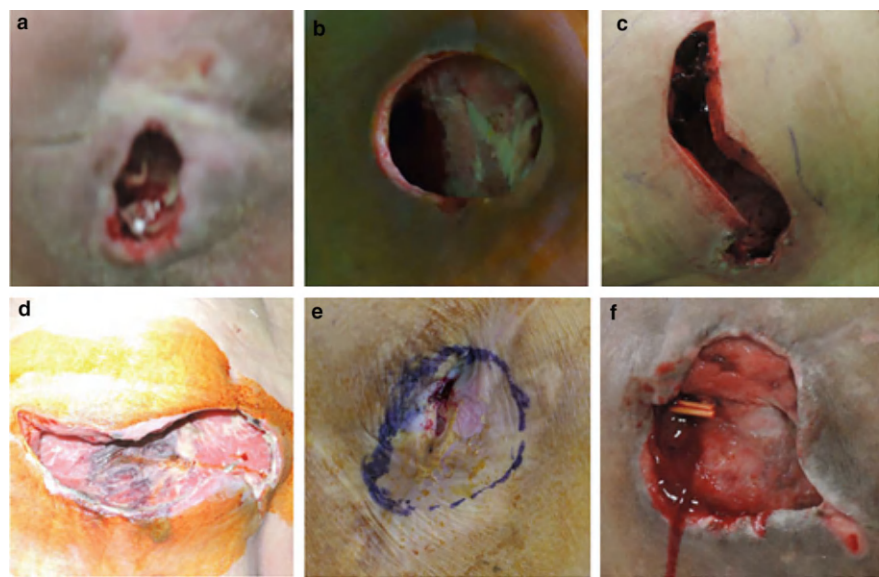


Fig. 1 Unqualified Pictures in collected dataset were removed. **a** Vague, **b** Outer shadows, **c** Inner Shadows, **d** Iodine stained, **e** Surgical Marking, **f** Bloodsheds

using a software called Labelme. These supplied annotation masks were formatted as numeric data with the extension “.json” (Fig. 2). We are going to utilize Labelme library integrated with the Python programming language to convert these json files into images with the “.png” file extension. Eventually, we had the data set shown in Fig. 3.

02.jpg	5/25/2022 10:12 PM	JPG File	25 KB
02.json	5/26/2022 3:56 PM	JSON File	18 KB
03.jpg	5/25/2022 10:12 PM	JPG File	124 KB
03.json	5/26/2022 4:02 PM	JSON File	64 KB
04.jpg	5/25/2022 10:12 PM	JPG File	131 KB
04.json	5/26/2022 4:03 PM	JSON File	63 KB
06.jpg	5/25/2022 10:12 PM	JPG File	29 KB
06.json	5/26/2022 4:04 PM	JSON File	22 KB
07.jpg	5/25/2022 10:12 PM	JPG File	155 KB
07.json	5/26/2022 4:05 PM	JSON File	89 KB
08.jpg	5/25/2022 10:12 PM	JPG File	105 KB
08.json	5/26/2022 4:06 PM	JSON File	50 KB
09.jpg	5/25/2022 10:12 PM	JPG File	72 KB
09.json	5/26/2022 4:07 PM	JSON File	44 KB

Fig. 2 Images and annotations preparation using Labelme. The “.jpg” files are original images while the “.json” files are the annotated images

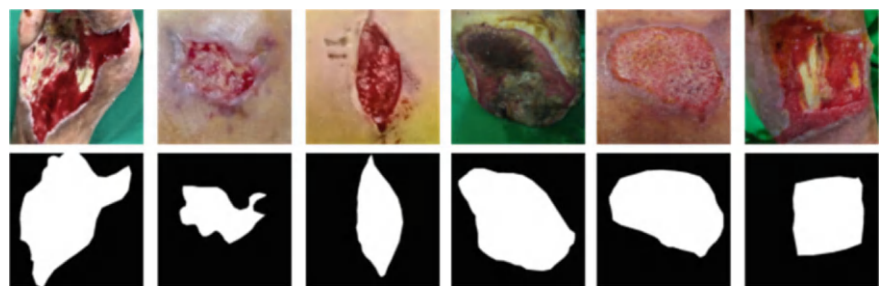


Fig. 3 Final dataset. Both images and corresponding annotated masks are in “.png” format

The dataset has a total of 365 images and corresponding annotation masks with size 512 pixels in both height and width. To prepare for experiments, we split 365 images into a training set of 292 images and a testing set of 73 images (based on a ratio of 80:20). To achieve better performance of the model, we used data augmentation techniques on our dataset before feeding it into the deep neural network model. These image transformations include arbitrary rotations in the $+90$ to -90° range, random left-right and top-down flipping with a probability of 1.0, center cropping, and grid distortion with a probability of 1.0 [12]. To generate convincing research findings, the training set is eventually expanded to 1752 images while the test dataset is left unchanged. Figure 4 displays some example images undergoing image augmentation.

Before feeding data into model for training process, training data after augmentation would be split into training set and validation set with the ratio 80:20. Finally, we had the dataset with the structure shown in Table 1. After that, data augmentation was applied to the training set. We continue to split the augmented training set into sub-training and validation set.

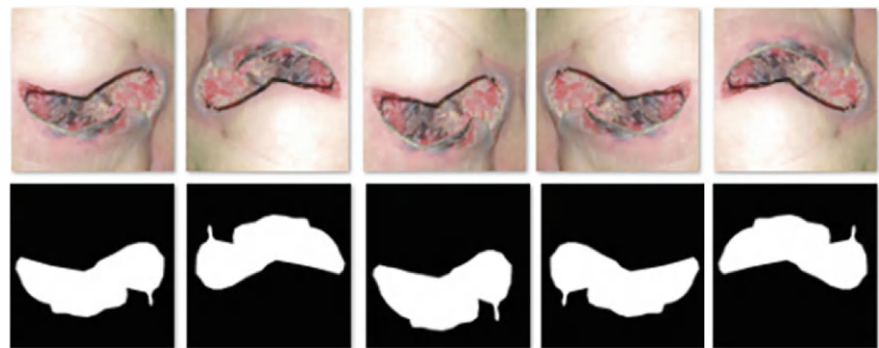


Fig. 4 Example images and masks undergoing image augmentation

Table 1 Data set division layout. The original images in the dataset were separated into training set and testing set with ratio 80:20

	Number of images	Data augmentation
Training set	292	1402
Validation set	–	350
Testing set	73	–

3.2 Model Architecture

The DoubleU-Net architecture employs two U-Net architectures in succession, each with two encoders and two decoders. The network’s first encoder is pre-trained EfficientNet [13], which was trained on ImageNet. The inventors also employ Atrous Spatial Pyramid Pooling (ASPP) [14]. As shown in Fig. 5, the architecture of DoubleU-Net begins with an EfficientNetB0 as an encoder subnetwork, followed by a decoder subnetwork. The usage of EfficientNetB0 marked in green, ASPP marked in red, and decoder block marked in light blue clearly delineates DoubleU-Net from U-Net in the first network (NETWORK 1). The squeeze-and-excite block [15] is part of the NETWORK 1 encoder and the NETWORK 1 and NETWORK 2 decoder blocks. When comparing NETWORK 1’s output to its input, an element-wise multiplication is done.

In the second network (NETWORK 2), the only difference between DoubleU-Net and U-Net is the use of ASPP and the squeeze-and-excite block. All other components remain unchanged. The input image is fed into the modified U-Net in NETWORK 1, which generates a predicted mask (Output1). The input image and the generated mask are then multiplied (Output1), which serves as an input for the second modified U-Net, which produces another mask (Output2). Finally, they combine both masks (Output1 and Output2) to examine the qualitative difference between the intermediate mask (Output1) and the final predicted mask (Output2). The primary reason for employing two U-Net models in the DoubleU-Net design is they are able to improve the generated output from Network 1 by retrieving the image and its respective mask and concatenating it with Output2 to produce a better segmentation result than the previous one.

3.3 Evaluation Metrics

Mean Pixel Metrics

The mean pixel metrics are derived from the pixel-level confusion matrix [16]. The confusion matrix stores four values: True Positives (number of pixels predicted to belong to a positive class that actually belong to that positive class), True Negatives (number of pixels predicted to belong to a negative class that actually belongs to that negative class), False Positives (number of pixels incorrectly predicted to belong to a positive class), and False Negatives (number of pixels incorrectly predicted to

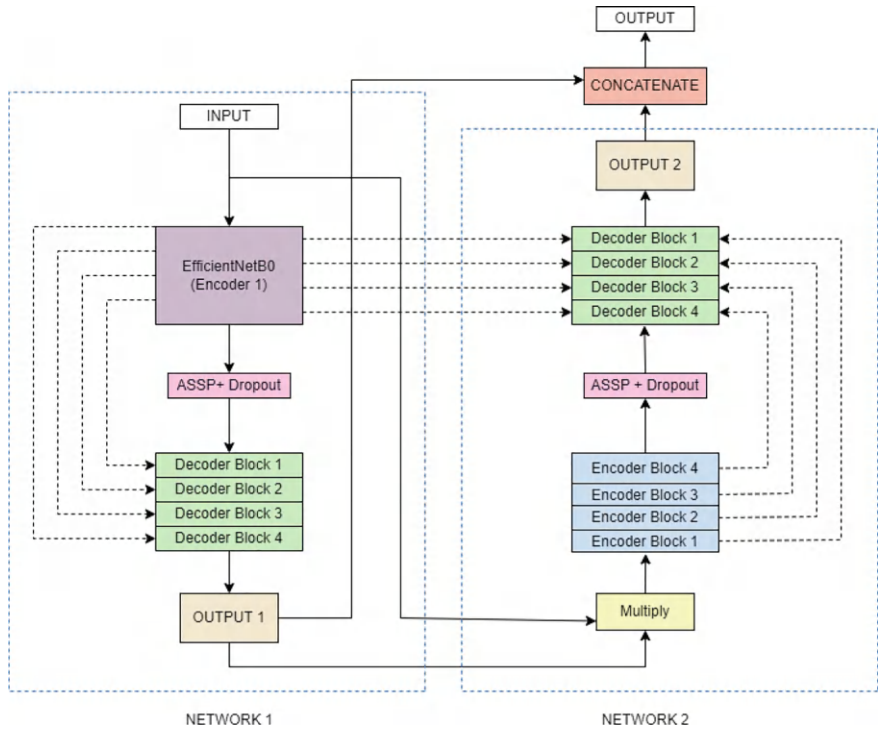


Fig. 5 DoubleU-Net architecture (encoder explanation). The first encoder in DoubleU-Net makes use of pre-trained EfficientNetB0, while the second encoder is constructed from scratch. Two 3×3 convolution layers are present in each encoder block of the second encoder, and each is followed by a batch normalization. Next, a Rectified Linear Unit (ReLU) activation function is applied following by a squeeze-and-excited block. Subsequently, a 2×2 window and stride 2 are used for max pooling in order to minimize the feature maps' spatial dimensions. (Decoder Explanation) Each block in the decoder part performs a 2×2 bi-linear upsampling on the input feature maps. The encoder's skip connection feature maps were appropriately concatenated with the decoder. Then, we execute two 3×3 convolution processes, with a batch normalization and a ReLU function in between. Next, a squeeze-and-excited block was performed. Finally, we apply a convolutional layer that generates the mask for the corresponding updated U-Net using a sigmoid activation function

belong to a negative class). We calculated the mean pixel accuracy (mPA), mean pixel sensitivity (mPS), and mean pixel precision (mPP) from the pixel confusion matrix. The PA and PS semantic segmentation metrics represent the percentage of correctly categorized and correctly predicted positive pixels in the total number of positive pixels in the image, respectively. Pixel precision (PP) is the percentage of correctly predicted positive pixels in the total number of predicted positive pixels in the image. Finding the common PA, PS, and PP in each clipped wound image yielded the mPA, mPS, and mPP. They are respectively computed as:

$$\text{mPA} = \frac{1}{N} \sum_{i=1}^N \frac{\text{TP}_i + \text{TN}_i}{\text{TP}_i + \text{TN}_i + \text{FP}_i + \text{FN}_i}. \quad (1)$$

$$\text{mPS} = \frac{1}{N} \sum_{i=1}^N \frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i}. \quad (2)$$

$$\text{mPP} = \frac{1}{N} \sum_{i=1}^N \frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i}. \quad (3)$$

where N is the number of images.

Mean Intersection-over-union (mIoU, Jaccard Index)

The Intersection-Over-Union (IoU), also known as the Jaccard Index, is one of the most commonly used metrics in semantic segmentation. It is the area of overlap between the predicted segmentation and the ground truth divided by the area of union between the predicted segmentation and the ground truth. This metric ranges from 0 to 1 (0–100%) with 0 signifying no overlap and 1 signifying perfectly overlapping segmentation. For binary (two classes) or multi-class segmentation, the mean IoU of the image is calculated by taking the IoU of each class and averaging them.

$$\text{mIoU} = \frac{1}{N} \sum_{i=1}^N \frac{X_i \cap Y_i}{X_i \cup Y_i} \quad (4)$$

Mean Dice Similarity Coefficient

The mean dice similarity coefficient (mDSC) is usually used for segmentation tasks, in which the predicted area is formed of pixels. The DSC ranges from 0 (no spatial overlap between the predicted and ground truth areas) to 1 (complete overlap of the two areas). The mean dice coefficient (mDC) is computed as the average over all images:

$$\text{mDSC} = \frac{1}{N} \sum_{i=1}^N \frac{2 * |X_i \cap Y_i|}{|X_i| + |Y_i|}. \quad (5)$$

where N is the number of objects X_i and Y_i are the predicted area Y_i is and ground-truth area box of the i th object, respectively.

4 Experiment and Results

In this model, the NETWORK1 was used transfer learning with EficientNetB0 backbone. DoubleU-Net was trained on randomly initializing the weights. The presented work’s deep learning model was built in Python using the Keras and Tensorflow backends. The models were trained on a 64-bit Windows PC with an 8-core 4.66 GHz CPU and a single NVIDIA GeForce RTX 3090 GPU to speed up the training process. We used the Adam optimization algorithm to modify the network parameters, which has gained wide acceptance in stochastic optimization due to faster convergence than other optimization functions. The loss function was dice coefficient loss, and the evaluation metrics were Accuracy, Precision, Recall, Intersection Over Union (IOU) score and the Dice similarity coefficient (DSC) score. In order to balance training accuracy and efficiency, the initial learning rate was set to 0.0001 and each mini-batch contained eight images. A single epoch training time took about 148 s. When there was no improvement in the validation dice score for more than 50 epochs, we used early stopping to stop the training. Our deep learning model was eventually trained for 300 epochs.

The DoubleU-Net model outperformed other models in terms of segmentation performance across all evaluation variables (Table 2). This model outperforms other models in terms of mean Pixel Precision (mPP), with a score of 0.97, indicating that it generates fewer false positives. Similarly, the proposed model received the highest mDSC of 0.95, implying higher overlap between predicted and ground truth masks, as well as the highest mIoU—0.90, revealing improved object localization.

In addition to our own dataset, we experimented on the Foot Ulcer Segmentation Challenge Dataset [3] and evaluated how well different techniques segmented the data. Table 3 presents the findings. Three networks were trained using the identical experimental setup and this dataset was labeled similarly to our dataset. The highest dice score is 0.88, and the greatest IoU score is 0.79 when utilizing our proposed model, DoubleU-Net. The performance evaluation aligns with the conclusion reached from our dataset, in which our method outperforms the other regardless of which type of chronic wound segmentation is employed, suggesting that our model is robust and unbiased.

Table 2 Segmentation Model Evaluation. Comparison of U-Net, Fully Convolutional Network (FCN) models trained on our wound dataset. mPA, mPP, mPS, mIoU, mDSC on test dataset figures describe the performance of each model on predicting the probability of wound pixel inside each wound image. * Indicates best-performing model. mPA, mean Pixel Accuracy; mPP: mean Pixel Precision; mPS, mean Pixel Sensitivity; mIoU, mean Intersection-over-Union; mDSC, mean Dice Similarity Coefficient

	mPA	mPP	mPS	mIoU	mDSC
U-Net	0.95	0.91	0.92	0.85	0.92
FCN	0.96	0.93	0.93	0.87	0.93
DoubleU-Net*	0.97	0.97	0.93	0.90	0.95

Table 3 The accuracy, precision, sensitivity, dice score, and IoU score evaluated using various models on the Foot Ulcer Segmentation Challenge Dataset. Bold values indicate the best performance among the present models

	mPA	mPP	mPS	mIoU	mDSC
U-Net	0.99	0.90	0.85	0.77	0.87
FCN	0.99	0.86	0.73	0.66	0.79
DoubleU-Net	0.99	0.89	0.88	0.79	0.88

5 Discussion and Conclusion

In this study, we attempted to solve the automated segmentation problem of pressure ulcer in our own collected dataset using deep learning. We conducted experiments and analyses on U-Net, FCN, and our model based on U-Net and EfficientNet architecture. In the comparison of these two famous model, our model has demonstrated its effectiveness in the field of image segmentation, especially in our own dataset, the proposed model achieved the highest performance. To enhance our proposed model’s capability, we also conducted an experiment on Foot Ulcer Segmentation Challenge dataset for both three models, and results showed that our model still has the best performance.

In the future, we plan to expand our work by segmenting different types of wound tissue and improve the model performance by using attention mechanisms to focus the model’s attention on important features within the wound area. In addition, we try to collect and label a larger dataset of wound images in diversity position to train our model. This will help the model to learn a more generalizable representation of wound tissue and improve its performance.

In general, the proposed model shows good segmentation performance, with high accuracy and coverage of wound areas. However, in certain predicted masks, they may display interruptions or contain small holes (Fig. 6). This may suggest that while the model is proficient at capturing the general wound region, it still requires refinement in handling fine details and small-scale features within the wound. Further optimization or post-processing techniques are necessary to address these specific challenges and enhance the model’s efficiency, particularly when dealing with sophisticated or complex wound patterns.

We believe that by resolving these issues, we will be able to develop a more robust model for wound segmentation. This model could be used to support clinicians in wound therapy, as well as providing valuable insights into wound healing.

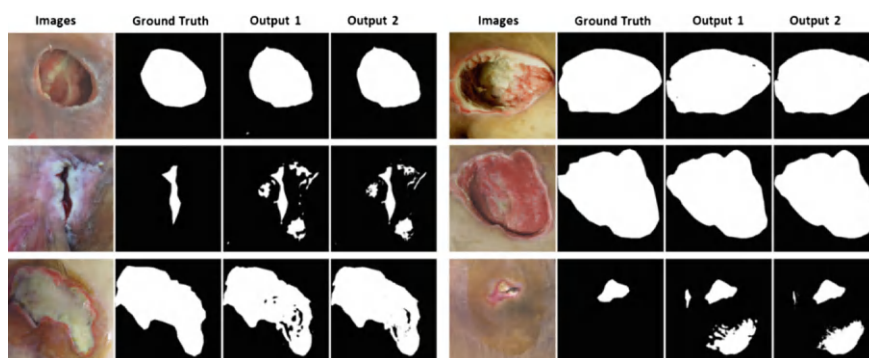


Fig. 6 Ulcer segmentation of image from FEMH dataset. The visualization includes input images, the annotated mask (ground truth), the predicted mask from Network 1 in proposed model architecture (Output 1), and the prediction segmentation map from the Network 2 (Output 2)

Acknowledgements We would like to express our sincere gratitude to the National Science and Technology Council (Taiwan), for their funding support of this study, grant number MOST 110WHK1210054. Additionally, we extend our appreciation to the funds for Far Eastern Memorial Hospital and Yuan Ze University collaboration project number FEMH-YZU-2023-015. This financial assistance has been instrumental in advancing our study and achieving our project goals.

Ethical Approval The study was approved by the Research Ethics Review Committee of FEMH (protocol code 110295-E, date of approval: 2020/11/25).

References

1. Zaidi SRH, Sharma S (2022) Pressure ulcer. <https://www.statpearls.com/point-of-care/20286>
2. Dhane DM, Krishna V, Achar A, Bar C, Sanyal K, Chakraborty C (2016) Spectral clustering for unsupervised segmentation of lower extremity wound beds using optical images. *J Med Syst* 40(9):207
3. Wang C, Yan X, Smith M, Kochhar K, Rubin M, Warren SM, Wrobel J, Lee H (2015) A unified framework for automatic wound segmentation and analysis with deep convolutional neural networks. In: 37th annual international conference of the IEEE engineering in medicine and biology society (EMBC)
4. Cui C, Thurnhofer-Hemsi K, Soroushmehr R, Mishra A, Gryak J, Dominguez E, Najarian K, Lopez-Rubio E (2019) Diabetic wound segmentation using convolutional neural networks. In: Annual international conference of the IEEE engineering in medicine and biology society, pp 1002–1005
5. Wang C, Anisuzzaman DM, Williamson V, Dhar MK, Rostami B, Niezgoda J, Gopalakrishnan S, Yu Z (2020) Fully automatic wound segmentation with deep convolutional neural networks. *Sci Rep* 10(1):21897
6. Chollet F (2017) Xception: deep learning with depth wise separable convolutions. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR)
7. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen LC (2018) MobileNetV2: inverted residuals and linear bottlenecks. In: 2018 IEEE/CVF conference on computer vision and pattern recognition

8. Scebbba G, Zhang J, Catanzaro S, Mihai C, Distler O, Berli M, Karlen W (2022) Detect-and-segment: a deep learning approach to automate wound image segmentation. *Inform Med Unlocked* 29:100884
9. Cassidy B, Reeves ND, Pappachan JM, Gillespie D, O'Shea C, Rajbhandari S, Maiya AG, Frank E, Boulton AJ, Armstrong DG, Najafi B, Wu J, Kochhar RS, Yap MH (2021) The DFUC 2020 Dataset: analysis towards diabetic foot ulcer detection. *touchREV Endocrinol* 17(1):5–11
10. Medtec, Medetec surgical dressings and wound management resource centre. <https://medetec.co.uk/>. Accessed 12 Dec 2023
11. Yang S, Park J, Lee H, Kim S, Lee BU, Chung KY, Oh B (2016) Sequential change of wound calculated by image analysis using a color patch method during a secondary intention healing. *PLoS ONE* 11(9):e0163092
12. Buslaev A, Iglovikov VI, Khvedchenya E, Parinov A, Druzhinin M, Kalinin AA (2020) Albumentations: fast and flexible image augmentations. *Information* 11(2):125. <https://doi.org/10.3390/info11020125>
13. Tan M, Le Q (2019) EfficientNet: rethinking model scaling for convolutional neural networks. In: Kamalika C, Ruslan S (eds) *Proceedings of the 36th international conference on machine learning*. PMLR: proceedings of machine learning research, pp 6105–6114
14. Chen LC, Papandreou G, Schroff F, Adam H (2017) Rethinking atrous convolution for semantic image segmentation. *abs/1706.05587*
15. Hu J, Shen L, Albanie S, Sun G, Wu E (2020) Squeeze-and-excitation networks. *IEEE Trans Pattern Anal Mach Intell* 42(8):2011–2023
16. Demir F (2022) Deep autoencoder-based automated brain tumor detection from MRI data. In: *Artificial intelligence-based brain-computer interface*. Academic Press, pp 317–351

Prediction of the Prevalence of COVID-19 Using Epidemic Differential Equations and Deep Learning Network



Liang Kong, Yanhui Guo, and Chung-Wei Lee

Abstract Prediction of the spread of Coronavirus Disease 2019 (COVID-19) is imperative for the efficient allocation and administration of national and worldwide healthcare. Traditional prediction models for COVID-19 pandemic are limited by their low accuracy because they assume homogeneous time-dependent transmission rates and isolate the study region in the absence of geographic features. To advance the prediction of COVID-19 spread, the parameters of the model should be refined via evolving understandings of the disease trajectory, transmission rates, and economic and social factors influencing infection. Hence, this study proposes a hybrid model combining the classic epidemic equations with a recurrent neural network (RNN) to predict the spread of the COVID-19 pandemic. The proposed model involves the number of patients who are Susceptible, Infectious, Recovered, and Deceased (SIRD) as time-dependent features and considers human mobility from neighboring regions as a spatial feature. This study derives a discrete-time function with the infection component of SIRD model for real-time application that avoids overfitting and is more efficient than other models. The proposed model was tested on a publicly available COVID dataset recorded from Italy. Experimental results show that it outperforms existing spatiotemporal models.

Keywords COVID-19 · Epidemic differential equations · Recurrent neural network · Prediction

L. Kong

Department of Mathematical Sciences and Philosophy, University of Illinois Springfield, Springfield, IL, USA

e-mail: lkong9@uis.edu

Y. Guo (✉) · C.-W. Lee

Department of Computer Science, University of Illinois Springfield, Springfield, IL, USA

e-mail: yguo56@uis.edu

C.-W. Lee

e-mail: clew84@uis.edu

1 Introduction

Coronavirus disease 2019 (COVID-19) is a global pandemic triggered by the emergence of a novel coronavirus that causes severe acute respiratory syndrome (SARS-CoV-2). Globally, as of December 10th, 2022, over 643 million confirmed cases of COVID-19, including over 6.6 million tragic deaths, have been reported to the World Health Organization (WHO) [1]. To mitigate the impact of COVID-19, collaborative efforts are needed to understand and predict the many complex facets of the spread of a previously unknown virus, beginning with the development of appropriate epidemic-spreading models and the investigation of non-pharmaceutical and pharmaceutical intervention policies [2–9]. Despite strict non-pharmaceutical controls (e.g., travel ban, mask mandate, gathering restrictions, closure of public transport and schools) and pharmaceutical measures (i.e., vaccination, novel antiviral drugs), throughout the pandemic, the number of coronavirus cases increased at an alarming rate in the United States and elsewhere [10]. Since December 2021, due to uncontrolled outbursts of new omicron variants, the COVID-19 pandemic has jumped globally to a new peak, surpassing the previous peak of April 2021 [11–13]. By July 2022, BA.5, an omicron subvariant of COVID-19, had emerged as one of the virus's dominant strains in the United States [14, 15]. Despite COVID-19-related infection cases and deaths recently decreasing worldwide, some countries, regions, and cities are experiencing an unprecedented pandemic peak. For instance, China is painfully experiencing unprecedented surging infections as it eased the COVID policy in early December 2022.

One of the primary goals of epidemiological modeling is to develop an efficient risk prediction model which allows public health workers and the government to assess the effectiveness of public health protocols and predict the risk of COVID-19 infection at the regional and national levels. However, because of the complexities of infection spread, prominent compartmental models have been unreliable [16–18]. Due to the paucity of well-curated data, it is challenging to develop a model while considering population density, the effect of viral mutations, human mobility under heterogeneous levels of lockdown, and qualitative social aspects such as culture and lifestyle [19, 20]. Despite these obstacles, it has been demonstrated that region-based mitigation factors such as lockdown level and monitoring have a significant impact on infection rate [21–23]. Very little work has been done using a combination of compartmental and deep learning-based prediction approaches for COVID-19. Our work aims to demonstrate the utility of differential equation models and deep learning methods for comprehending the pandemic and providing a workable framework for a larger group of quantitative researchers to predict COVID-19 infections. Because these models are parsimoniously designed, they are particularly well suited for identifying the pandemic's critical features and making policy-relevant discoveries with limited data.

In this study, we emphasize the significance of spatial heterogeneity for computational epidemiology. Specifically, we propose the development of a hybrid spatiotemporal epidemic model to forecast infection numbers by combining the susceptible-infected-recovered-deceased (SIRD) compartmental model with a long short-term memory (LSTM) network. Our spatiotemporal model considers human mobility by incorporating the infection rates of nearby neighbors. As a result, the hybrid spatial-temporal model produces superior predictions compared to individual conventional epidemiological or machine learning-based models, exhibiting increased prediction accuracy and reduced tendencies to over- or under-fit [24].

2 Proposed Model

To accurately and efficiently forecast the trend of COVID-19 spread, we propose a hybrid model incorporating the SIRD time-dependent features with human mobility effects through LSTM.

2.1 SIRD Compartmental Epidemiology Process Model

A variant of the Kermack-McKendrick equations can be used to model the dynamics of the disease. The classic SIR model divides a population into susceptible, infected, and recovered compartments to model disease spread. The SIRD model adds a death compartment to the SIR model to account for disease-induced mortality. By modeling deaths separately from recoveries, SIRD provides a more realistic representation of outbreaks of deadly diseases like COVID-19. The SIRD model enables analysis of the full disease burden and population impacts, better informing public health planning and interventions. In addition, we consider a geographical region that is assumed to be related to close neighbors and separated from other regions, and we define the following quantities $S(t)$ as the number of susceptible individuals at time t , $I(t)$ as the number of infected individuals at time t ; $R(t)$: the cumulative number of recovered individuals up to time t ; $D(t)$: the total number of individuals who deceased because of the disease up to time t .

We propose a novel spatiotemporal model that combines spatial information with a discrete-time Infection equation (I-equation) derived from the SIRD differential equations. While the I-equation models the observed temporal information, a recurrent neural network (RNN) is employed to characterize latent spatial information. The model, named SIRD-eRNN, differs from [22] in that a differential equation with 3 parameters (the I-equation) fits the limited temporal data, which makes the I-equation far more compact than the RNN. Equations describe the proposed SIRD-RNN model as:

$$\frac{dS}{dt} = -\beta SI \quad (1)$$

$$\frac{dI}{dt} = \beta SI - \gamma I - \mu I \quad (2)$$

$$\frac{dR}{dt} = \gamma I \quad (3)$$

$$\frac{dD}{dt} = \mu I \quad (4)$$

where β is the transmission rate, γ is the recovery rate, and μ is the death rate. In this model, the underlying hypothesis is that the recovered people are no longer susceptible to infection. The classic SIRD epidemiological model assumes a closed population and does not consider the importation of infections from outside the modeled region. It conserves the total population size, normalized to 1: $S+I+R+D = 1$.

2.2 Derivation of Discrete-Time I-Equation

We add a novel feature I_e that represents external infection as a result of human mobility from the neighbors of a region. The SIRD epidemic system with I_e as external forcing becomes:

$$\frac{dS}{dt} = -\beta_1 SI - \beta_2 SI_e \quad (5)$$

$$\frac{dI}{dt} = \beta_1 SI + \beta_2 SI_e - \gamma I - \mu I \quad (6)$$

$$\frac{dR}{dt} = \gamma I \quad (7)$$

$$\frac{dD}{dt} = \mu I \quad (8)$$

It follows from (7) and (8) that:

$$R(t) = R(t_0) + \gamma \int_{t_0}^t I d\tau \quad (9)$$

$$D(t) = D(t_0) + \mu \int_{t_0}^t I d\tau \quad (10)$$

Therefore,

$$S(t) = 1 - I(t) - R(t) - D(t) = 1 - I(t) - R(t_0) - \gamma \int_{t_0}^t I d\tau - D(t_0) - \mu \int_{t_0}^t I d\tau \quad (11)$$

By substituting $S(t)$ into (6) we have,

$$\begin{aligned} \frac{dI}{dt} &= (\beta_1 I + \beta_2 I_e) S - \gamma I - \mu I = (\beta_1 I + \beta_2 I_e) \\ &\quad \left(1 - I(t) - R(t_0) - \gamma \int_{t_0}^t I d\tau - D(t_0) - \mu \int_{t_0}^t I d\tau \right) - \gamma I - \mu I \end{aligned} \quad (12)$$

Approximating the integral by Riemann sum and applying the forward Euler method, the discrete approximation is written as:

$$\begin{aligned} I(t+1) &= S - \gamma I - \mu I = (1 - \gamma - \mu)I(t) + (\beta_1 I(t) + \beta_2 I_e(t)) \\ &\quad \cdot \left(1 - I(t) - R(t_0) - D(t_0) - \frac{(\gamma + \mu)(t - t_0)}{p+1} \sum_{j=0}^p I(t-j) \right) \end{aligned} \quad (13)$$

where p is the number of days that are considered before the time t .

At the beginning of the infection of the model, when $t_0 = 0$, $R(t_0) = 0$ and $D(t_0) = 0$. The discrete-time I-equation is simplified as:

$$\begin{aligned} I(t+1) &= S - \gamma I - \mu I = (1 - \gamma - \mu)I(t) + (\beta_1 I(t) + \beta_2 I_e(t)) \\ &\quad \cdot \left(1 - I(t) - (\gamma + \mu) \frac{t}{p+1} \sum_{j=0}^p I(t-j) \right) \end{aligned} \quad (14)$$

If we let $I_e(t) = 0$, then we have an approximation of $I(t)$ for the original SIRD model, which is a pure temporal model, named I-model:

$$\begin{aligned} I(t+1) &= S - \gamma I - \mu I = (1 - \gamma - \mu + \beta)I(t) - \beta I^2(t) \\ &\quad - \beta(\gamma + \mu) \frac{t}{p+1} I(t) \sum_{j=0}^p I(t-j) \end{aligned} \quad (15)$$

It is difficult to predict how a region's population interacts with those of adjacent areas that affects the infection speed. As a result, $I_e(t)$ is latent information that is difficult to model mathematically. We use recurrent neural networks with long short-term memory (LSTM) cells of RNN to extract latent spatial information I_e .

2.3 Edge Feature and Latent I_e

The spatial information is used to define the latent information $I_e^v(t)$ in a region v . Let C be the neighbors of region v . The spatial feature of v at time t is defined as:

$$f_e^v(t) = \frac{1}{|C|} \sum_{i:v_i \in C}^p [I_i(t-1), \dots, I_i(t-p)] \quad (16)$$

where $I_i(t)$ is the infection population of the region v_i at time t . An RNN is employed to estimate the value of $I_i(t)$ based on its spatial feature $f_e^v(t)$. We feed $f_e^v(t)$ into an edge RNN, which includes three stacked LSTM cells with a following dense layer to output I_e . We name our model SIRD-eRNN as a result of its integrated design of SIRD equations and edge RNN.

2.4 Training the Proposed Model

The discrete approximation of the proposed model is written as:

$$\begin{aligned} \hat{y}(t) = & S - \gamma I - \mu I = (1 - \gamma - \mu)I(t-1) \\ & + (\beta_1 y(t-1) + \beta_2 I_e(t-1)) \\ & \left(1 - y(t-1) - (\gamma + \mu) \frac{t}{p+1} \sum_{j=1}^{p+1} y(t-j) \right) \end{aligned} \quad (17)$$

To train the RNN model, the loss function is defined using the mean squared error (MSE) of data and the model's outputs as:

$$\text{Loss} = \frac{1}{T-p-1} \sum_{t=p+1}^T (y(t) - \hat{y}(t))^2 \quad (18)$$

Training consists of minimizing the loss function over parameters in both the infection equation and the recurrent neural network, hence the two components (I-equation and RNN) of the model are fully coupled while learning from data. Weights of the RNN and the dense layer, in addition to I-equation parameters β_1 , β_2 , and γ , are learned using Adam gradient descent.

3 Experimental Results

Retrospective estimates of Covid-19 activity were produced using our SIRD-eRNN model for the period of 2020-02-24 to 2020-06-18 of 20 Italian regions with the new confirmed cases reported on each date in each region. The isolated regions that have less than three neighbor regions are excluded. The seven regions used in the experiment are Campania, Emilia-Romagna, Lazio, Lombardia, Molise, Piemonte, and Umbria.

We compare SIRD-eRNN's estimates with the ground truth: the daily reported Covid-19 new (current) cases [25] and calculate the Root Mean Squared Error (RMSE) as an evaluation metric. We set $p = 3$ in Eq. (14), and we split the current data for 113 days into 80 days to train and 33 days to test our model, which is a 7:3 training/testing ratio.

All experiments were implemented using the software Python 3.7 and TensorFlow 1.5 and run on a server with an Intel Xeon processor having 128 GB memory and an NVIDIA Tesla K40 GPU having 12 GB memory.

Figure 1 presents a comparative analysis of daily confirmed COVID-19 cases and their corresponding forecasted values across seven distinct regions (labeled from (a) to (g)). The x-axis represents time in days, while the y-axis represents the number of COVID-19 cases normalized by the population of each respective region, providing a per capita perspective on the data. The predictions of the SIRD-eRNN model are shown in Fig. 1 with a solid orange line. It shows that the proposed model is able to estimate the infection rate accurately in the training and testing stages and capture similar dynamics of the transmission as the real ground truth rate. Compared to the prediction results of IeRNN model [22] in Fig. 2, the SIRD-eRNN is more like the ground truth rate in most regions except the Umbria region.

We examined our model predictions using root mean square error (RMSE) which is defined as:

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^T (\hat{y}_t - y_t)^2}{T}} \quad (19)$$

where y_t and $\hat{y}_t$ are real and predicted values representing the number of COVID-19 cases normalized by the population of each respective region at time t , respectively. T is the total number of samples. The smaller of RMSE, the better the prediction results.

In Table 1 on the three-day ahead forecast, SIRD-eRNN achieves smaller RMSE errors than the IeRNN model except for one region. The average of the RMSE in the testing stage for all regions is $3.19314\text{E-}4$ and less than the result of IeRNN, $3.23159\text{E-}4$. Except for the Umbria region, the average of the RMSE in the testing stage is $3.62327\text{E-}4$ for the SIRD-eRNN and $3.72205\text{E-}4$ for the IeRNN model. The compact and simple I-Model with three parameters performs inadequately for a 3-day ahead prediction compared to the proposed model. It is noted that an infection

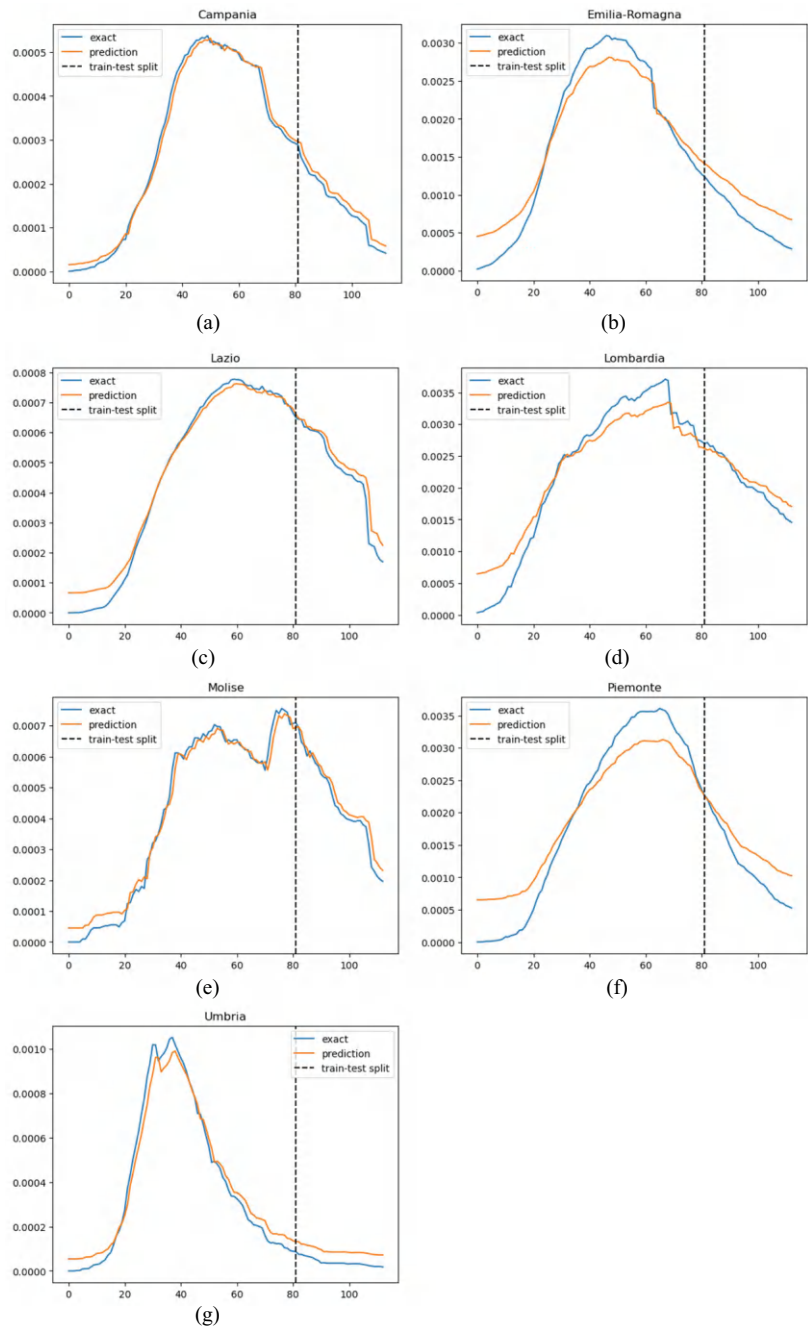


Fig. 1 Prediction results of the SIRD-eRNN model in different regions

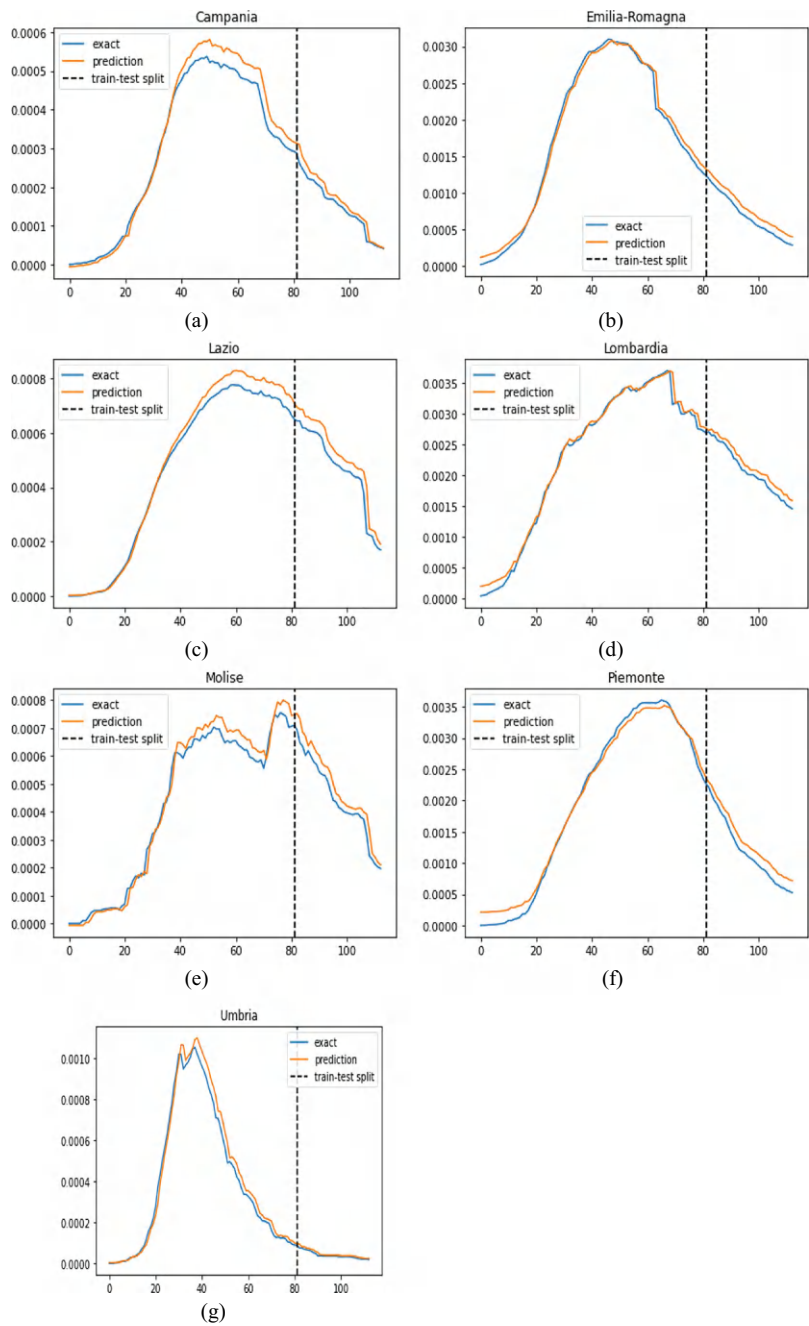


Fig. 2 Prediction results of the IeRNN model in different regions

Table 1 Comparison of RMSE between SIRD-eRNN and IeRNN using 3-day ahead forecast

State	RMSE			
	SIRD-Ernn		IeRNN	
	Training	Test	Training	Test
Campania	0.00027	0.00009	0.00029	0.00010
Lazio	0.00041	0.00020	0.00044	0.00021
Lombardia	0.00150	0.00049	0.00168	0.00054
Molise	0.00037	0.00019	0.00041	0.00021
Piemonte	0.00164	0.00071	0.00181	0.00074
Umbria	0.00046	0.00006	0.00050	0.00003
Average	0.00085	0.00032	0.00094	0.00032
Average without Umbria	0.00092	0.00036	0.00101	0.00037

population rate is a percentage number, and its base number is huge. Even small differences in RMSE will cause a significant difference in the estimated rate.

Since the nonlinearity of the infectious disease evolution, we compare nonlinear modes for three-day ahead prediction. The results presented here show the superiority of our model in terms of accuracy compared to IeRNN. Our SIRD-eRNN model can outperform other models and make accurate predictions.

As indicated in Table 1, our proposed SIRD-RNN does exhibit slightly better performance compared to IeRNN. While the improvement may not be substantial at first glance, it is indicative of the effectiveness of the modifications we have made to the existing approach. These improvements could potentially translate into enhanced predictive accuracy, more robust generalization, or better capturing of underlying dynamics within the system.

4 Conclusion

In this study, we propose a mixed model that combines the I-equation from the Susceptible, Infectious, Recovered, and Deceased (SIRD) model with spatial features to create a unique spatiotemporal infectious disease model. Its accurate estimation makes it suitable for real-time forecasting of COVID-19 and aiding authorities in making decisions. Our model outperforms standard temporal models in three-day ahead prediction scenarios. The future direction is to study: (1) human mobility information, such as airline data, for understanding transmissions and interactions beyond geographic neighbors; (2) COVID-19 data from other countries of the world; (3) more sophisticated differential equation models, which allows for the loss of immunity, treatment and prevention of disease, and population turnover.

References

1. WHO coronavirus (COVID-19) dashboard. <https://covid19.who.int/>. Accessed 20 Sep 2022
2. Estrada E (2020) COVID-19 and SARS-CoV-2. Modeling the present, looking at the future. *Phys Rep* 869:1–51. <https://doi.org/10.1016/j.physrep.2020.07.005>
3. Gallo L, Frasca M, Latora V, Russo G (Jan 2022) Lack of practical identifiability may hamper reliable predictions in COVID-19 epidemic models. *Sci Adv* 8(3). <https://doi.org/10.1126/sciadv.abg5234>
4. Huang C et al (2020) Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. *The Lancet* 395(10223):497–506. [https://doi.org/10.1016/S0140-6736\(20\)30183-5](https://doi.org/10.1016/S0140-6736(20)30183-5)
5. Chen N et al (2020) Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in Wuhan, China: a descriptive study. *Lancet* 395(10223):507–513. [https://doi.org/10.1016/S0140-6736\(20\)30211-7](https://doi.org/10.1016/S0140-6736(20)30211-7)
6. Wiersinga WJ, Rhodes A, Cheng AC, Peacock SJ, Prescott HC (2020) Pathophysiology, transmission, diagnosis, and treatment of coronavirus disease 2019 (COVID-19). *JAMA* 324(8):782. <https://doi.org/10.1001/jama.2020.12839>
7. Wang L, Wang Y, Ye D, Liu Q (2020) Review of the 2019 novel coronavirus (SARS-CoV-2) based on current evidence. *Int J Antimicrob Agents* 55(6):105948. <https://doi.org/10.1016/j.ijantimicag.2020.105948>
8. Chinazzi M et al (Apr 2020) The effect of travel restrictions on the spread of the 2019 novel coronavirus (COVID-19) outbreak. *Science* (1979) 368(6489):395–400. <https://doi.org/10.1126/science.aba9757>
9. Sartorius B, Lawson AB, Pullan RL (2021) Modelling and predicting the spatio-temporal spread of COVID-19, associated deaths and impact of key risk factors in England. *Sci Rep* 11(1):5378. <https://doi.org/10.1038/s41598-021-83780-2>
10. Samuel J et al (2020) Feeling positive about reopening? New normal scenarios from COVID-19 US reopen sentiment analytics. *IEEE Access* 8:142173–142190. <https://doi.org/10.1109/ACCESS.2020.3013933>
11. Potential rapid increase of omicron variant infections in the United States.
12. le Rutte EA, Shattock AJ, Chitnis N, Kelly SL, Penny MA (2022) Modelling the impact of Omicron and emerging variants on SARS-CoV-2 transmission and public health burden. *Commun Med* 2(1):93. <https://doi.org/10.1038/s43856-022-00154-z>
13. Barnard RC et al (2022) Modelling the medium-term dynamics of SARS-CoV-2 transmission in England in the Omicron era. *Nat Commun* 13(1):4879. <https://doi.org/10.1038/s41467-022-32404-y>
14. Callaway E What omicron's BA.4 and BA.5 variants mean for the pandemic
15. Khan K et al (2022) Omicron BA.4/BA.5 escape neutralizing immunity elicited by BA.1 infection. *Nat Commun* 13(1):4686. <https://doi.org/10.1038/s41467-022-32396-9>
16. Arias Velásquez RM, Mejía Lara JV (Jul 2020) Forecast and evaluation of COVID-19 spreading in USA with reduced-space Gaussian process regression. *Chaos Solitons Fractals* 136:109924. <https://doi.org/10.1016/j.chaos.2020.109924>
17. Yousaf M, Zahir S, Riaz M, Hussain SM, Shah K (2020) Statistical analysis of forecasting COVID-19 for upcoming month in Pakistan. *Chaos Solitons Fractals* 138:109926. <https://doi.org/10.1016/j.chaos.2020.109926>
18. Saba AI, Elsheikh AH (2020) Forecasting the prevalence of COVID-19 outbreak in Egypt using nonlinear autoregressive artificial neural networks. *Process Saf Environ Prot* 141:1–8. <https://doi.org/10.1016/j.psep.2020.05.029>
19. Ren H et al (2020) Early forecasting of the potential risk zones of COVID-19 in China's megacities. *Sci Total Environ* 729:138995. <https://doi.org/10.1016/j.scitotenv.2020.138995>
20. Chin V et al (2020) A case study in model failure? COVID-19 daily deaths and ICU bed utilisation predictions in New York state. *Eur J Epidemiol* 35(8):733–742. <https://doi.org/10.1007/s10654-020-00669-6>

21. Anderson RM, Heesterbeek H, Klinkenberg D, Hollingsworth TD (2020) How will country-based mitigation measures influence the course of the COVID-19 epidemic? *Lancet* 395(10228):931–934. [https://doi.org/10.1016/S0140-6736\(20\)30567-5](https://doi.org/10.1016/S0140-6736(20)30567-5)
22. Li Z, Zheng Y, Xin J, Zhou G (2020) A recurrent neural network and differential equation based spatiotemporal infectious disease model with application to COVID-19. In: *Proceedings of the 12th international joint conference on knowledge discovery, knowledge engineering and knowledge management*, pp 93–103. <https://doi.org/10.5220/0010130000930103>
23. Zheng Y, Li Z, Xin J, Zhou G (2021) A spatial-temporal graph based hybrid infectious disease model with application to COVID-19. In: *Proceedings of the 10th international conference on pattern recognition applications and methods*, pp 357–364. <https://doi.org/10.5220/0010349003570364>
24. Sun J et al (2020) Forecasting the long-term trend of COVID-19 epidemic using a dynamic model. *Sci Rep* 10(1):21122. <https://doi.org/10.1038/s41598-020-78084-w>
25. Italian region. Italian COVID-19 data. <https://github.com/lzj994/COVID-19-Data/blob/master/covid-19/italy.csv>. Accessed March–July 2020

Russo-Ukrainian War on Facebook: A Temporal and Spatial Study Based on BERTopic Modeling



Kai-Yi Tan , Chun-Ming Lai , Yin-Jie Foo , and Yuya Shibuya

Abstract In today's interconnected world, the number of social media users, such as Twitter, Facebook, and Reddit, has continued to grow. The vast number of users generates a wealth of social data, which holds significant value regarding semantic sentiment, public opinion, and societal interests. This study delves into topics and sentiments of Facebook page contents related to the Russo-Ukrainian War. In specific, the study employs the unsupervised text clustering method, BERTopic, to identify the predominant themes in these Facebook posts. Additionally, we leverage VADER (Valence Aware Dictionary and sentiment Reasoner) to uncover the semantic emotions conveyed in the posts. The study revealed the main topics discussed on Facebook during the war and the changes in sentiment distribution across different topics over time.

Keywords Topic modeling · BERTopic · Data mining · Information war

1 Introduction

The Russia-Ukraine war escalated on February 24, 2022, with Russia invading Ukraine and Putin describing the war as a “special military operation”. In addition, the Russia-Ukraine war triggered global financial and economic turmoil, triggered a humanitarian crisis, affected the global economy, and resulted in a rapid increase in news reports and public opinion discussions. Due to the existence of social media and the high degree of free communication it provides, there is an abundant amount of opinions and information spread around it. As a result, social media has become

K.-Y. Tan · C.-M. Lai (✉) · Y.-J. Foo

Department of Computer Science, Tunghai University, Taichung City, Taiwan

e-mail: cmlai@thu.edu.tw

Y. Shibuya

The University of Tokyo, Bunkyo, Japan

e-mail: yuya-shibuya@csis.u-tokyo.ac.jp

a primary source for people to extract new information or a platform to release their beliefs or sentiments.

In today's community, few people do not have any accounts on the popular social media platform, such as Twitter, Facebook, Reddit, etc. The massive amount of users and the social data added daily make the data labeling for further academic research difficult or even impossible. On the other hand, the interaction of various opinions on social media is quite complex and diverse, even with mistakes or misinformation. Nonetheless, it is undeniable that the underlying value behind Facebook posts, i.e., the semantic sentiment, public opinion, and interest of the public, is enormous and cannot be ignored.

To address the above challenges, this study illustrates an analysis of the social data on the Facebook platform which related to the Russo-Ukrainian War through topic modeling and sentiment analysis approaches. An unsupervised text clustering method, BERTopic, is employed to explore the primary subjects among these Facebook posts. VADER (Valence Aware Dictionary and sEntiment Reasoner) is applied to mine the semantic emotion of the posts. Furthermore, this study answers the following research questions from various perspectives.

RQ1: What are the main topics underlying Facebook posts during the war?

RQ2: How has the sentiment distribution on different topics changed over time?

The primary contributions of this study are as follows.

1. This study demonstrates the capability of BERTopic to cluster enormous datasets depending on the field. However, each cluster does not have a precise label, but the keywords generated are enough to interpret its content.
2. This study employs a temporal analysis regarding the tendency of sentiment variation to analyze the sentiment of Facebook users.

The remainder of this study is organized as follows. Section 2 discusses relevant studies. Section 3 introduces the dataset used in this study together with the necessary preprocessing step, descriptive statistical information of the dataset, and the methodological approach applied in this study. Section 4 presents the results obtained and attempts to answer the research questions. Section 5 concludes this study and describes the challenges and limitations.

2 Related Work

Social media platforms are irreplaceable elements nowadays as they enable people from worldwide to discuss, interact, and spread thoughts with each other. As social media continues to grow persistently, the data has become enormously large and diverse. To simplify the analysis and study of social media data, the sentiments of users are frequently analyzed to extract emotions from vast amounts of data as they likely reflect the population's behaviors and preferences [1]. Topic modeling is also a popular method to simplify large amounts of data into key topics. Social media is another popular research area in computational social science [2]. Researchers

applied sentiment analysis and topic modeling in diverse domains to discover the reflection of peoples' thoughts and behaviors on the internet [3].

Topic modeling is an unsupervised, popular technique for text data evaluation, and demonstrating discrete data, aiming to identify and extract hidden meaningful semantic topics from gigantic datasets [4]. There are several algorithms of topic modeling techniques, LDA (Latent Dirichlet allocation) is one of the most popular and frequently used algorithms for Topic Modeling. LDA stands as a generative probabilistic model for analyzing a corpus. Its fundamental concept lies in representing documents as stochastic blends of hidden topics, each topic being defined by a word distribution [5]. BERTopic, the state-of-the-art topic modeling method, offers a coherent representation of topics generated by applying clustering techniques and incorporating class-based variations of TF-IDF (c-TF-IDF). BERTopic is dynamic, allowing documents to be organized sequentially for the modeling, which distinguishes it from traditional topic modeling techniques. The temporal dependencies can be found and analyzed as c-TF-IDF for each cluster at each timestep without the need for embedding and clustering, which results in faster computation [6]. The study of [7] shows that the potency of BERTopic and NMF is higher than LDA and Top2Vec. While both BERTopic and NMF hold similar positions in the ranking of topic modeling methods, a notable distinction is that NMF shares a common drawback with LDA in that it necessitates users to predefine the number of topics before conducting the topic modeling.

To study discussions related to the Russo-Ukrainian war on various social media platforms, researchers recently created datasets from platforms such as Twitter, Reddit, and Weibo. Because Reddit consists of sub-communities covering a focal topic, through exhaustive subreddit exploration, [8] provided a Reddit dataset of 12 prominent subreddits, starting from 24 February 2022 to 29 May 2022, related to the ongoing Russo-Ukrainian war, with 300,000 posts and 8M comments. Furthermore, categorizing these posts based on their primary focus into two categories, "R-U Conflict" and "Military Related". The authors showed an obvious Pro-Ukraine stance in "R-U Conflict" and a neutral stance in "Military Related" by textual analysis. Reference [9] presented an open dataset from Weibo posts relevant to the ongoing 2022 Russo-Ukrainian crisis, called Russia-Ukraine Crisis Weibo (RUW). RUW is continuously updated to facilitate further research in time analysis, with over 27,341 Weibo posts and over 3.5M corresponding Weibo comments in the first version. The dataset contains more than 1.6M tweets [10] shared during the first week of the crisis related to the Russia-Ukraine war that was introduced in the first version.

Reference [11] explored the sentiments and emotions of Twitter users regarding green energy during the 2022 Ukraine-Russia conflict, which plays a crucial role in global energy policy. Through comparative analysis and factor analysis, the authors observed two latent variables before conflict: one optimism and the other pessimism. However, after the conflict began, the optimism was subdivided into optimism and expectation. Reference [12] analyzed public sentiment against the economic sanctions discussed on Facebook during the Russia-Ukraine war from February 1, 2022, to March 31, 2022. The authors demonstrated that public sentiment in the country can be explained by the intensity of Russia's trade with it.

Reference [13] proposed a comprehensive approach to study the evolutionary opinion dynamics in Weibo, associated with RUW, regarding the Russia-Ukraine conflict and release their dataset. First, the authors use Roberta's pre-trained model to identify the emotions of posts, then use an unsupervised clustering LDA model to cluster the social data of the two emotions into four categories and generate opinion content through keyword extraction, and lastly, propose an opinion adversarial evolution algorithm to identify the interactions and dominance degree of each opinion. Reference [14] revealed the sentiment of around 80000 demographically diverse tweets representing Twitter users of 7 countries and the focal point of people across the globe during the Russia-Ukraine conflict. The author uses sentiment analysis to explore the dominance of negative feelings toward these conflicts and applies topic modeling to show the massive amount of topics in various groups. Reference [15] primarily focused on collecting data from Ukrainian Telegram users during the first 2 months of the war in Ukraine. The authors apply topic modeling to extract the focal point and sentiment analysis to analyze the emotions of users per day as well as for the whole period.

The aforementioned studies attempt to understand the concerns of people through behavior on social media or reveal the tendency of sentiment variation. Since the impact of the Russo-Ukrainian war is global and misinformation is sometimes spread on social media, worldwide awareness of the war should be emphasized. Therefore, the scope of this study is to be adopted in integrated with relevant studies in this field (i.e., topic modeling and sentiment analysis), considering their demonstrated applicability and effectiveness in the related studies discussed earlier. In addition, this study will mainly focus on time and geographical location information to illustrate the tendency of sentiment variation across diverse countries.

3 Methodology

3.1 Data Collecting and Preprocessing

The dataset used in this study was collected by CrowdTangle API [16], a public insight tool proposed by Meta to retrieve public content on various social media such as Facebook, Instagram, and Reddit. To ensure the relevance of posts to the Russia-Ukraine war, the search keyword is set as "Ukraine" and "Russia". The dataset includes 299,993 Facebook page posts from 170 countries spread throughout 2022.

The dissemination of information on social media comprises various mediums, including articles, videos, images, and more. Since the CrowdTangle API categorizes Facebook posts as "Link", "Live Video", "Full Live Video", "Scheduled Live Video", "Native Video", "Photo", "Status", "Video", and "YouTube", Facebook posts type which is "Link" and a "Status", which consist mainly of textual information, are retained. While each post contained 41 attributes, only "Page Admin Top Countries", "Post Creation Date", "Message" and "Total Interactions" are relevant for the subsequent experiments. To investigate spatial information at the national level,

Table 1 Descriptive statistics of the dataset with grouping by diverse country

Page Admin Top Country	#Posts	Word length		
		Median	Mean	Std.
US	74983	30	51.90	96.73
CA	10008	34	87.93	145.73
GB	9741	29	68.50	126.24
UA	7831	37	59.52	76.56
IN	5614	27	34.81	50.37
PH	4563	32	40.70	52.30
AU	4396	28	49.54	86.85
TR	2945	29	30.00	10.75
IE	2264	30	38.12	54.00
NG	1691	36	62.18	106.74
ZA	1675	34	50.74	83.79
AE	1524	28	29.25	14.64
SG	1504	24	54.89	124.38
IL	1266	32	40.82	45.89
BE	1228	39	50.15	61.59
CZ	1002	29	34.67	32.22
NZ	862	29	54.30	104.39
JP	769	30	40.41	57.48
KE	581	47	55.02	62.39
KR	163	34	44.89	72.40
Total	132510	31	53.85	97.85

“Page Admin Top Countries” is utilized as the geographical indicator for posts. Among these, the 20 with the highest frequency of posts accounted for 174,865 out of 194,150 posts, a 90% coverage rate. However, due to insufficient sample size, countries beyond this top 20 were excluded from this study. The third preprocessing step removes non-English text from the posts. To ensure the adequacy of textual information within each post, any posts with a length below 18 words (below the first quartile of word length) will be removed.

Table 1 provides descriptive statistics of the dataset. The US contains the highest number of posts in the dataset, around 57%. The country with the fewest posts is KR, with only 163 posts. Since Facebook posts are limited to a maximum length of 63,206 characters, the length of words in posts varies widely.

3.2 Topic Modeling Analysis

Since users in social media are interested in various types of Russo-Ukrainian War events, topic modeling with BERTopic is employed to discover latent topics present in different sentiment groups of posts. BERTopic can be divided into a sequence of

steps, lending a degree of flexibility in modular for analysts to integrate or apply supplementary techniques in each step. The first step in BERTopic is document embedding, and Sentence-BERT (SBERT) is selected due to it achieves state-of-the-art performance in Semantic Textual Similarity (STS) across a wide range of applications. Additionally, SBERT exhibits notable computational efficiency in clustering compared to BERT/roBERTa frameworks [Sentence-bert: Sentence embeddings using siamese bert-networks]. Since the post is in English, “all-MiniLM-L6-v2” is more applicable due to being exclusively in English and is more streamlined than “paraphrase-multilingual-MiniLM-L12-v2”. After that, BERTopic performs dimensionality reduction and clustering, and leverages class-based TF-IDF procedure to create topic representation. The provided steps are executed with default parameters.

Initially, we performed a preliminary analysis of the most influential parameter “min_topic_size”, which significantly determines the minimum size of each topic. Figure 1 illustrates the performance of BERTopic on various “min_topic_size” against different sentiment groups of posts as measured by Topic Coherence (TC) and Topic Diversity (TD). Due to the inclusion of randomness in BERTopic, the results are based on experiments run with ten iterations. To avoid contextual overlap across topics, the higher the TD score, the greater the diversity between questions; therefore, the “min_topic_size” was set by 55.

Table 2 indicates that the results of BERTopic training against different sentiment groups of posts, 73,194 posts remained after removing the outlier topic. BERTopic identified 84 topics. Due to the difficulty of interpreting all the topics, we focus on the issues with the top 10 in quantity.

3.3 *Sentiment Analysis*

To investigate user sentiments across the posts of diverse countries against Russo-Ukrainian War events, VADER is employed to perform sentiment analysis. VADER is a rule-based sentiment analysis tool that relies on a lexical dataset that maps lexical features (e.g., words) to emotion intensities known as sentiment scores. A lexical dataset is a list of lexical features which are generally labeled according to their semantic orientation as either positive or negative. In other words, VADER provides the sentiment scores based on the words used in posts. Since VADER has been specially adapted to determine the sentiments expressed in social media texts, it is well-suited for our study.

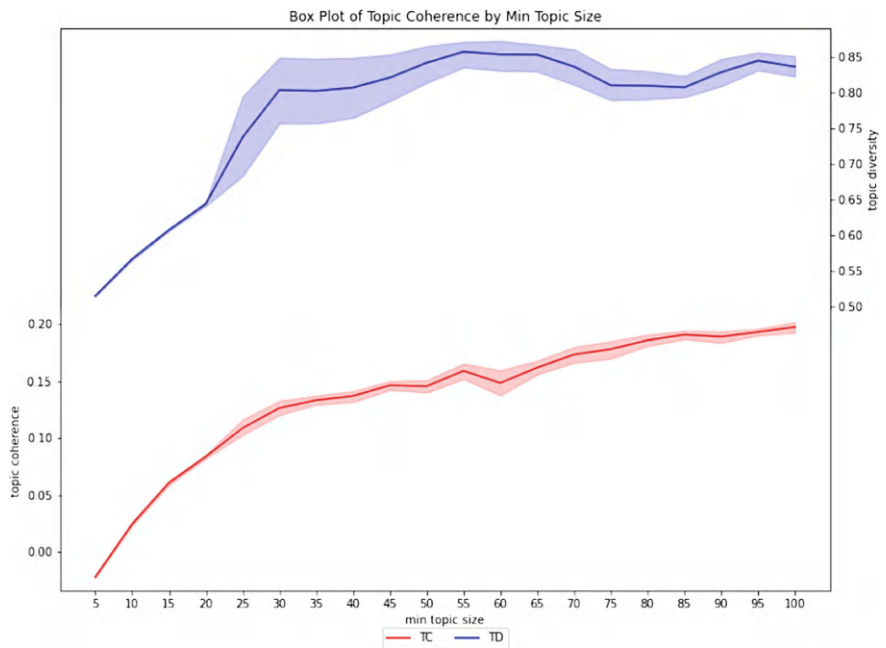


Fig. 1 Topic modeling metrics against to min_topic_size

4 Results

4.1 RQ1: What Are the Main Topics Underlying Facebook Posts?

Table 2 reveals the 10 topics selected, which are discussed most frequently about Russia-Ukraine War on Facebook. Each topic is clustered through BERTopic. The topics of BERTopic are characterized by topic label, followed by the 10 most symbolic terms (unigram/bigram) according to the c-TF-IDF metric, and the number of posts in different sentiments (Negative, Neutral, Positive).

The resulting outcome from BERTopic generally involved the topics related to the events of the Russian Invasion of Ukraine, such as the reaction of world leaders, support from the world citizens, the impact on the economy caused by the war, and the role of the internet in modern warfare.

Topic 0 is an introductory overview of the Russian Invasion of Ukraine discussed and shared on Facebook. It includes general information related to the conflict, such as news articles, posts, comments, and discussions among Facebook users.

Topic 1 discussed the actions and standpoint of Ukraine’s President Volodymyr Zelensky, which are the protagonist of the war. He strongly criticized Russia and Russian President Putin for invading Ukraine. Besides, he is the spiritual leader of

Table 2 Topic representation for the Facebook posts

Topic label	Keywords	#Posts (neg)	#Posts (neu)	#Posts (pos)
Topic 0: Russia-Ukraine War in General	Russian, Ukraine, Said, Russia, invasion, nuclear, war, President, Ukrainian, Putin	31,248	5,374	11,877
Topic 1: The Actions and Standpoint of Ukraine President Volodymyr Zelensky	Volodymyr, Zelensky, Zelenskyy, president, Ukrainian, Zelenskiy, said, address, Ukraines, country	1,957	475	1,139
Topic 2: Prayer from the World Pray and Urge	Church, pray, orthodox, god, prayer, peace, patriarch, catholic, prayers, kirill	656	147	835
Topic 3: Impact to the World due to the Damage of Supply Chain	Grain, food, wheat, ports, farmers, exports, sea, global, black, prices	815	150	282
Topic 4: The Standpoint of China on the War	China, Chinese, xi, Taiwan, Beijing, chinas, Jinping, Russia, Putin, president	628	132	412
Topic 5: The Talks and Negotiations between Russia and Ukraine	Talks, negotiations, ceasefire, peace, delegation, ready, kremlin, negotiate, round, negotiators	428	132	429
Topic 6: Cyber	Cyber, cyberattacks, hackers, cybersecurity, websites, internet, government, anonymous, cyberattack, attacks	616	96	250

(continued)

Table 2 (continued)

Topic label	Keywords	#Posts (neg)	#Posts (neu)	#Posts (pos)
Topic 7: The Actions and Standpoint of Canada Prime Minister Justin Trudeau	Canada, trudeau, justin, canadian, prime, minister, canadas, canadians, sanctions, support	456	144	326
Topic 8: The Response and Standpoint of Donald Trump	Trump, Trumps, Donald, Putin, republicans, republican, election, president, genius, manafort	367	69	363
Topic 9: The Response of Israel to Russian Invasion of Ukraine	Israel, Jewish, Israeli, jews, Israels, bennett, holocaust, Jerusalem, Naftali, minister	436	68	211
	Total	37,607	6,787	16,124

the Ukraine people. In Topic 5, the talks and negotiations between Russia and Ukraine are discussed, revealing that both sides consider war may be the last resort.

In Topic 2, the top keywords “church”, “pray”, and “orthodox” reveal that religion holds significance and plays a crucial role for individuals in times of despair. On the contrary, it also reveals how religion can be used to manipulate and justify the actions of people, leading to miserable outcomes. (“The head of the Russian Orthodox Church has urged Russian citizens not to be afraid of death amid Vladimir Putin’s decision to mobilize troops in Ukraine”) is posted by Media News Benzinga on Facebook.

Ukraine is one of the major exporters of agriculture [17], while Russia ranks as the second-largest exporter of oil and gas [18]. The aftermath of the war and its impact on these countries and the rest of the world is discussed in Topic 3. It also included the sanctions taken toward Russia.

Topic 6 reveals the significance of cyber warfare in the modern era, encompassing cyber defences, hackers’ activities targeting infrastructure, and the dissemination of fake news.

Concerns and actions of national leaders can be found in Topics 4, 7, 8, and 9. The information underlying these topics is as follows. For China in Topic 4, the war between Russia and Ukraine is seen as analogous to the conflict between China and Taiwan by outsiders, and China’s stance in this war is ambiguous. Apart from this, in Topic 7, Canada has a strict standpoint which condemns Russia for invading Ukraine. The rumor about former US President Donald Trump and his standpoint on

this event is revealed on Topic 8. In Topic 9, the actions taken by Israel, including the evacuation of its citizens from Ukraine and Russia are discussed.

In conclusion, these topics included various critical aspects of the Russian Invasion of Ukraine, ranging from political dynamics, and economics to the role of the internet and cybersecurity. The topics shed light on the introductory overview of the conflict between Russia and Ukraine, and the actions and perspectives taken by key individuals such as Ukraine President Volodymyr Zelensky and the head of the Catholic Church Pope Francis. It also revealed the economic consequences throughout the globe, including the sanctions toward Russia. In addition, the involvement of cyber warfare can be found to be increasing rapidly. To sum up, these topics offer a diverse and comprehensive understanding of this conflict.

4.2 RQ2: How Has the Sentiment Distribution on Different Topics Changed over Time?

Figure 2 illustrates the distribution of sentiment on different topics over each month. The rate of negative sentiment rapidly increased in April, with one huge event on 8 April, the war entering its next phase. The sentiment of posts between May and November remained predominantly negative, indicating that conflict incidents are still ongoing. In December, the tide of conflict eased, with the United States making a promise and other countries beginning to provide direct supplies.

Furthermore, since the number of Topic 0 overwhelms the other topics, Fig. 2 cannot represent the sentiment trend behind Topic 1 to 9. Returning briefly to the topics explained in the previous subsection, we are interested in how the sentiment distribution on Topic 2 and Topic 3 (represented the most positive and the most negative topics, respectively) changed over time.

The variation in sentiment distribution over each month on Topic 2 and Topic 3 is illustrated in Fig. 3. When the war launched in February, Pope Francis expressed his empathy for the unfortunate Ukrainian people with the words: “The Holy Father is present in Ukraine. He is suffering with you”. Although most religious activities are positive, negative posts outnumbered positive posts in March due to the Russian Orthodox supporting Vladimir Putin’s war against Ukraine. The percentage of negative posts reached one of the peaks in June when Russia justified the invasion as an act of holy war and caused panic among humanity. The rapid decline in the proportion of negative posts between July and August is possibly in reaction to the rapid decrease in the number of negative posts, but positive posts urging prayers for peace still exist. In September, the negative rate in posts reached another peak due to Patriarch Kirill saying: “Russians who die in Ukraine have their sins washed away”.

Moreover, the distribution of posts over each month on Topic 3 is opposite from overall topics and Topic 2. After the war began, there were diverse actions taken against Russia and Ukraine adopted by various countries. For instance, sanctions

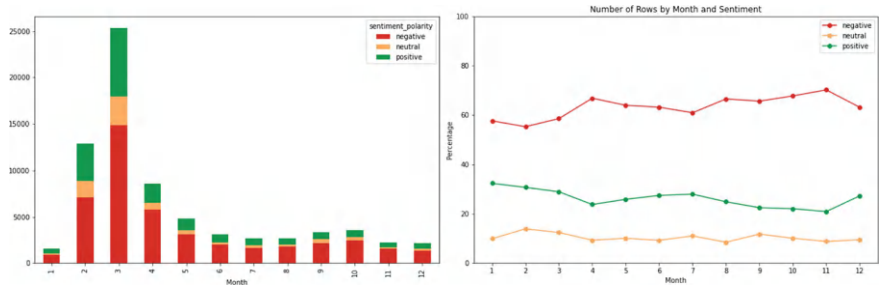


Fig. 2 The sentiment distribution on different overall topics changed over time

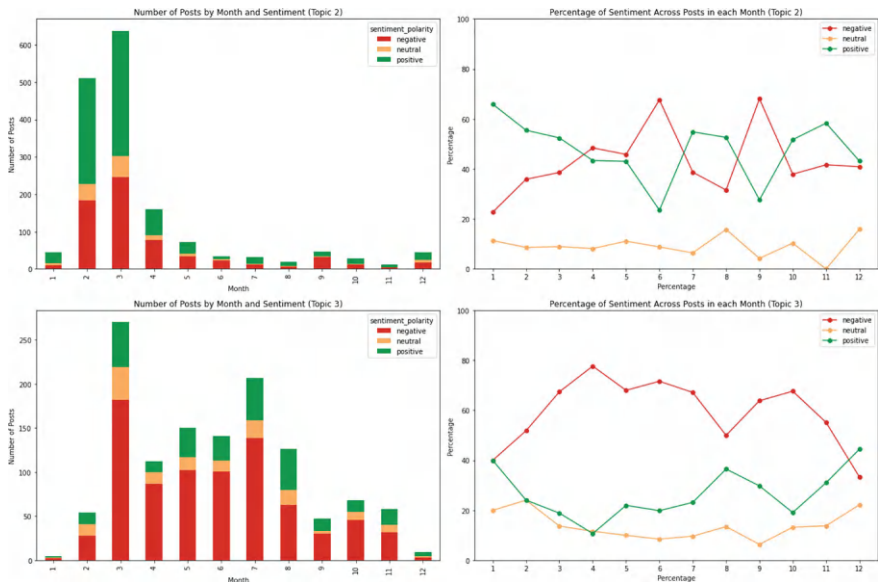


Fig. 3 The sentiment distribution on Topic 2 and Topic 3 changed over time

were imposed on Russia, including the ban on importing Russian oil and gas by European countries, international companies leaving Russia, etc.

One of the consequences of the sanction is the raise of petroleum price leading to peoples' concerns. Additionally, Ukraine's grains export also affected by the war, leading to social issues, "The Russian war in Ukraine has already pushed the world to the brink of a global humanitarian and food crisis" posted by page "Union of Concerned Scientists". Therefore, these actions come up with an abundant discussion about the impacts from different aspects so that the number of posts is not reduced immediately. The impact of these issues is seriously discussed as Russia and Ukraine play a significant role among exporting countries; therefore, the negative percentage remains persistently high.

5 Conclusion

In this study, we conducted an analysis of Facebook posts related to the Russo-Ukrainian War, employing topic modeling and sentiment analysis techniques. To identify and categorize the primary subjects of discussion on Facebook, we utilized BERTopic as our topic modeling method. While the generated Topic may not come with a precise label, its content can still be roughly interpreted. For instance, in Topic 1, keywords like “Volodymyr”, “Zelensky”, “president”, and “Ukrainian” act as dots. By examining the messages within this Topic, it becomes relatively easy to connect the dots and assign an appropriate topic for further study. Given the presence of various sentiments within a single topic, we utilize VADER for sentiment analysis on each topic. By incorporating the element of time into our sentiment analysis, our study provides a temporal perspective for each topic. This enables us to uncover patterns in sentiment fluctuations and delve into the emotional responses of Facebook users. For instance, in Topic 2, we observe a shift in sentiment trends from positive to negative during the first six months. This topic delves into prayers and the church’s influence. Upon closer examination of Facebook posts, we uncover that the increase in negative sentiment is attributed to Russian Orthodox Patriarch Kirill justifying and rationalizing the war. This research provides a more oriented view of the discussion and enthusiasm of various national communities on the two sides of the issue at the time of the war, and it can be seen that there are obvious emotional and regional differences.

References

1. Wankhade M, Rao ACS, Kulkarni C (2022) A survey on sentiment analysis methods, applications, and challenges. *Artif Intell Rev* 55(7):5731–5780
2. Guan W, Gao H, Yang M, Li Y, Ma H, Qian W, Cao Z, Yang X (2014) Analyzing user behavior of the micro-blogging website sina weibo during hot social events. *Phys A: Stat Mech Its Appl* 395:340–351
3. Zhang X, Lyu H, Luo J (2021) Understanding the hoarding behaviors during the covid-19 pandemic using large scale social media data. In: 2021 IEEE international conference on big data (big data). IEEE, pp 5007–5013
4. Vayansky I, Kumar SA (2020) A review of topic modeling methods. *Inf Syst* 94:101582
5. Jelodar H, Wang Y, Yuan C, Feng X, Jiang X, Li Y, Zhao L (2019) Latent dirichlet allocation (lda) and topic modeling: models, applications, a survey. *Multimedia Tools Appl* 78:15169–15211
6. Grootendorst M (2022) Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint [arXiv:2203.05794](https://arxiv.org/abs/2203.05794)*
7. Egger R, Yu J (2022) A topic modeling comparison between lda, nmf, top2vec, and bertopic to demystify twitter posts. *Front Sociol* 7:886498
8. Zhu Y, Haq EU, Lee LH, Tyson G, Hui P (2022) A reddit dataset for the russo-ukrainian conflict in 2022. *arXiv preprint [arXiv:2206.05107](https://arxiv.org/abs/2206.05107)*
9. Fung YR, Ji H (2022) A weibo dataset for the 2022 russo-ukrainian crisis. *arXiv preprint [arXiv:2203.05967](https://arxiv.org/abs/2203.05967)*
10. Haq EU, Tyson G, Lee LH, Braud T, Hui P (2022) Twitter dataset for 2022 russo-ukrainian crisis. *arXiv preprint [arXiv:2203.02955](https://arxiv.org/abs/2203.02955)* (2022)

11. Ibar-Alonso R, Quiroga-García R, Arenas-Parra M (2022) Opinion mining of green energy sentiment: A russia-ukraine conflict analysis. *Mathematics* 10(14):2532
12. Ngo VM, Huynh TL, Nguyen PV, Nguyen HH (2022) Public sentiment towards economic sanctions in the russia-ukraine war. *Scottish J Polit Econ* 69(5):564–573
13. Chen B, Wang X, Zhang W, Chen T, Sun C, Wang Z, Wang FY (2022) Public opinion dynamics in cyberspace on russia-ukraine war: a case analysis with chinese weibo. *IEEE Trans Comput Soc Syst* 9(3):948–958
14. Sazed S (2022) The dynamics of ukraine-russian conflict through the lens of demographically diverse twitter data. In: 2022 IEEE international conference on big data (big data). IEEE, pp 6018–6024
15. Maathuis C, Kerkhof I (2023) The first two months in the war in ukraine through topic modeling and sentiment analysis. *Region Sci Policy Pract* 15(1):56–74
16. Team C (2023) Crowdtangle. Facebook, Menlo Park, California, United States
17. Rose A, Chen Z, Wei D (2023) The economic impacts of russia–ukraine war export disruptions of grain commodities. *Appl Econ Perspect Policy*
18. List of countries by oil exports. https://en.wikipedia.org/wiki/List_of_countries_by_oil_exports, (Accessed on 16 Aug 2023)

A Performance Study of Selected Machine Learning Techniques for Predicting Heart Diseases



**Blessing Oluwatobi Olorunfemi, Adewale Opeoluwa Ogunde,
Abidemi Emmanuel Adeniyi, Joseph Bamidele Awotunde,
Agbotiname Lucky Imoize, and Chun-Ta Li**

Abstract Heart Disease remains a leading cause of mortality worldwide. It alarmingly rises at a quick rate, making early heart disease prediction crucial for effective prevention and timely intervention. Heart disease diagnosis is a difficult process that requires technical skills and accuracy to complete. With improvements in technology, computing has lent its voice to simplify the diagnosis of various health problems. Machine learning uses past or existing history to predict future results. Various machine learning techniques have been developed over the years and used in predicting heart diseases with various levels of performance. Identifying the best-suited machine learning technique to use for prediction purposes can be a challenging

B. O. Olorunfemi · A. O. Ogunde

Faculty of Natural Sciences, Department of Computer Science, Redeemer's University, Osun, Nigeria

e-mail: olorunfemi@run.edu.ng

A. O. Ogunde

e-mail: ogundea@run.edu.ng

A. E. Adeniyi

Department of Computer Science, Bowen University, Iwo, Nigeria

e-mail: abidemi.adeniyi@bowen.edu.ng

J. B. Awotunde

Faculty of Information and Communication Sciences, Department of Computer Science, University of Ilorin, Ilorin 240003, Nigeria

e-mail: awotunde.jb@unilorin.edu.ng

A. L. Imoize

Faculty of Engineering, Department of Electrical and Electronics Engineering, University of Lagos, Akoka, Lagos 100213, Nigeria

e-mail: aimoize@unilag.edu.ng

C.-T. Li (✉)

Bachelor's Program of Artificial Intelligence and Information Security, Fu Jen Catholic University, New Taipei City 24206, Taiwan

e-mail: 157278@mail.fju.edu.tw

task. This research work analyses the performance of seven (7) machine learning techniques, comprising AdaBoost Algorithm, KNN, Logistic Regression, Naïve Bayes Classifier, Random Forest, SVM, and XGBoost. The heart disease dataset was downloaded from the UCI repository and analysed using Python programming language in the Jupyter Notebook environment. A comparative analysis of the seven (7) techniques was performed based on Accuracy, Precision, and Recall. From the results obtained, KNN, Random Forest, and XGBoost showed superior performance over the others with an accuracy of 100%, AdaBoost Algorithm followed with an accuracy of 92.2%, SVM followed with an accuracy of 91.71%, Naïve Bayes Classifier followed with an accuracy of 88.29% while Logistic Regression has the least accuracy of 86.34%. KNN, RF, and XGBoost outperformed AdaBoost, SVN, and LR.

Keywords Machine learning · Predictive model · Classification algorithm · Heart disease · Performance evaluation

1 Introduction

Heart disease can occur in someone who does not feel exhausted. There are signs in a few populations with cardiac disease. This is when the body experiences changes or pain that indicate a disease is present. Several signs of cardiac disease include: a sternal ache, difficulty breathing, palpitations (an inclination that the heart is pulsating excessively quickly), swelling in the legs or feet, and feeling tired because the body and brain aren't getting sufficient oxygen to oxygenate them. The author [1] defines cytositis as "blue skin." Age, sex, cigarette smoking, actual idleness, drinking too much alcohol, overweight or obese weight, hereditary vulnerability and family history of heart disease, raised pulse (hypertension), raised glucose levels (diabetes mellitus), elevated blood cholesterol (hyperlipidemia), undiagnosed celiac infection, psychosocial factors, and genetic susceptibility variables, and air pollution are just a few of the risk factors for heart disease [2, 3]. A number of these risk variables, such as age, gender, and relatives' ancestry/hereditary pattern, are unchangeable; nevertheless, many key cardiovascular risk variables can be changed through changes in lifestyles, changes in society, and medication (for example, hypertension, hyperlipidemia, and diabetes prevention) [4–6]. Obese people are more prone to develop atherosclerosis (i.e. fatty tissue, lipids, and other chemicals accumulate in and on the arterial walls).

Ageing is an indicator component of getting cardiovascular or heart disorders, with the risk typically growing dramatically with each passing decade of life. According to estimates, 82% of people who pass away from heart disease are 65 years of age or older [4, 7–9]. The risk of stroke also doubles every ten years beyond the age of 55. Many explanations are put out to explain why ageing increases the risk of heart and circulatory infections. One of them claims to have high serum cholesterol. The blood absolute cholesterol level rises with ageing in many populations. This increase stops in men between the ages of 45 and 50. The increase in women continues sharply until they are 60–65 years old. Finding a heart problem may be the most difficult

task in the medical services industry. To save lives, it must be quickly, effectively, and precisely examined. Numerous tests are required of the patient, and the results should be carefully examined by medical professionals. Because of this, scientists have been interested in predicting heart illness and have developed a variety of heart disease forecasting frameworks using various AI computations.

Others, of them achieved the desired results, while some used data obtained from the many clinics that were available to them, others prepared and tested their classifier using the well-known UCI heart disease dataset. Therefore, cardiovascular disease prevention has grown more crucial than ever [10, 11]. Excellent information-driven frameworks for predicting heart diseases can enhance the entire examination and preventative measure, ensuring that more people can continue living healthy lives. AI assists in predicting heart diseases, and the predictions are quite accurate. It involves looking into the dataset of patients with heart disease and gathering the necessary data [12, 13]. At that point, several models were created, and predictions were generated using KNN, SVM, XGboost, Naive Bayes, Random Forests, Logistic Regression, and Adaboost Algorithm [14].

This study makes a new addition by offering a complete and up-to-date assessment of state-of-the-art machine learning strategies for heart disease prediction, as well as a comparative performance assessment of seven distinct approaches to machine learning on a real-world heart disease dataset [15]. The study identifies the best-performing machine learning techniques for heart disease prediction, which can be used by healthcare professionals to develop more accurate and reliable heart disease prediction systems [16, 17].

The remaining of this study is divided into five sections. Section 1 describes the review of related studies in the area of heart disease and machine learning. Section 2 gives the detailed methodology used to achieve the aim of this study. Section 3 gives the results and discussion using tables and graphs. Section 4 concludes the study.

2 Materials and Methods

Cardiovascular disease is heightened as a silent killer that results in the untimely deaths of numerous persons without evident signs. The nature of the sickness is the source of growing concern about the illness and its effects. That is why it is highly germane to forecast the occurrence of this lethal illness.

2.1 Data Preprocessing

In addition to the calculations used, the nature of the dataset and its methods also have an impact on the presentation and accuracy of the predictive model. This preprocessing step is crucial because it prepares the dataset and organizes it so that the computation may use it. The size range that the dataset has is another crucial factor.

Some datasets include many attributes, which makes it more difficult to analyse, discover instances of, or anticipate accurately. By looking at the dataset and applying the proper information preparation techniques, these problems may be resolved.

Information cleansing, information modification, attribution of missing characteristics, information standardization, inclusion determination, and other advancements depending on the concept of the dataset are all examples of information preparation stages. Datasets may contain errors, omissions, redundancies, clamour, and a variety of other problems that make the material unsuitable for direct use by AI computation.

2.2 Model Architecture

The model architecture shown in Fig. 1 includes all the various components and their interactions to implement a predictive model for the detection of cardiovascular diseases (see Table 1).

The UCI Cleveland dataset, which is a publicly accessible dataset source, was utilized in this work as the heart disease data source. It was obtained from the UCI machine learning repository website. There are 303 instances in the data collection, and there are 14 attributes. The dataset is a collection of four alternative dataset combinations and it's used to diagnose heart disease; however, contrary to some sources, only the UCI Cleveland dataset was utilized in this work (<https://archive.uci.edu/ml/datasets/Cleveland+heart+disease+project>).

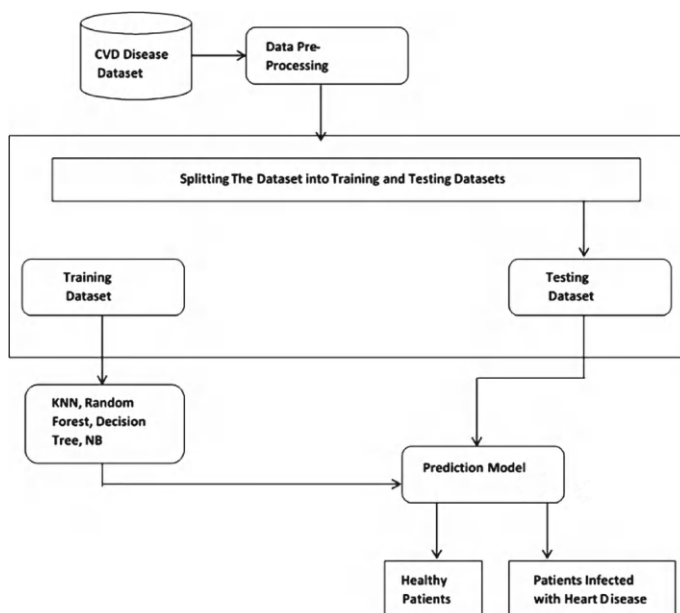


Fig. 1 The model architecture for the predictive model

Table 1 Tabular representation of the methodology used

Objective	Methodology	Output
Acquire dataset for cardiovascular disease	The cardiovascular disease dataset was acquired from the machine learning pool at UCI	Use the dataset acquired to build a predictive model
Develop a machine learning model to aid cardiovascular disease prediction	The model would be developed using Python: Data preprocessing is done to check for missing values in the dataset Feature extraction is also done to reduce the initial dataset to manageable groups for easy processing The classification method for the model was done using the selected classification algorithms	Identify the relevant features in the dataset, and develop a model using machine learning algorithms
Evaluate the performance of the developed model and build the proposed system	Evaluating the model based on its performance: accuracy, precision, recall, F1 score, and MCC	Results of the performance of the model

ics.uci.edu/ml/datasets/heart+Disease). The premise for the data is that there are 75 qualities or features, however, only 14 attributes have been processed in the published study [18, 19]. The UCI Cleveland dataset, which is part of the Dataset repository, contains the information, including the ‘Diagnosis’ attribute, which is used to predict outcomes or is known as the dependent variable’s name. The remaining attributes will be entered as input or are known as the independent variables. Table 2 lists the attribute descriptions.

2.3 Applying Various Machine Learning Algorithms

Logistic Regression: This algorithm teaches grouping calculations that are used to predict how similar two objective variables will be. Ward variables’ central concept is dichotomous, meaning that it can only be either 1 or 0. A logistic regression model predicts $P(Y = 1)$ as a part of X in terms of numbers. It is one of the least complicated ML algorithms that may be applied to a variety of grouping problems, such as spam recognition, diabetes prediction, illness localization, and so on.

KNN Algorithm: This method does a simple, controlled AI computation that may be used to address both the arrangement and relapse problems. Although it is simple to use and understand, it has the important drawback of gradually becoming less effective as the amount of data being used grows. It makes use of data from a few classes to predict how the new example point would be characterized.

Table 2 Dataset attribute

No.	Features/attributes	Type values	Description features	Value
1	Sex	Discrete variable	Male or female	1: Male 0: Female
2	Age	continuous variable	Patient's range shown by age	Numerous values between 28 and 77 (years of age)
3	CP (Chest Pain) Type	Discrete variable	Illustrate the following types of chest pain: classic angina, unusual angina, non-anginal pain, and asymptomatic	0: standard angina 1: unusual angina 2: Non-anginal discomfort 3: no symptoms
4	Rest Blood Pressure (Trestbps)	Continuous variable	Represents the resting heart rate (in mmHg on admission to the hospital)	Multiples continuous worth in mmHg
5	Serum Cholesterol (Chol)	Continuous variable	Serum cholesterol in Mg/Dl	Several Nonstop values in Mg/Dl
6	Fasting Blood Sugar (FBS)	Discrete variable	Represents the patient's fasting blood sugar level	0: false (FBS > 120 Mg/Dl) 1: true
7	Max Heart Rate (Thalach)	Continuous variable	Represents the patient's maximum heart rate	Manifold values from 71 to 202 Low: under 50 Beats/min Normal: 51–119 Beats/min High: 120–180 Beat/min
8	Res Electrocardiographic (Restecg)	Discrete variable	Describe the results of the ECG, where each number reflects the intensity of discomfort	0: Normal 1: having ST-T Wave abnormality (T wave inversions and/or ST elevation or depression of >0.05 mV) 2: showing probable or definite left ventricular hypertrophy by Estes' criteria
9	Exercise Induced (Exang)	Discrete variable	Calculated exercise-induced angina	1: yes 0: no

(continued)

Table 2 (continued)

No.	Features/attributes	Type values	Description features	Value
10	Oldpeak	Discrete variable	Show how exercise-induced ST depression compares to rest	Multiple decimal number values between 0 and 6.2
11	Slope	Discrete variable	Outline the patient's condition at the peak of activity. This section is divided into three components	0: upsloping 1: flat 2: downsloping
12	Major Vessels (Ca)	Discrete variable	The number of major vessels that fluoroscopy can illuminate. This section shows the total number of coloured vessels	Number of major vessels (0–3) coloured by fluoroscopy → (0, 1, 2, 3) value
13	Thal	Discrete variable	Patients experiencing chest discomfort or respiratory difficulty should also have this variable checked. This section displays three distinct value types displaying Thallium test results	0 = normal 1 = fixed defect 2 = reversible defect and the label
14	Target/condition	Discrete variable	The dataset's final column. This Target column is also known as the Class column or the Label column. This column provides prediction results with two classes, class 0 and class 1, based on the examination of the preceding 13 factors. If the class number is false, the chance of having heart disease is "0". If the class shows the number "1," the inverse is true, especially about the possibility of acquiring heart disease	0: no disease 1: Disease

SVM Backing Vector Machines (SVMs): are excellent but flexible directed AI computations that are used for both setup and relapse. But, they are used in characterization problems. When compared to other AI computations, SVMs have a unique technique of execution.

naïve Bayes: Calculations carried out by this method mostly rely on Bayes' Theorem. This is just one calculation, but other calculations are conducted, each of which follows a common norm, such as the independence of each pair of highlights in the arrangement. A probabilistic AI computation called Credulous Bayes may be applied to a variety of order projects. Common applications include filtering spam, aggregating reports, idea prediction, and others.

Random Forest: The technique of the classes (characterization) or mean/normal expectation, together with the construction of many choice trees during preparation time, are the key components of the random forest group learning approach for arranging, relapsing, and various tasks. The simplicity and versatility of this computation make it one of the most common ones.

XG Boost: It uses decision tree terminal nodes and computes shrinking leaf nodes as part of the Meta classifier in the machine learning environment. In that it reduces tree correlation, this strategy is comparable to gradient boosting. It has a greater range and is faster, though.

Ada Boost: In 1996, Yoav Freund and Robert Schapire created this strategy. The main goal of this technique for detailed data is to obtain maximum accuracy or to transform weak learners into strong learners.

3 Results and Discussion

The dataset used in this study was acquired from the UCI machine learning repository containing a total of 1025 patients and 14 attributes. The dataset was split into two, 80% for training and 20% for testing. The following are the attributes of the dataset as shown in Table 3 (age, sex, chest pain, blood pain, ECG findings, maximal heart rate, exercise-induced angina, ST depressive disorders, ST slope, Principal vessels, Thalassemia kinds and goal).

3.1 Performance Evaluation Matrix

A confusion matrix was used to evaluate the model. The number of accurate and inaccurate forecasts is totaled. Since we are working with binary classification, the matrix is 2×2 . The evaluation's usage of the confusion matrix is represented in Table 4. The two groups are 0 and 1, which indicate, respectively, negative (no heart disease) and positive (heart disease). While the non-diagonal numbers signify erroneous forecasts, the diagonal values reflect accurate predictions.

Table 3 A Sample of the dataset

Age	Sex	CP	Trestbps	Chol	FBS	Resteeg	Thalach	Exang	Oldpeak	Slope	Ca	Thal	Target
52	1	0	125	212	0	1	168	0	1.000000	2	2	3	0
53	1	0	140	203	1	0	155	1	3.100000	0	0	3	0
70	1	0	145	174	0	1	125	1	2.600000	0	0	3	0
61	1	0	148	203	0	1	161	0	0.000000	2	1	3	0
62	0	0	138	294	1	1	105	0	1.900000	1	3	2	0

Table 4 Confusion matrix

Actual		Predicted		Total P + N
		YES	NO	
	YES	TP	FN	
	NO	FP	TN	
	Total	P	N	

3.1.1 Terminologies of the Confusion Matrix are as Follows

1. True Positives [TP]: These are instances where the classifier correctly categorized a positive case.
2. True Negatives [TN]: These negative instances were correctly identified by the classifier.
3. False Positives [FP]: These are instances where a negative case was mistakenly labelled as a positive one.
4. False Negatives (FN): These are instances where a positive outcome was mistakenly categorized as negative.

The model's performance was assessed using the metrics listed below.

Accuracy: This is based on the confusion matrix; the rate of accuracy was computed using the formula below:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

Precision: These are referred to as positive predictive values. It is calculated by using this formula:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Recall: This is also referred to as sensitivity; it is calculated by using this formula:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

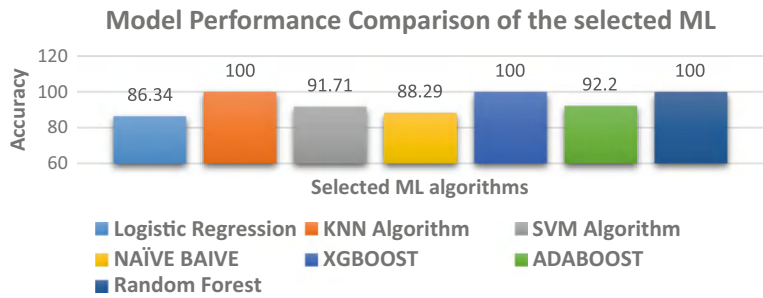


Fig. 2 Model performance comparison

Table 5 Results of accuracy

Algorithm	Accuracy (%)
Logistic regression	86.34
KNN algorithm	100.00
SVM algorithm	91.71
Naïve Baive	88.29
XGBoost	100.00
AdaBoost	92.20
Random forest	100.00

3.2 Discussion of the Model Performance Comparison

The accuracy of the selected models is displayed in Fig. 2. The figure shows that RF, KNN and XGBoost outperformed other selected models.

Most of the models work fine but among all the algorithms: KNN, XGBoost and Random Forest are the best with 100%. Thus, the accuracy rate for the following algorithms is explained below in Table 5.

The confusion matrix for the following machine learning algorithms is given below in Fig. 4a–f.

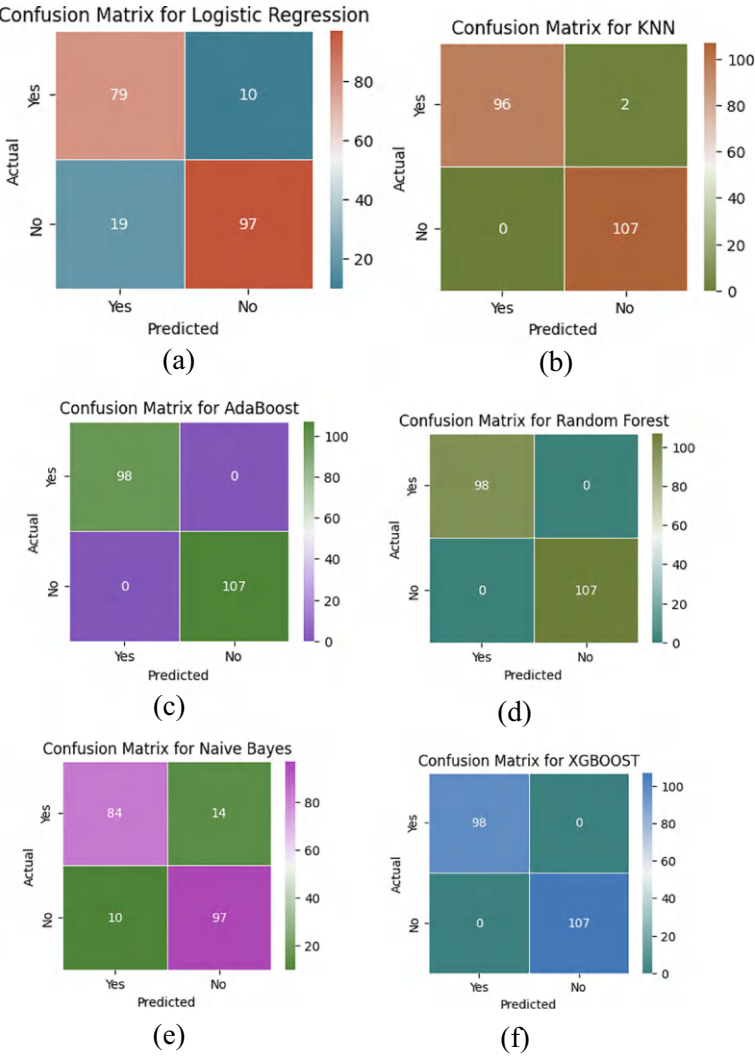


Fig. 4 a–f Confusion matrices for the selected algorithms

4 Conclusion

This task describes several machine learning grouping techniques for the cardiovascular heart disease predictive model. It has been sorted after reviewing several example articles on using AI techniques. The accuracy of the suggested models varies depending on the equipment used, the dataset used, the number of attributes captured in the dataset, etc. Understanding of the dataset was increased via information research. On the dataset, several machine learning computations were built

and tested. KNN, XGBoost, and random forest computations all had 100% accuracy, which was the best out of all the calculations. The study assumes that a dataset with sufficient instances and accurate information should be used to create an exact cardiac illness. Building a forecast model should also involve using an acceptable computation. Finally, using AI to classify cardiac disease is an important topic that may benefit both patients and medical professionals. Although patient information is monstrously accessible at medical clinics or centres, very little of it is transmitted because of rigorous ethical issues one has to deal with.

Acknowledgements This work was partially supported by the National Science and Technology Council, Taiwan, R.O.C., under contract no.: NSTC 110-2410-H-165-001-MY2.

References

1. Smith FC (2014) Chest pain, syncope, and palpitations in the pediatric patient. In: Functional symptoms in pediatric disease: a clinical guide, pp 27–45
2. Cosaro E, Bonafini S, Montagnana M, Danese E, Tretene MS, Minuz P, Fava C et al (2014) Effects of magnesium supplements on blood pressure, endothelial function and metabolic parameters in healthy young men with a family history of metabolic syndrome. *Nutr Metab Cardiovasc Dis* 24(11):1213–1220
3. Münzel T, Miller MR, Sørensen M, Lelieveld J, Daiber A, Rajagopalan S (2020) Reduction of environmental pollutants for prevention of cardiovascular disease: it's time to act. *Eur Heart J* 41(41):3989
4. Groenewegen A, Rutten FH, Mosterd A, Hoes AW (2020) Epidemiology of heart failure. *Eur J Heart Fail* 22(8):1342–1356
5. Biswas N, Ali MM, Rahaman MA, Islam M, Mia MR, Azam S, Moni MA et al (2023). Machine learning-based model to predict heart disease in early stage employing different feature selection techniques. *BioMed Res Int*
6. Folorunso SO, Awotunde JB, Adigun AA, Prasad LN, Lalitha VL (2023) A hybrid model for post-treatment mortality rate classification of patients with breast cancer. *Healthc Anal* 4:100254
7. Chandrasekhar N, Peddakrishna S (2023) Enhancing heart disease prediction accuracy through machine learning techniques and optimization. *Processes* 11(4):1210
8. Shukur BS, Mijwil MM (2023) Involving machine learning techniques in heart disease diagnosis: a performance analysis. *Int J Electr Comput Eng* 13(2):2177
9. Bakar WAWA, Josdi NLNB, Man MB, Zuhairi MAB (Mar 2023). A review: heart disease prediction in machine learning & deep learning. In: 2023 19th IEEE international colloquium on signal processing & its applications (CSPA). IEEE, pp 150–155
10. Awotunde JB, Ajagbe SA, Florez H (Oct 2023) A bio-inspired-based salp swarm algorithm enabled with deep learning for Alzheimer's classification. In: International conference on applied informatics. Springer Nature Switzerland, Cham, pp 157–170
11. Srivenkatesh M (2020) Prediction of cardiovascular disease using machine learning algorithms. *Int J Eng Adv Technol (IJEAT)* vol 9(3)
12. Shah D, Patel S, Bharti SK (2020) Heart disease prediction using machine learning techniques. *SN Comput Sci* 1:345. <https://doi.org/10.1007/s42979-020-00365-y>
13. Folorunso SO, Awotunde JB, Adigun AA, Panigrahi R, Garg A, Bhoi AK (2023) Multi-label learning model for diabetes disease comorbidity. *J Inst Eng (India): Ser B*, 1–13
14. Shouman M, Turner T, Stocker R (2012) Applying K-nearest neighbour in diagnosing heart disease patients. *Int J Inf Educ Technol* 2(3):220

15. Awotunde JB, Folorunsho O, Mustapha IO, Olusanya OO, Akanbi MB, Abiodun KM (2023) An enhanced internet of things enabled type-2 fuzzy logic for healthcare system applications. Recent trends on type-2 fuzzy logic systems: theory, methodology and applications. Springer International Publishing, Cham, pp 133–151
16. Pouriyeh SE (Jul 2017) A comprehensive investigation and comparison of machine learning techniques in the domain of heart disease. In: Proceedings of the IEEE symposium on computers and communications (ISCC). Heraklion, Greece IEEE, pp 204–207
17. Medhekar D, Bote M, Deshmukh S (2013) Heart disease prediction system using naive Bayes algorithm. Int J Innov Sci Eng Technol 2(3):1–5
18. Rjeily CB, Badr G, Hajjam A, Andre's E, Hajjarm A, Hassani E, Andres E (2019) Medical data mining for heart diseases and the future of sequential mining in the medical field, vol 149. Springer, pp 71–99. <https://doi.org/10.1007/978-3-319-94030-44>
19. Vembandasamy K, Sasipriya R, Deepa E (Jul 1988 [online]) Heart disease data set. <http://www.archive.ics.uci.edu/ml/dataset/heart+disease>

Unmanned Aerial Vehicles Detection Using CNN-Transfer Learning



John Irungu, Hubert Smith, Jose Cruz, Thabet Kacem, and Anteneh Girma

Abstract The high rate of intrusion of private and controlled air environments by unmanned aerial vehicles (UAVs) pose a serious security threat to radar, visual and acoustics systems that may be used to carry surveillance or intrude confined spaces and installations. To address the misuse of recreation drone technology resulting from the proliferation of privately owned UAVs, we investigate in this paper how to prevent UAV privacy intrusion, damage and security threats around confined spaces using a pre-trained Convolutional Neural Network (CNN) Image Recognition model. A binary image classification model is tested to distinguish the presence of drones' images in an aerial environment from other flying and aerial objects. Our experimental test was based on a pre-trained Transfer learning model, Mobile Net-V2, implemented with a binary classification achieving an accuracy of 93% in detection drones. Performance evaluation also shows a 93%, 90% and 94% score for the F1-score, Recall and Precision, respectively.

Keywords UAV · Transfer learning · Image classification · CNN

J. Irungu · H. Smith · J. Cruz · T. Kacem (✉) · A. Girma

Department of Computer Science and Information Technology, University of the District of Columbia, Washington, DC 20008, USA

e-mail: thabet.kacem@udc.edu

J. Irungu

e-mail: john.irungu@udc.edu

H. Smith

e-mail: hubert.smith@udc.edu

J. Cruz

e-mail: jose.cruz2@udc.edu

A. Girma

e-mail: anteneh.girma@udc.edu

1 Introduction

According to the Federal Aviation Authority (FAA) there are 1.6 million commercial and privately owned drones in the United States [1]. The exponential growth in use of unmanned aerial vehicles by private persons and enterprises has several uses including visual shooting capabilities for construction projects, data collection for researchers, and aerial safety. The estimated annual growth rate of 20% in drone sales is projected to reach \$55.6 billion in 2027, which constitutes an increase of \$19.5 billion since 2020 [2]. In the United States, by January 2022, there were 259 certified pilots and 853,000 registered drones constituting 38% commercial drones and 62% recreational drones, respectively. Other sectors that have embraced non-commercial and non-recreational drone use include the military and the scientific industry. For military purposes, drone technology is vital for surveillance, air strikes and intelligence gathering while in scientific research drones have become ubiquitous in tasks such as agricultural survey and geo-mapping, habitat and wildlife research, and documenting deforestation. Among the factors identified and attributed to the rapid growth of drone technology include optimization, speed, cost, flexibility, and accuracy in aerial use. The aforementioned factors improved the performance of missions' accuracy, reduced the air mobility costs and improved the flexibility of geographical reach in confined or disadvantaged topographies.

Although the use of drone technology has its socio-economic, the proliferation of UAVs has resulted also in some misuse of private drones thereby posing vulnerabilities and threats such as security risks and privacy intrusions, unsolicited surveillance, damage, and legal liabilities. A more versatile challenge of unmanned aerial vehicles is the Internet of Drones (IoD) technology, which is a decentralized network that links drones to inter-location services and controlled air spaces [3]. IoD which is vulnerable to IoT network intrusion, may pose a cyber security threat in the IoD layered network control architecture exposing IoD infrastructure enterprise. An example is a localization error attack which may prohibit accurate and reliable positioning of drones, and this may affect the performance of the IoD network and reliability. IoD is expected to play a consequential role in future smart cities hence, cyber vulnerabilities to IoD drone links may paralyze infrastructure in these cities.

Other malicious uses of drones include intruding privacy and theft of stealth information, which gives another compelling reason to ensure that this technology is regulated. In one documented instance, a teenager modified a drone to a fire handgun creating a harmful aerial weapon [4]. Although drones technology is used for military aviation surveillance and attacks, redesigning unmanned aerial vehicles as weapons may be detrimental to public safety. In one experiment carried out at DEFCON conference, as shown in Fig. 1, a researcher attached a WiFi pineapple to a Phantom drone in order to sniff wireless radio signals. In the experiment, the pineapple device was attached to the drone flying over and maintain a 3G WiFi connection by creating reverse SSH relay server to sniff wireless signals.

To address the existing regulatory gaps in unmanned aerial vehicles, there is a need to design and develop more automated and reliable UAV detection capabilities

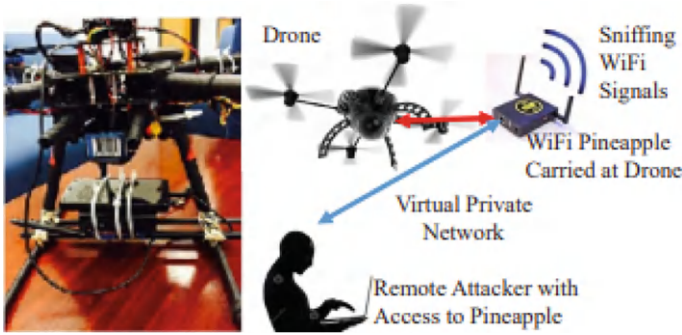


Fig. 1 WiFi pineapple mounted on a drone in an attack model [4]

that can match the sophisticated and stealth privacy intrusion capabilities that private drones may pose. In this paper, we developed an artificial intelligence model that detects UAV intrusion using image classification. Using an available public dataset of birds and drones, this experimental study focuses on drone detection using binary classification. Performance evaluation of the model is performed to analyze the efficiency of the model in detection images of UAVs against other aerial species i.e., Birds. This experiment compared to other drone detection works demonstrates the gained edge of our Mobile Net-V2 pre-trained model in image classification compared to other models.

The rest of the paper is organized as follows. Section 2 is the related works. Section 3 describes the dataset and our methodology. The experimental results are shown in Sect. 4. In Sect. 5, we conclude our paper and provide an outlook toward our future work.

2 Related Work

In this paper, the following related works provides a comprehensive overview of recent research articles on drone detection using various techniques such as audio-based detection, multi-camera strategies, autonomous long-range systems, automated algorithms like YOLOv4, and machine learning-based methods. The reviewed articles highlight the importance of utilizing advanced technologies such as deep learning algorithms and generative adversarial networks to enhance dataset quality and improve the accuracy of drone detection and identification. Despite their numerous advantages, unauthorized or malicious use of drones [5] can lead to privacy infringement, safety concerns, and potential threats to critical infrastructure. Therefore, developing effective drone detection and identification systems has become a crucial research area.

Zhang et al. [7] suggested the use of popular audio-based techniques usage for drone aerial detection and vehicle identification. The authors commonly referred

to propose a method of drones that have skyrocketed deep across learning various techniques, specifically generative military adversarial operations, networks surveillance (GANs), photography, and enhancing the delivery of the services. This research explores the use of multiple camera setups combined with deep learning algorithms for drone detection and tracking. The authors investigate different approaches, including object detection methods like Faster R-CNN, YOLO, and SSD, to identify drones from real-time video streams captured by multiple cameras installed at various locations.

Al-Emadi et al. [23] focused on using deep learning techniques combined with generative adversarial networks (GANs) to enhance the dataset for audio-based drone detection and identification. This approach harnesses the power of artificial intelligence to improve the accuracy and reliability of detecting and identifying drones based on their audio signatures.

Unlu et al. [22] proposed another approach to drone detection, which involves utilizing multiple cameras in conjunction with deep learning algorithms. This strategy aims to detect and track drones more effectively by analyzing visual data captured from different angles, enabling timely response measures.

To ensure safety in critical infrastructure areas an independent long-range drone detection system is proposed by Zhang and Kusurini [24]. The proposed system leverages advanced technologies, including artificial intelligence and machine learning algorithms, to detect and classify drones operating within a specified range. This system can provide real-time caution and help prevent implicit security breaches in critical areas by automating the detection process.

Singha and Aydin presents an automated approach to drone detection using YOLOv4 [6]. The authors utilized the YOLOv4 (You Only Look formerly) algorithm, known for its high accuracy and effectiveness in object detection tasks. The authors proposed an end-to-end system that combines object discovery with profound learning ways to directly identify drones from images or videos captured by surveillance cameras. Also, the study highlights the advantages of YOLOv4 in terms of speed and accuracy compared to former versions.

Taha and Shoufan [8] presented an overview of machine learning drone detection and classification advancements. The authors discuss various ways, including deep learning algorithms, support vector machines, and arbitrary forest classifiers, that have been employed to detect and classify drones based on their visual features. The study presented a comprehensive analysis of the strengths and limitations of different classification methods, highlighting the significance of combining multiple modalities for robust drone detection and classification. Additionally, the authors discuss implicit challenges in real-world scenarios, such as cluttered backgrounds, varying lighting conditions, and camouflage ways drones use. This comprehensive review highlights the current state of exploration in this field and identifies implicit areas for further enhancement.

In conclusion, the articles we reviewed highlight different approaches to drone discovery using a variety of methods. These studies demonstrate the significance of

leveraging advanced technologies such as deep learning algorithms and generative inimical networks to enhance dataset quality and improve the accuracy of drone discovery and identification.

However, despite advancements in this field, challenges still need to be addressed. These include mitigating false favorable rates, improving real-time response measures, enhancing system scalability for large-scale deployments, and conforming to evolving drone technologies. The ethical implications surrounding privacy violations and the implicit abuse of these detection systems also need careful consideration.

3 Methodology

3.1 Data Description

In the paper, we used a dataset containing drone and bird images to perform the binary classification using a Convolutional Neural Network model. The public image dataset from Kaggle [9] contains two folders of 400 birds images and 428 drone images present in the sky. The images used in this dataset are in .jpeg format are pre-processed for data analysis. The bird images in the study were scrapped while the drones' images were extracted from a different dataset. The data was then loaded in a Python. To start pre-processing, we used NumPy, Pandas, Sklearn metrics for data analysis and Matplotlib for Visualization. We also used TensorFlow [10] API for machine learning and artificial intelligence.

The Bird and Drone dataset path was loaded and split into 3 categories namely, train, validate and tests dataset and converted into a data frame with its own labels. After loading the sample images, a directory was created for the lists images and appended into a data frame for balancing and labeling of the data into train, tests and split. For the drone images the train_test_split images were split in a ration of 72%, 14% and 14% respectively. Whereas for the bird images the train, test, split ratio was at 75%, 12.5% and 12.5% respectively. Figure 2 shows a sample of the drones and birds' images that were used in this experiment.

3.2 Data Augmentation

Using sklearn metrics module, the training, validation, and test dataset was split into x predictors and y labels with a target size of 256 and a batch size of 32 for a binary mode classification of the two classes. Keras ImageDataGenerator is used to augment the data due to the small number of images by improving in the model generalization for training and validation. Data augmentation is used to apply multiple transformations to original images resulting to similar copies of the original image through flipping ,rotating and shifting [11]. In this study, Keras ImageData-Generator which has unique augmentation techniques such as brightness resolution,

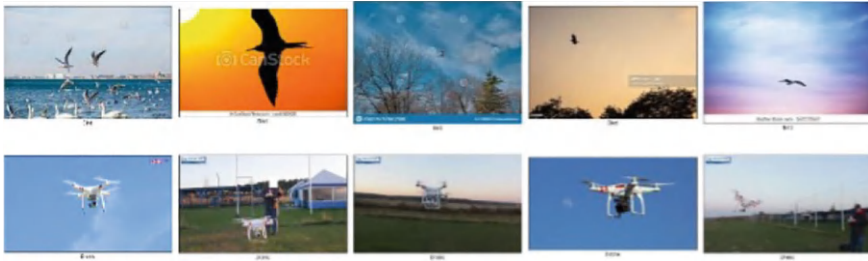


Fig. 2 Randomly selected batch of 5 sample images under the Drone and Bird Images data used in the experiment

standardization among others was used to transform the Drone and Bird images in the experiment.

3.3 MobileNet-V2 Transfer Learning Model

After collecting the dataset, performing the data augmentation and splitting the dataset, the core of our approach relies on the CNN transfer learning module. For that effect, we leveraged MobileNet-V2, which is a Google model used as a convolution neural network to train on the augmented drone and bird images dataset.

The pre-trained network has the ability to classify 1000 object categories using feature representation of the images [12]. The input size of the network is 224 by 224. Mobilenet-V2 used in this experiment has better-performing object detection capabilities resulting from its architecture. The neural network consists of 2 blocks of stride 1 (one-pixel) and 2 (two-pixels) moves by the filter and with 3 layers each for each block [13]. A 1 by 1 convolution with rectified linear unit (RELU), the second layer is a depth wise convolution and the third a 1×1 convolution devoid of non-linearity. The Eq. 1 shows the rectified linear unit function.

$$f(0) = \max(0, o_i) \quad (1)$$

where $f(0)$ is the ReLu function [11].

In our approach, depicted by Fig. 3, we used the pre-trained convolution neural network, MobileNet-V2 with a 2D convolution layer (Conv2D) with 3 layers and a ReLu activation function. Batch Normalization was also implemented to normalize the data to zero mean and unit variance. The purpose of normalization is to eliminate feature values imbalance through scaling thereby eliminating bias during training. In neural networks, batch normalization is essential since they normalize the activation from hidden input of each layer 1 (before they reach layer 2) [14] making the gradient descent converge effectively during training. In the output layer of the pre-trained MobileNetV2 neural network, the sigmoid function was used to convert

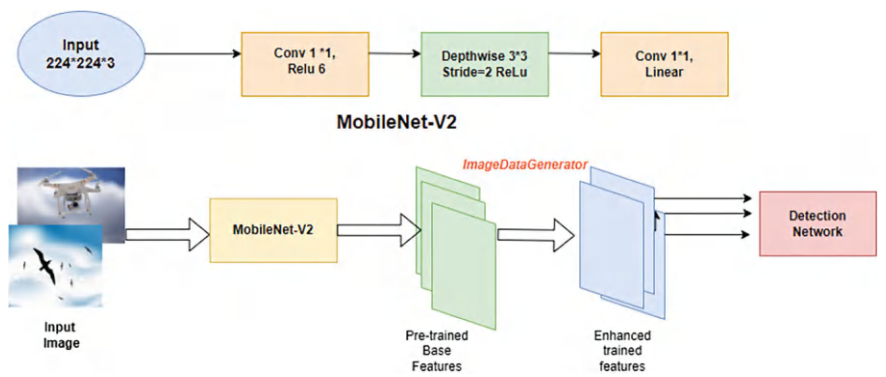


Fig. 3 Proposed MobileNetV2 CNN architecture

the output of the linear model into probability which is used in the experiment’s binary classification [15] of whether the image was a bird or a drone. The Sigmoid activation function below has a value of between 0 and 1 given by Eq. 2:

$$f(x) = \frac{1}{1 + e^{-x}}$$

(2)

The convolutions neural network was fitted for training. The model was trained at 25 epochs and achieved an accuracy rate of 93%.

4 Experimental Results

After training, testing, and validating the model for drone detection, a predicted accuracy of 93% for drone detection was achieved as shown in the following results in Table 1. Further performance evaluation was done using the following evaluation metrics on Eqs. 3, 4, and 5 listed below [16].

Table 1 Classification summary of unmanned aerial vehicles detection

	Precision	Sensitivity	F1-score	Support	Confusion matrix
Birds	0.94	0.90	0.92	50	45 5
Drone	0.92	0.95	0.93	60	3 57
Accuracy			0.93	110	
Macro avg	0.93	0.93	0.93	110	
Weighted avg	0.93	0.93	0.93	110	
Accuracy	93.023%				

F1-Score represents the average of the precision and sensitivity taking false positives and false negatives. F-1 score is given by Eq. 3:

$$F1 - Score = \frac{TP}{TP + \frac{FP+FN}{2}} \quad (3)$$

Precision is the measure of the positive predictions made of true positives is given by Eq. 4:

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

Recall, also known as sensitivity, is a measure of how many positive cases were correctly predicted by the classifier. i.e., the overall number of positive cases, as shown in Eq. 5

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

As shown in Table 1, the model achieved a drone precision score of 92%, a recall score of 95%, and an F1-score of 93% which compared to other related works [6] was considerably high. The summary of the classification results is shown in Table 1.

To validate the experimental results, validation of the predicted valuables was carried out using the validation accuracy and load metrics. We computed the validation accuracy and loss, to estimate the fit of the model at 25 epochs, as shown in Figs. 4 and 5, respectively. The training and validation loss plots provide insights into how the learning performance of changes is based on different number of epochs. A high number of epochs may lead to an increase in the fit of the model, also known as over-fitting, while a low number of epochs may lead to under-fitting. Therefore, the validation accuracy and loss plots shows the favorable number of epochs to use the training weights [17].

Fig. 4 Validation accuracy

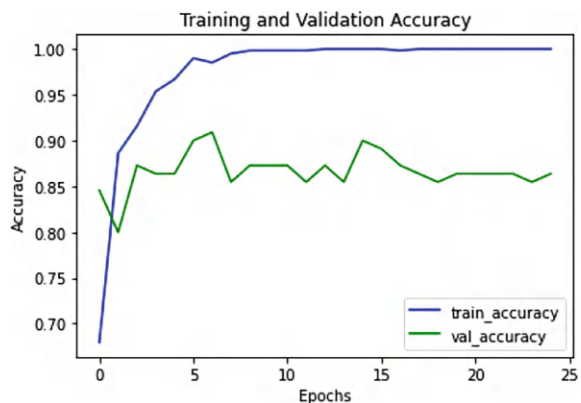


Fig. 5 Validation loss

To further validate the drone detection binary classification results of the MobileNet-V2 transfer learned model for UAV detection, a receiver operating curve (ROC) is plotted to evaluate the performance of the prediction model, as shown in Fig. 6. In this experiment, a receiver operating curve quantifies the accuracy of the experimental tests in distinguishing the bird and drone object identification.

The ROC curve shows the relationship between the false positives fraction (FPF) and the true positives fraction (TPF) [18]. In our case, the 0.98 Area Under the Curve (AUC) score indicates a 98% performance in distinguishing the birds from the drones and (vice versa). This AUC score is an excellent model in the detection of drone since it means that the pre-trained MobileNet-V2 object detection model using Convolutional Neural Network had a good measure of separability in the binary classification [19]. This means that there is a 98% chance that the model is able to distinguish a bird from a drone.

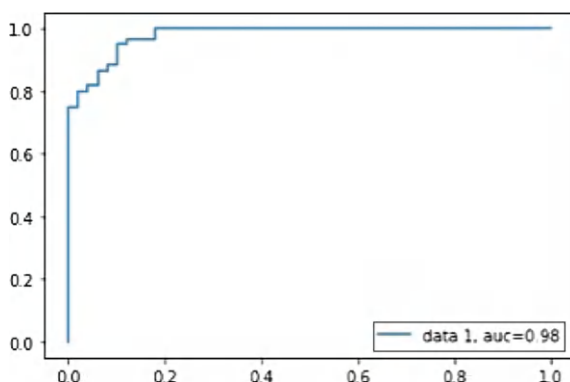
Fig. 6 Receiver operating curve for drone detection model

Table 2 Performance evaluation comparison with related work

Authors	Dataset	Technique	Performance
Jeong et al.	Audio Drone Sound Data	RNN, CNN, GMM	Recall: 84%, F1-Score: 80%, Precision: 90.31%
Jeon et al.	Google and Kraggle dataset	YoloV4	Recall: 84%, F1-Score: 80% and Precision: 90.31%
Samadzadegan et al.	Bird, helicopter and multi-rotor dataset	YoloV4, CNN	Accuracy: 83%, F1-Score: 83% and Precision: 84%
Our work	Kragle dataset	MobileNetV2, CNN	Accuracy: 93.8%, F1-Score: 95%, Precision: 93%

5 Discussion

Compared to other studies on drone detection our study achieved a higher detection rate as shown in Table 2. For example, Singha [6] who used YoloV4 image detection method achieved a recall of 68% and an F1-score of 78% in the detection of drones. Compared to Singha, in a similar binary classification for drone detection, our model performance evaluation had a recall, and F1-score was 95% and 93%, respectively showing a better drone detection.

Although audio detection of drone signals is effective in detecting the presence of drones, this present approach was compared to audio analysis methods using the same artificial neural network for drone detection. As shown in a related empirical study by Jeon [20] where drone sounds were detected in a real-life environment using deep neural networks, the use of deep learning models such as CNN and Recurrent Neural Networks (RNN) had a lower detection capability compared to our study that relies on pre-trained CNN model using MobileNetV2. Jeon attributes one of the challenges of audio classification as the background data which may weaken the drone sound characteristics. Although deep learning has high computational resources, our paper shows that the use of image classification and precisely a pre-trained models may have a better detection rate than raw audio data with background noise reflections.

Samadzadegan [21] work on detection and recognition of drones using image based Convolution Neural Networks evaluated an image recognition approach to detect drones by distinguishing 2 different types of drones from each other and also from birds. In the study, which consisted of 10,000 images of drones and multi-rotor, Samadzadegan achieved an accuracy rate of 83% using deep learning techniques and a recall and an F1-score of 84% and 83%. Although the study used the same computing resources as our present study, our MobilenetV2 model had a higher performance in the detection of drones, promising a better detection using image classification.

6 Conclusion

The use of unmanned aerial vehicles (UAVs) has been gaining traction from industry players due to the security and commercial benefits that Unmanned Aerial Vehicles and systems realize in enterprise and security sectors. However, although the proliferation of UAVs comes with its challenges, especially from private users, there is a need to create a detection model that can detect the presence of UAVs in confined places or secure aerial environments, which may be subject to privacy intrusion or information theft using signal eavesdropping or camera surveillance (live stream and live video and image capture) using UAVs [5]. In this paper, we were able to successfully design and test a neural network model that can solve this challenge by detecting the different types of drones using a transfer learning image classification based on the Mobile Net-V2 model. Our results show that we achieved a considerably higher score in the prediction of the drone's image from other more similar objects.

In the future, we plan to design, develop and test an audio detection model and compare the model performance of the audio detection to image classification of drones using artificial neural networks.

References

1. Drone Events | Federal Aviation Administration. Accessed: Sep 26, 2023. <https://www.faa.gov/uas/events>
2. Sabino H, Almeida RV, de Moraes LB, da Silva WP, Guerra R, Malcher C, Passos D (2022) Passos FG (2022) A systematic literature review on the main factors for public acceptance of drones. *Technol Soc* 71:102097. <https://doi.org/10.1016/j.techsoc.2022.102097>
3. Derhab A, Cheikhrouhou O, Allouch A, Koubaa A, Qureshi B, Ferrag MA, Maglaras L, Khan FA (2021) Internet of drones security and privacy issues: taxonomy and open challenges. *IEEE Access* 9:57243–57270. <https://doi.org/10.1109/ACCESS.2021.3072030>
4. Vattapparamban E, Güvenç İ, Yurekli AI, Akkaya K, Uluğaç S (2016) Drones for smart cities: issues in cybersecurity, privacy, and public safety. In: 2016 international wireless communications and mobile computing conference (IWCMC), Sep 2016, pp 216–221. <https://doi.org/10.1109/IWCMC.2016.7577060>
5. Yaacoub J-P, Noura H, Salman O, Chehab A (2020) Security analysis of drones systems: attacks, limitations, and recommendations. *Internet Things* 11:100218. <https://doi.org/10.1016/j.iot.2020.100218>
6. Al-Emadi S, Al-Ali A, Al-Ali A (2021) Audio-based drone detection and identification using deep learning techniques with dataset enhancement through generative adversarial networks. *Sensors (Basel)* 21(15):4953. <https://doi.org/10.3390/s21154953>. PMID: 34372189; PMCID: PMC8348319
7. Drones | free full-text | automated drone detection using YOLOv4. Accessed: Oct 03, 2023. <https://www.mdpi.com/2504-446X/5/3/95>
8. Taha B, Shoufan A (2019) Machine learning-based drone detection and classification: state-of-the-art in research. *IEEE Access* 1. <https://doi.org/10.1109/ACCESS.2019.2942944>
9. Birds vs drone dataset. Accessed: Sep 27 2023. <https://www.kaggle.com/datasets/harshwalia/birds-vs-drone-dataset>
10. TensorFlow. TensorFlow. Accessed: Sep 27 2023. <https://www.tensorflow.org/>

11. Bhandari A (2023) Image augmentation on the fly using Keras ImageDataGenerator!. Analytics Vidhya. Accessed: Sep 27 2023. <https://www.analyticsvidhya.com/blog/2020/08/image-augmentation-on-the-fly-using-keras-imagedatagenerator/>
12. MobileNet-v2 convolutional neural network - MATLAB mobilenetv2. Accessed: Oct 02 2023. <https://www.mathworks.com/help/deeplearning/ref/mobilenetv2.html>
13. Review: MobileNetV2 - light weight model (image classification) | by Sik-Ho Tsang | Towards data science. Accessed: Oct 02 2023. <https://towardsdatascience.com/review-mobilenetv2-light-weight-model-image-classification-8febb490e61c>
14. Doshi K (2023) Batch norm explained visually - how it works, and why neural networks need it. Medium. Accessed: Oct 02 2023. <https://towardsdatascience.com/batch-norm-explained-visually-how-it-works-and-why-neural-networks-need-it-b18919692739>
15. Vishwakarma S (2023) Why is sigmoid function important in artificial neural networks?. Analytics Vidhya. Accessed: Oct 02 2023. <https://www.analyticsvidhya.com/blog/2023/01/why-is-sigmoid-function-important-in-artificial-neural-networks/>
16. Irungu J, Oladunni T, Denis M, Ososanya E, Muriithi R (2023) A CNN transfer learning-electrocardiogram (ECG) signal approach to predict COVID-19. In: 2023 15th international conference on computer and automation engineering (ICCAE). IEEE, pp 367–371
17. Cristina S (2023) Plotting the training and validation loss curves for the transformer model. MachineLearningMastery.com. Accessed: Oct 02 2023. <https://machinelearningmastery.com/plotting-the-training-and-validation-loss-curves-for-the-transformer-model/>
18. Hajian-Tilaki K (2013) Receiver operating characteristic (ROC) curve analysis for medical diagnostic test evaluation. *Casp J Intern Med* 4(2):627–635
19. Narkhede S (2023) Understanding AUC - ROC curve. Medium. Accessed: Oct 03, 2023. <https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5>
20. Jeon S, Shin J-W, Lee Y-J, Kim W-H, Kwon Y, Yang H-Y (2017) Empirical study of drone sound detection in real-life environment with deep neural networks. arXiv, Jan 20, 2017. Accessed: Oct 03 2023. <http://arxiv.org/abs/1701.05779>
21. Samadzadegan F, Dadrass Javan F, Ashtari Mahini F, Gholamshahi M (2022) Detection and recognition of drones based on a deep convolutional neural network using visible imagery. *Aerospace* 9(1):Art. no. 1 (2022). <https://doi.org/10.3390/aerospace9010031>
22. Unlu E, Zenou E, Riviere N, Dupouy PE (2019) Deep learning-based strategies for the detection and tracking of drones using several cameras. *IPSN Trans Comput Vis Appl* 11(1):1–13
23. Al-Emadi S, Al-Ali A, Al-Ali A (2021) Audio-based drone detection and identification using deep learning techniques with dataset enhancement through generative adversarial networks. *Sensors* 21(15):4953
24. Zhang X, Kusurini K (2021) Autonomous long-range drone detection system for critical infrastructure safety. *Multimedia Tools Appl* 80(15):23723–23743

Learning and Defense

Multi-head Temporal Learning for Human Activity Recognition Using Smartphone



So Zheng Wei, Pang Ying Han, Khoh Wee How, and Ooi Shih Yin

Abstract Human activities can be grouped into basic activities, complex activities and postural transitions within or between basic activities. In the literature, fewer works are focusing on postural transition recognition. Postural transitions, i.e., stand-to-sit and sit-to-stand, etc., are regular functional tasks indicative of muscle balance and power performance. Hence, examining postural transitions delivers an objective tool for a human mobility assessment. In this paper, a smartphone-based human activity recognition architecture with the utilization of multi-head temporal convolutional operators is proposed. In this work, a custom-tailored learning architecture for each sensor type, i.e., accelerometer and gyroscope sensors in this case, is developed. Feature learning is enhanced by deliberating on a single sensor instead of on all at once. This allows better data characterization of the multichannel inertial data. Empirical results reveal the superiority of the proposed model to the existing machine and deep learning models, with an accuracy of 98.35%.

Keywords Multi-stream · Temporal deep learning · Smartphone · Human activity recognition

S. Z. Wei · P. Y. Han (✉) · K. W. How · O. S. Yin

Faculty of Information Science and Technology, Multimedia University, Jalan Ayer Keroh Lama, 75450 Ayer Keroh, Melaka, Malaysia

e-mail: yhpang@mmu.edu.my

S. Z. Wei

e-mail: 1181201610@student.mmu.edu.my

K. W. How

e-mail: whkhoh@mmu.edu.my

O. S. Yin

e-mail: syooi@mmu.edu.my

1 Introduction

In the past decade, human activity recognition (HAR) research has drawn considerable interest in enhancing recognition performance and increasing inference speed. HAR has been applied to various smart assistive systems, such as behaviour monitoring, emergency medical conditions, health care, elderly care, fitness, etc. [1–3]. Although there are numerous approaches for effective HAR, these approaches typically focus on basic human activities but disregard postural transitions.

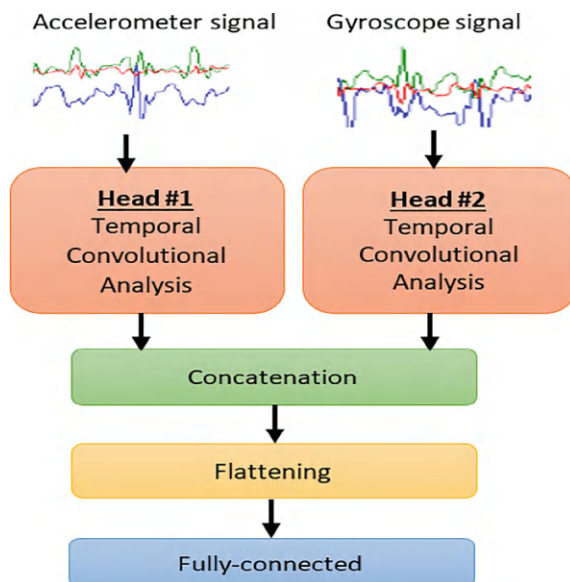
Basically, we could categorize human activities into three spheres: basic activities (i.e., walking, running, standing, etc.), complex activities (i.e., cycling, cooking, eating, etc.) and postural transition within or between basic activities (i.e., stand-to-sit, sit-to-stand, stand-to-walk, walk-to stand, etc.). In the literature, there are fewer works performed on postural transitions; most of the works consider basic and complex activities. Postural transitions, such as stand-to-sit and sit-to-stand, are frequent daily functional tasks indicative of muscle balance and power performance [4]. Examining postural transitions delivers an objective tool for a human mobility assessment.

In this paper, a smartphone-based HAR system is proposed. In this system, an architecture with the utilization of multi-head temporal convolutional operators is developed to detect human basic activities and postural transitions. The input data are multichannel time series signals acquired from different types of inertial sensors which are embedded in smartphones. With multiple heads, a temporal convolutional analysis is performed in each head to implement distinctive independent feature learning according to different types of input signals captured by different sensors. This allows a custom-tailored learning architecture for each type of sensor, i.e., accelerometer and gyroscope sensors in this case. Such learning is significant in characterizing this multichannel time series inertial data. The proposed system is termed Multi-head Temporal Convolutional Model. Figure 1 illustrates the overview of the system.

2 Related Work

There are plentiful works regarding human activity in different techniques. The techniques can be categorized into machine learning and deep learning models [5–11]. In the former approach, feature engineering is a crucial step to extract salient features since the captured raw inertial signals are not in the appropriate form with high fluctuation and oscillation. Hence, manually computed statistical features are generated to resolve the problem. The handcrafted features will be input into machine learning classifiers for human activity classification [12]. These handcrafted techniques heavily depend on the prior knowledge of domain experts and are not task-dependent. Over the years, deep learning approaches have been extensively applied to various domains, including HAR, due to the automatic feature learning capability.

Fig. 1 Overview of the proposed MTCM architecture



For instance, Convolutional Neural Network (CNN)-based models were applied in smartphone-based HAR by utilizing the underlying characteristics of human activities and the multichannel time series signals [13, 14]. Encouraging performances were reported with these CNN-based architectures on basic human activities, as well as postural transitions.

In addition, Thu and Han developed a hybrid method by integrating discrete wavelet transform and bi-directional long short-term memory (BiLSTM) on postural transition recognition [15]. In the system, similar-pattern transitions were grouped as the same groups. Then, a discrete wavelet transform was implemented for feature extraction. The extracted features were then further classified by using BiLSTM. The authors reported that the proposed system was able to achieve competitive performance compared to a non-transition model. Ahmed et al. proposed an efficient feature extraction technique with Static Postural Transition-based Post-Processing (SPTPP) [16]. The recognition performance was improved with the implementation of the proposed post-processing technique by analysing the existence of postural transition from the activity history to the current state.

Olena et al. proposed a transfer learning approach based on continuous wavelet transformation (CWT) [17]. The model was pre-trained on scalograms computed using CWT. The transfer learning accelerated the model training process with smoother gradient descent on small databases. In this work, nine CNN architectures with different CWT configurations were taken into account for the best-performing combination. The empirical result showed that the proposed transfer learning model achieved a promising performance on the postural transition database, i.e., HAPT, with about 92% accuracy.

3 Multi-head Temporal Convolutional Model

IMU (Inertial Measurement Unit) inertial signal is a multi-channel time series data that provides three-dimensional (i.e., three channels with x -, y - and z -axis) motion signal patterns captured by accelerometer and/or gyroscope. Different sensors provide information of different natures, the former sensor senses axis orientation, whereas the latter sensor senses angular orientation. Hence, the time series data captured by each sensor should be treated independently for a more custom-tailored architecture for each sensor type.

Time series data is a one-dimensional (1D) vector. Thus, one-dimensional convolutions are always applied for the kernels moving over the vector in a single dimension, i.e., time dimension. To process these multiple time series data, a multi-head temporal convolutional analysis is performed by implementing multiple one-dimensional convolutions with a single sensor, respectively. Specifically, the acceleration and angular velocity data captured by the accelerometer and gyroscope are respectively fed into each head for independent temporal convolutional analysis to encode the underlying temporal features of motion acceleration and angular velocity, correspondingly. Four Convolutional Blocks (Conv Blocks) are embedded in each temporal convolutional analysis head, as illustrated in Fig. 2. After that, the produced feature maps for each sensor are concatenated, flattened and classified in a fully-connected layer.

Each Conv Block comprises of 1D dilated causal convolution, ReLU activation function, batch normalization and spatial dropout for regularization, see Fig. 3. The implementation of dilated causal convolution can effectively allocate the network a flexible receptive field size to adapt different signals. The dilated convolutions could enlarge the receptive field while constructing feature maps of multi-scale receptive fields with different dilated rates for different data interpretation. On top of that, the incorporation of the dilated causal convolution operator in MTCM helps prevent data leakage from the future to the past. This facilitates long-term dependency preservation which is crucial in analysing the inertial motion data.

Batch normalization facilitates quicker and more stable model training. Besides that, it possesses a regularizing effect due to the noise added to the inputs of each layer. This helps reduce overfitting because the model does not produce deterministic values for the training example alone. ReLU activation function ensures the model learns non-linearities in the data and spatial dropout is a kind of regularization for curbing the vanishing gradients problem. Further, we also embed 1×1 convolution to account for the discrepancy of the input–output lengths.

In this work, a softmax classifier is adopted to classify the extracted features. The probability of each class is computed with the range from zero to one. The activity class that scores the highest probability is classified as the target class. Softmax activation is formulated as follows,

$$\sigma(\tilde{x})_i = \frac{e^{x_i}}{\sum_{j=1}^C e^{x_j}} \quad (1)$$

Fig. 2 The overview of multi-head temporal convolutional model (MTCM)

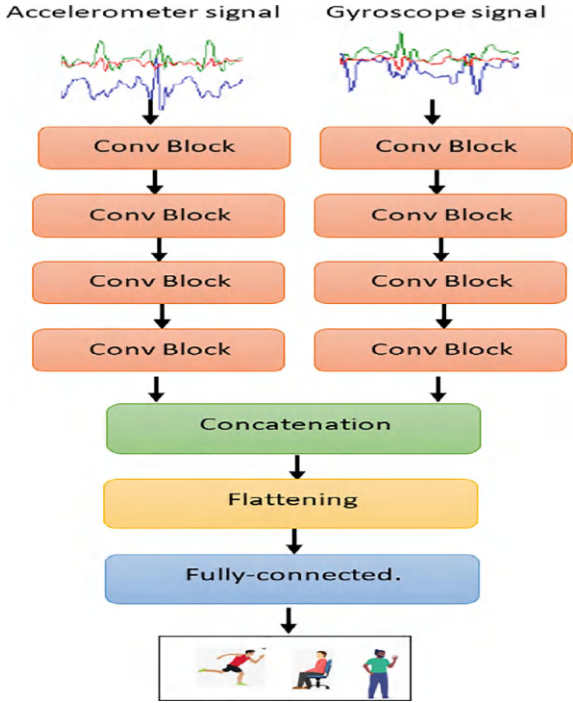
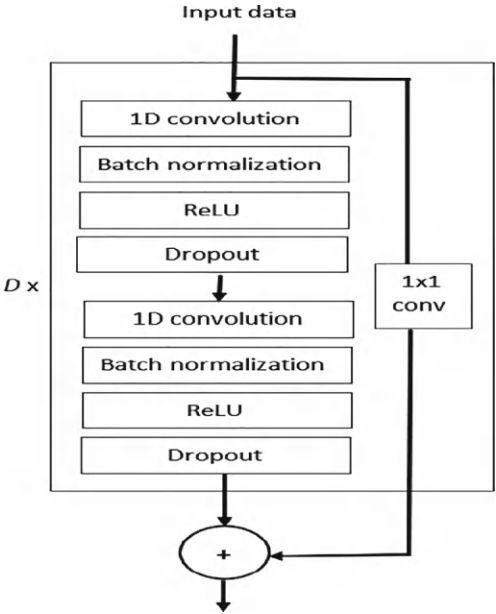


Fig. 3 The structure of a convolutional block (conv block)



where e^{x_i} is the exponential function of the i th input, C is the number of activity classes and e^{x_j} is the exponential function of the j th output vector.

4 Database

UCI HAR dataset, i.e., Smartphone-based Recognition of Human Activities and Postural Transitions Dataset (HAPT), was adopted to assess the efficacy of the proposed model [1]. Different from the first version of the UCI HAR dataset, HAPT is an updated version with six basic activities and six postural transitions that occurred between the static postures:

Basic static postures:

Lying, sitting, standing

Basic dynamic postures:

Walking, walking upstairs and walking downstairs

Postural transitions:

Sit-to-stand, stand-to-sit, lie-to-sit, sit-to-lie, stand-to-lie and lie-to-stand.

This dataset was conducted with thirty volunteers (aged from 19 to 48 years old). A Samsung Galaxy S II smartphone was adopted as the data acquisition device. Three-axial angular velocity and three-axial linear acceleration were captured at a constant rate of 50 Hz. After that, both signals were pre-processed by applying noise filters. 2.56 s fixed-width sliding windows with 50% overlapping were applied on the pre-processed signals to produce 128 readings per window. These sampled data were the input of the proposed model.

5 Experimental Results

The performance of the proposed model is reported in this section. First, the effect of dilation factors on the recognition performance is analysed. Table 1 records the performance of different dilation factors in the proposed MTCM. It is observed that there is a slight influence on the dilation factors towards the feasibility of the proposed model. We adopt dilation factor set of {1, 2, 4, 8 and 16} in our deep learning architecture for the subsequent experiments.

Next, the performance of the proposed MTCM for recognizing each activity, comprising basic dynamic activities, basic static activities and postural transitions, is analysed. Table 2 records the performance results in terms of precision, recall, F1 score and accuracy. On top of that, we also include the confusion matrix of each activity in Fig. 4. From the confusion matrix, it is observed that there are 13 out

Table 1 Performance results of different dilation factors in MTCM

Dilation factors	Precision	Recall	F1 score	Accuracy (%)
1, 2, 4	0.9812	0.9796	0.9804	98.15
1, 2, 4, 8	0.9820	0.9789	0.9804	98.10
1, 2, 4, 8, 16	0.9788	0.9834	0.9808	98.35
1, 2, 4, 8, 16, 32	0.9859	0.9847	0.9852	98.20

of 255 sitting samples are misclassified as standing class. This could be because both activities are still, and the sensor readings have similar patterns, as illustrated in Fig. 5. Furthermore, it is observed that the proposed model performs exceptionally in identifying basic dynamic activities. There is no misclassification of the human activity classes of walking, walking upstairs and walking downstairs by the proposed MTCM. The reason could be the signal patterns of these activities are distinct and distinguishable (see Fig. 6). Overall, we can deduce that the proposed MTCM is able to extract the discriminative features of these activities, leading to promising performance in human activity recognition.

Finally, we also conduct a performance comparison between the proposed model with the existing HAR models on this human activity and postural transition database. Table 3 records the performance comparison. From the table, we notice that Stacking Ensemble Learning introduced by Ramanujam and Rasikannan [12] demonstrates good performance on the HAPT database. In this ensemble learning model, Support

Table 2 Performance results of MTCM with different types of activities

Activities	Precision	Recall	F1 score	Accuracy (%)
<i>Basic dynamic activities</i>				
Walking	1.0	1.0	1.0	100
Walking upstairs	1.0	1.0	1.0	100
Walking downstairs	1.0	1.0	1.0	100
<i>Basic static activities</i>				
Sitting	0.94	0.96	0.95	98.70
Standing	0.97	0.95	0.96	98.80
Laying	1.0	1.0	1.0	99.95
<i>Postural transitions</i>				
Stand-to-sit	0.96	1.0	0.98	99.85
Sit-to-stand	1.0	0.91	0.95	99.90
Sit-to-lie	1.0	1.0	1.0	100
Lie-to-sit	0.98	0.95	0.97	99.80
Stand-to-lie	0.99	0.99	0.99	99.90
Lie-to-stand	0.96	0.99	0.98	99.80
Overall accuracy				98.35

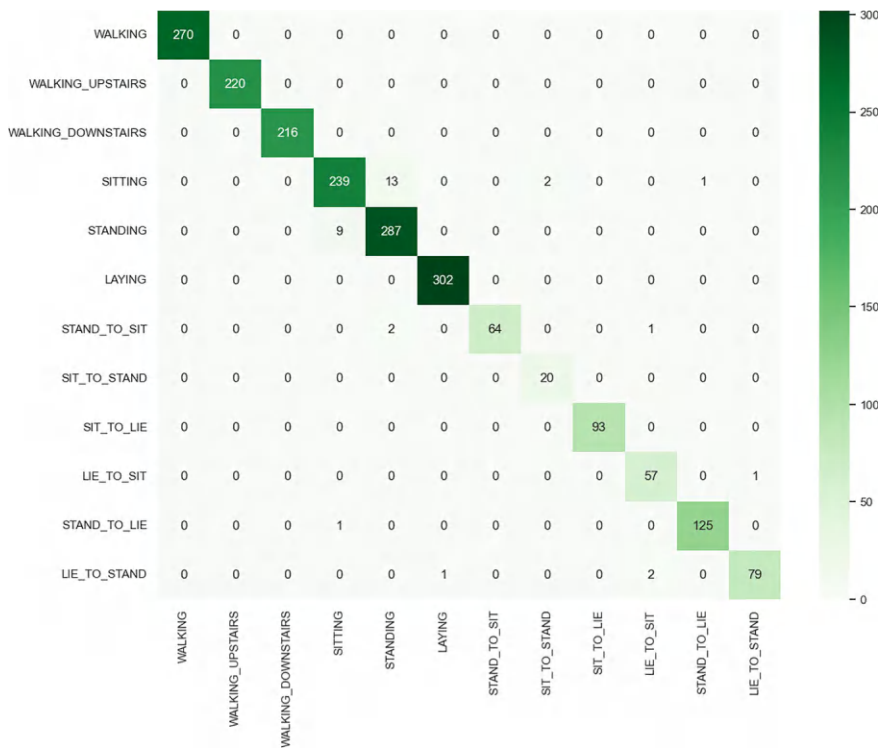


Fig. 4 Confusion matrix for each human activity using HAPT database

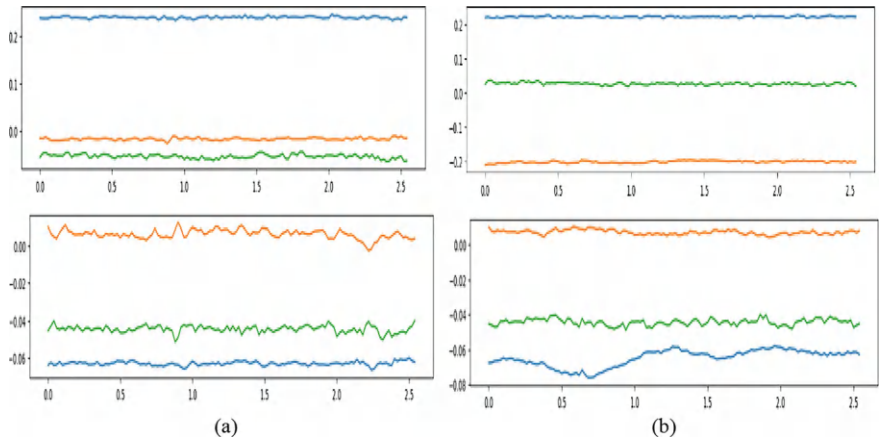


Fig. 5 Patterns of accelerometer signal (above) and gyroscope signal (down) of basic static activities, i.e., **a** sitting data sample and **b** standing data sample

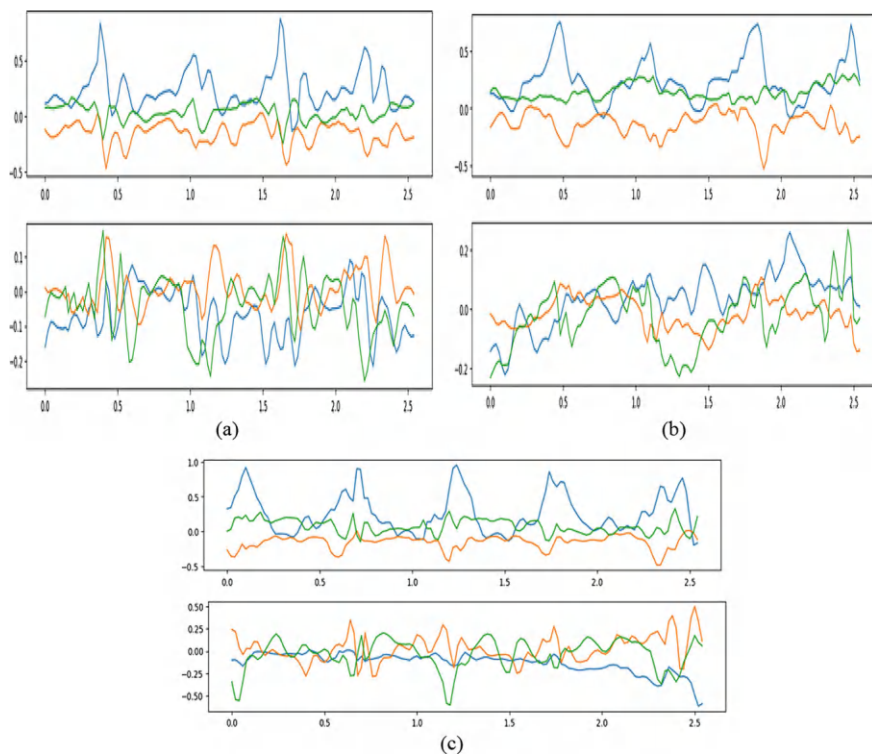


Fig. 6 Patterns of accelerometer signal (above) and gyroscope signal (down) of basic dynamic activities, i.e., **a** walking data sample, **b** walking upstairs data sample and **c** walking downstairs data sample

Vector Machine and Naïve Bayes are used as weak learners, and the supervised method, i.e., Linear Discriminate Analysis, is used as the Meta-model. This technique offers minimal computational cost since it employs stacked learning. Furthermore, the proposed HAR system with Static Postural Transition-based Post-Processing, denoted as SPTPP, proposed by Ahmed et al., also shows promising performance with an accuracy score of 97.05% [16]. The capability of the technique in analysing the presence of postural transition from the previous activity history to the current activity helps improve the detection accuracy of the posture change.

On the other hand, from the empirical results, our proposed MTCM shows superior performance to the other models with an accuracy score of 98.38% and an F1 score of 0.9808. The empirical results indicate the feasibility of MTCM dealing with static human activities as well as postural transitions. Specifically, the embedded multi-head temporal convolution is able to execute distinctive feature learning independently according to the different types of input signals captured by different sensors. This allows a custom-tailored learning architecture for each type of sensor,

Table 3 Performance comparison with the existing models on HAPT

Method	Accuracy (%)	F1
Support Vector Machine* [1]	84.49	–
LSTM* [1]	93.24	–
DWT+BiLSTM* [1]	94.87	–
Convolutional Neural Network* [2]	91.98	0.9302
Stacking Ensemble Learning* [3]	97.49	0.72
Conv-LSTM* [4]	91.6	–
Static Postural Transition-based Post-Processing (SPTPP)* [5]	97.05	–
Pre-trained DenseNet121, 1st 308 layers frozen* [6]	92.44	0.9252
The proposed MTCM	98.35	0.9808

* The result is extracted from the cited paper(s)

i.e., accelerometer and gyroscope sensors. Such learning is a promising facet for characterizing the multichannel time series inertial data.

6 Conclusions

A multi-head temporal convolutional analysis framework was presented in this paper for effective recognition of human basic activities as well as postural transitional activities. The architecture learns data features directly from the captured inertial signals without the necessity for manual feature engineering. After data segmentation using the fixed sliding window technique, the segmented data was processed via multi-head temporal convolutional analysis. In this model, an independent custom-tailored learning architecture for each accelerometer sensor and gyroscope sensor, respectively, is developed. This allows more precise data characterization of the multichannel inertial data. The empirical results demonstrated a promising recognition performance of the proposed model on the UCI HAPT database with a high accuracy score.

Acknowledgements This research is funded by the Ministry of Higher Education (MOHE) under Fundamental Research Grant Scheme (FRGS) with the grant number FRGS/1/2020/ICT02/MMU/02/7.

References

1. Ahad MAR, Antar AD, Ahmed M (2021) IoT sensor-based activity recognition, vol 173. Springer

2. Nguyen B, Coelho Y, Bastos T, Krishnan S (2021) Trends in human activity recognition with focus on machine learning and power requirements. *Mach Learn Appl* 5:100072
3. Friday Nweke H, Ying Wah T, Alo U (2018) Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: state of the art and research challenges. *Expert Syst Appl* 105:233–261
4. Atrsaei A, Dadashi F, Hansen C et al (2020) Postural transitions detection and characterization in healthy and patient populations using a single waist sensor. *J Neuroeng Rehabil* 17(1):1–14
5. Mohd Noor MH, Tan SY, Ab Wahab MN (2022) Deep temporal Conv-LSTM for activity recognition. *Neural Process Lett* 54(5):4027–4049
6. Gonçalves R, Pereira FL, Ribeiro VM, Rocha AP (2021) Roll padding and WaveNet for multivariate time series in human activity recognition. *Adv Intell Syst Comput (AIST)* 1365:238–248
7. Yong SC, Pang YH, Yin OS, Quan Wei P (2022) Stacking spatial-temporal deep learning on inertial data for human activity recognition. *J Eng Sci Technol* 17(5):3235–3253
8. Guan Y, Ploetz T (Mar 2017) Ensembles of deep LSTM learners for activity recognition using wearables. In: *Proceedings of the ACM interactive, mobile, wearable ubiquitous technologies*, vol 1, no 2, pp 1–28
9. Ahmed N, Rafiq JI, Islam MR (2020) Enhanced human activity recognition based on smartphone sensor data using hybrid feature selection model. *Sensors* 20(1):317
10. Tran DN, Phan DD (Jul 2016) Human activities recognition in android smartphone using support vector machine. In: *Proceedings—international conference on intelligent systems, modelling and simulation, ISMS*, pp 64–68
11. Zhang X, Wong Y, Kankanhalli MS, Geng W (2019) Hierarchical multi-view aggregation network for sensor-based human activity recognition. *PLoS ONE* 14(9):e0221390
12. Ramanujam E, Rasikannan L (2022) Coupling of dimensionality reduction and stacking ensemble learning for smartphone-based human activity recognition. *Int J E-Serv Mob Appl* 14(1)
13. Ronao CA, Cho S-B (2015) Evaluation of deep convolutional neural network architectures for human activity recognition with smartphone sensors. <http://www.iro.umontreal.ca/~bengioy/dlbook>. Accessed 15 Jul 2020
14. Cruciani F, Vafeiadis A, Nugent C et al (2020) Feature learning for human activity recognition using convolutional neural networks: a case study for inertial measurement unit and audio data. *CCF Trans Pervasive Comput Interact* 2(1):18–32
15. Thu NTH, Han DS (Feb 2020) Utilization of postural transitions in sensor-based human activity recognition. In: *2020 international conference on artificial intelligence in information and communication ICAIIC 2020*, pp 177–181
16. Ahmed M, Das Antar A, Ahad MAR (2021) Static postural transition-based technique and efficient feature extraction for sensor-based activity recognition. *Pattern Recognit Lett* 147:25–33
17. Pavliuk O, Mishchuk M, Strauss C (2023) Transfer learning approach for human activity recognition based on continuous wavelet transform. *Algorithms* 16(77)

Windows Registry Attack and Defense: Behavior Analysis and Persistence Detection Based on ETW



Jian-Xin Chen, Yang-Hsin Chang, Yan-Ling Hwang, Jian-Hong Hunag, Hao-Jyun Wang, and Fu-Hau Hsu

Abstract Windows Registry contains various information about users and can be viewed as a database. While Microsoft gives users customization choices, it also unintentionally turns into a tool that attackers may fully use to carry out fileless assaults, incursion persistence, and other harmful actions. The opening section of this paper introduces the registry's logical structure and internal data structures. The following section examines registry-related threats and how the current Windows 10 operating system responds to them. Finally, it shows how a system based on Windows' integrated logging system, ETW, has been implemented. The monitoring application records and filters registry write operation data, and the filtered data is then examined and sent to VirusTotal to ascertain if the written data was malicious. This study implements an ETW-based registry attack detection system that is effective and dependable. By integrating with VirusTotal, we can more quickly and correctly identify and stop registry assaults or misuse, protecting the security of user PCs.

Keywords Windows registry · Attack · Defense · ETW · VirusTotal

1 Introduction

Due to the increasing use of computers for data processing and exchange in cyberspace and information technology, computer viruses spread more readily when computers interact. According to data from CERT/CC, viruses and Trojan horses are the most common attack vectors against computers and pose the most significant security risk to users. They typically spread across networks and drives—the risk and unpredictability of communication increase when Trojans and viruses are present.

J.-X. Chen · Y.-H. Chang · J.-H. Hunag · H.-J. Wang · F.-H. Hsu (✉)

Department of Computer Science and Information Engineering, National Central University,
Taoyuan 32001, Taiwan
e-mail: hsufh@csie.ncu.edu.tw

Y.-L. Hwang

School of Applied Foreign Languages, Chung Shan Medical University, Taichung 40201, Taiwan

The most widely used operating system is Windows, which has over 1.5 billion users globally. Therefore, it is still crucial to create new malware detection methods for Windows systems. Security flaws must still exist in order to prevent malware, even if Windows is not likely to be the most used operating system in a few years.

A computer virus is an autonomously replicating piece of malicious software. Viruses can obstruct a computer's operation, damage hardware and software, or steal data. They typically reproduce by duplicating themselves. Malware may spread or be present when computers communicate if precautions are not in place. Techniques for proactively detecting malware now in use include eigenvalue scanning, signature matching, and proactive protection. However, each of these strategies has flaws of its own. For instance, eigenvalue scanning may be fooled by malware with new signatures, proactive protection might have a high false alert rate, and signature matching is only effective against known infections.

Computer attackers utilize various techniques to attack or weaponize computers as their use in daily life increases [1–3]. With each new operating system release or update, computers grow more secure, but there are several methods that hackers may use to get around these security measures. Malware most frequently modifies its behavior and source code on each infected machine to get around security measures. Malware analysis is the collective term for all techniques analysts employ to find malware. The phrase “malware analysis” is broad and includes numerous phases. These steps entail inspecting the suspicious program's contents without executing it, running it in a controlled environment, looking at DNS resolution requests, studying registry reads and writes, looking at file accesses, and examining Application Programming Interface (API) calls. This virus serves a specific function. We know they are used for various things, including hindering system performance, gaining illegal network access, and gathering private data.

Numerous platforms have been attacked, including servers, laptops, mobile phones, and cameras. Today, there are an increasing number of platforms being targeted. The virus created for various systems is very different as a result. People invest much time and resources to defend themselves from this program. Many of the anti-malware programs developed nowadays are both commercial and academic. The parties attempting to identify the virus and the attackers who produce it are in solid rivalry. Both sides have more strength and power as a result of this circumstance. The outcome of this conflict is degenerate malware. As we can tell from its evolution, malware has undergone a flawless structural shift. However, one aspect remains the same across all phases. These qualities are intended to have adverse effects. They act maliciously, in other terms. All malware should take some action to complete its mission. The virus must use particular operating system functions if it runs on a machine with the Windows operating system.

The Windows Registry, which houses many systems and application configuration data, is a crucial part of the Windows operating system. A vital element of the Windows system that houses a wealth of design and application configuration data is the Windows registry. Nevertheless, it is a target of malware assaults. Malware often either reads or writes to the registry file. Reading the registry file indicates that the virus reads any available information, including sensitive information. The stolen

secrets may be inserted into the registry file by slyer malware, which will then upload it to the C2 server it controls and attempt to erase all traces of the written data. The virus may write the path to the malware's executable file into the startup key, which will cause it to run automatically when the machine boots up. Alternatively, it may write the code into the key, where the compiler command will execute it. Some keys, like the task manager or the built-in Windows firewall, can change their values by editing the registry file. With the help of the registry file, it may be disabled.

The excitement of malware for exploiting the registry file is demonstrated by the fact that up to 143 different forms of malware employ the registry file modification approach in tactical defense evasion, according to MITRE ATT&CK [4]. This study examines the attack and defensive mechanisms of registry files and the layout of the registry file in the hopes that it will be helpful to future researchers in related studies because there are currently few studies on registry files. In order to achieve fileless persistence, the attacker will place obfuscated shell code, payload, or a portion of the code in the registry file, according to MITRE ATT&CK, Technology ID: T1027.011, Fileless Storage [5]. This study will be based on the RegGuard system by ETW, demonstrating the viability of real-time registry file activity monitoring as a further measure of registry file security.

The remainder of the paper is divided into the following sections: Sect. 2, Historical Context for Windows and Register Editor Technologies. The taxonomy of ETW assaults is presented in Sect. 4 of the paper, which also describes the study methodology. The information analysis and experimentation findings will be transmitted to the registry file in Sect. 5. The article was then concluded and further discussed in Sect. 6.

2 Background

2.1 *Windows Register Editors*

There might be a misunderstanding or confusion regarding the term “Windows register editors” in your question. It may refer to the Windows Registry Editor and its organization, but it needs to clarify the three logical structures. The Windows Registry Editor typically deals with keys and values, which are the primary components of the Windows Registry. The Windows Registry is organized into hives, keys, and values. Hives are the top-level containers, and keys are subfolders that group related settings and values to store the actual configuration data. There may be various logical structures within the Windows Registry based on how it is organized for different purposes. However, it is not typically described as having three specific logical structures. If it has a more specific question or if there is a particular aspect it would like to know more about, please provide additional details. Let us explain the Windows Registry and its key components. The built-in Windows utility Registry Editor (regedit.exe),

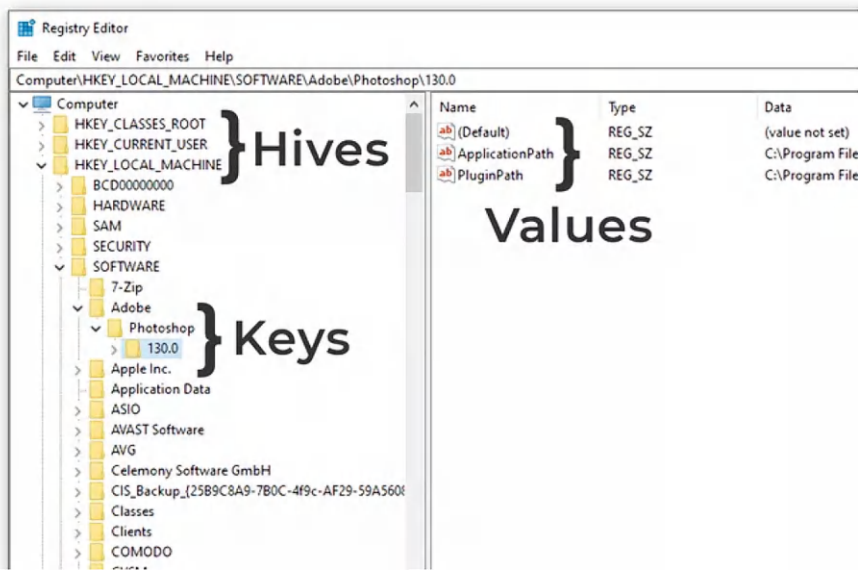


Fig. 1 Windows of the register editor

which is seen in Fig. 1, offers users an efficient method to inspect the registry files in the system through the GUI and to carry out simple actions.

In the Hives, the Windows Registry is a hierarchical database that stores configuration settings and options on Microsoft Windows operating systems. It is organized into several “hives,” like top-level containers or folders for different categories of system and user settings. A hive represents a logical structure made up of keys, subkeys, and values. The necessary support files are loaded into memory when the system boots or a user sign on. For instance, the user profile hive keeps track of user-related data, including program preferences, desktop environment, and network connection. HKEY_USERS is the key that corresponds. The five main hives are as shown in Fig. 2. In the HKEY_CLASSES_ROOT (HKCR) is that the hive contains information about file associations, OLE (Object Linking and Embedding) objects, and other settings related to file types. In the HKEY_CURRENT_USER (HKCU), the hive stores configuration data for the currently logged-in user. Each user account has its own set of settings in this hive. In the HKEY_LOCAL_MACHINE (HKLM), the hive contains configuration data for the local computer. It includes settings that apply to all users and the system itself. In the HKEY_USERS (HKU), the hive contains user profiles and their associated settings for all user accounts on the computer. In the HKEY_CURRENT_CONFIG (HKCC), the hive contains information about the current hardware configuration of the computer.

In the Keys, each hive contains a set of keys, like subfolders, that organize related settings. Keys can have child keys and values. For example, it might find keys related to hardware, software, and system settings in the HKLM hive.

Registry hive	Supporting files
HKEY_CURRENT_CONFIG	System, System.alt, System.log, System.sav
HKEY_CURRENT_USER	Ntuser.dat, Ntuser.dat.log
HKEY_LOCAL_MACHINE\SAM	Sam, Sam.log, Sam.sav
HKEY_LOCAL_MACHINE\Security	Security, Security.log, Security.sav
HKEY_LOCAL_MACHINE\Software	Software, Software.log, Software.sav
HKEY_LOCAL_MACHINE\System	System, System.alt, System.log, System.sav
HKEY_USERS\DEFAULT	Default, Default.log, Default.sav

Fig. 2 Related files for Hive

In the Values, within each key, it can find values, which are entries that store specific configuration data. Values can be of various data types, such as strings, numbers, or binary. They are used to configure various aspects of the Windows operating system and installed software. The types REG_SZ, REG_EXPAND_SZ, and REG_MULTI_SZ, as these types of values can be written in the arbitrary length of the string. As a result, the author believes that the attacker is most likely to exploit these types of vulnerabilities. Value (Value), the type of Microsoft, also has a unique definition composed of REG_*, as shown in Fig. 3 [6]. In the REG_SZ is a null-terminated string that can be encoded using ANSI or Unicode. The REG_EXPAND_SZ is essentially the same as REG_SZ, except it can be supplied with an environment variable like %Path% that the system can interpret and replace in the string.

The REG_MULTI_SZ, which consists of many REG_SZs, it is possible to think of it as an array of the REG_SZ type.

2.2 Event Trace on Windows

Event Trace on Windows (ETW) is an important technology that enables real-time monitoring. Event Trace on Windows (ETW) is an efficient and lightweight logging mechanism built into Windows. ETW provides the infrastructure for events triggered by user-mode (applications) and core-mode (drivers) components using the buffering and logging mechanisms implemented in the operating system core. ETW can be used for system and application diagnostics, troubleshooting, and performance monitoring.

It also has the following beneficial features: With dynamic tracking, tracking sessions can be dynamically started, stopped, and reset without rebooting the machine. Built into Windows, there is no need to download additional tools. Lightweight and evolving over time, currently offers four tracking mechanisms to

Value	Type
REG_BINARY	Binary data in any form.
REG_DWORD	A 32-bit number.
REG_DWORD_LITTLE_ENDIAN	A 32-bit number in little-endian format . Windows is designed to run on little-endian computer architectures. Therefore, this value is defined as <code>REG_DWORD</code> in the Windows header files.
REG_DWORD_BIG_ENDIAN	A 32-bit number in big-endian format . Some UNIX systems support big-endian architectures.
REG_EXPAND_SZ	A null-terminated string that contains unexpanded references to environment variables, for example, <code>%PATH%</code> . It's either a Unicode or an ANSI string, depending on whether you use the Unicode or ANSI functions. To expand the environment variable references, use the ExpandEnvironmentStrings function.
REG_LINK	A null-terminated Unicode string that contains the target path of a symbolic link that was created by calling the RegCreateKeyEx function with <code>REG_OPTION_CREATE_LINK</code> .
REG_MULTI_SZ	A sequence of null-terminated strings, terminated by an empty string (<code>\0</code>). The following is an example: <code>String1\0String2\0String3\0LastString\0\0</code> . The first <code>\0</code> terminates the first string, the second-from-last <code>\0</code> terminates the last string, and the final <code>\0</code> terminates the sequence. Note that the final terminator must be factored into the length of the string.
REG_NONE	No defined value type.
REG_QWORD	A 64-bit number.
REG_QWORD_LITTLE_ENDIAN	A 64-bit number in little-endian format. Windows is designed to run on little-endian computer architectures. Therefore, this value is defined as <code>REG_QWORD</code> in the Windows header files.
REG_SZ	A null-terminated string. It's either a Unicode or an ANSI string, depending on whether you use the Unicode or ANSI functions.

Fig. 3 Microsoft’s comprehensive breakdown of each kind [6]

work with. Stable, providing an efficient buffering and logging mechanism. The tracking buffer is managed by the kernel.

3 Related Works

The Process Monitor in Procmon is a sophisticated Windows monitoring tool for recording and examining system processes and activities. Its comprehensive event recording and filtering features allow it to monitor actions, including file system activity, registry access, network connectivity, and process formation. However, gathering malicious login files necessitates a certain amount of labor and time commitment, and Procmon demands that researchers have pertinent knowledge of the registry. RegGuard offers low-latency, real-time monitoring while automatically securing machine code that attackers could exploit. Registry data and the Forensic Toolkit (FTK) Imager were used by Saidi et al. [7] to gather evidence and determine if keylogger malware was supplied to the target workstation by an attacker who infected the machine through VNC. Data carving, volatile memory extraction [8], and registry file extraction [9] are the three extraction techniques used by Chirath et al. [9] to get

Table 1 Comparison with other systems

	RegGuard	RAD
Data collection	ETW	Win32 hook
Filter	Yes	No
Prediction	VirusTotal	Probabilistic model
Protection	Delete related files/values	Only alert

the registry file information needed for forensic investigation. A host-based intrusion detection system (IDS) was suggested by Apap et al. The system intercepts a trip’s operation data to the registry file using the Win32 hook technique, uses these data features to train a probability model and its algorithm, and uses this information to determine whether the trip is normal or abnormal. If it is abnormal, the system then sends out an alarm. The comparison between RegGuard and RAD is shown in Table 1 below. Although the performance cost of this technique has yet to be discussed in the study, RAD uses the Win32 hook mechanism for data collecting to intercept system operation data on the registry file. RegGuard uses the ETW method, which is a way to intercept the operation data of the system on the registry file. In contrast, the system’s operation data on the registry file is so large that each API must be added with its program logic, resulting in an unavoidable performance loss for the entire system. RegGuard employs ETW, a Windows kernel-integrated logging technique with fast speed, minimal loss, and essentially little influence on the original system characteristics. RegGuard also offers a filter mechanism that can efficiently block irrelevant event data, cutting down on analysis time and computational costs. By leveraging the features of typical journeys, RAD employs machine learning to train a model to determine whether a trip is expected. This type of prediction will invariably lead to the incidence of missed judgments. RegGuard can have improved stability since it employs VirusTotal, which does a collaborative examination of many anti-virus programs and sandboxes to assess whether or not files are harmful.

4 System Design

This section primarily introduces the system’s fundamental workings, as shown in Fig. 4. Three parts make up the system in this article: ETW, the data analysis component, and ELK. We track the login file events of the subordinate computers using Windows’ built-in log monitoring mechanism. The login file is sent to the ETW consumer application program whenever any program on the machine operates on it (open, delete, create, write.). The ETW consumer application program filters out any trips or events we are not interested in and then sends the filtered data to the data analysis component for additional analysis. The data analysis component will receive the files transmitted by ETW. Through the VirusTotal API, the information will be transmitted to VirusTotal for analysis to determine if it is dangerous or suspicious.

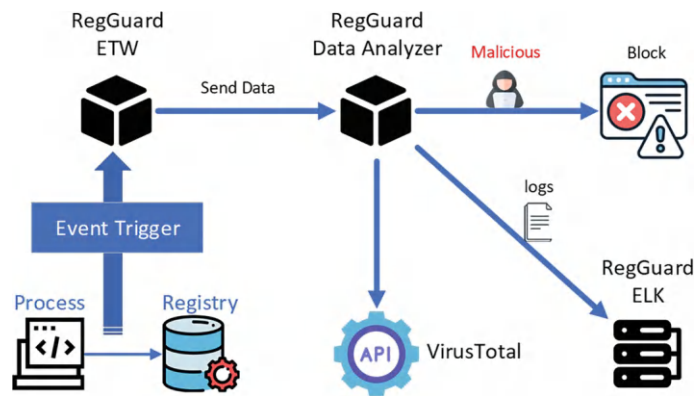


Fig. 4 Organogram for the system

If the outcome is malicious or suspicious, the application will remove whatever data it has written and send the log file to ELK. ElasticSearch, Logstash, and Kibana make up ELK. When it comes to forensic investigation and information security, this technology combination is common. Elastic is in charge of data storage and search; Logstash is in charge of data filtering, analysis, and transmission to Elastic; and Kibana offers a user-friendly UI interface. The login file events of the entire system can now be presented more visually, thanks to ELK, and users may filter the data based on their expertise to find additional, more valuable information.

5 Experimental Results and Analysis

The experimental setup for this part is as follows as shown in Table 2.

Since the way to distinguish malicious behavior relies on VirusTotal, this part is not the system’s focus, so the evaluation test is mainly based on whether it can successfully filter out unnecessary noise and listen to all events in a real-time environment without missing anything. This study chooses to open an additional thread in the ETW component to realize the concept of writing and listening to the evaluation so that the current PID can be filtered to listen only to the writing events of this

Table 2 Experimental environment

Environment	
Board	TUF GAMING X570-PLUS
OS	Windows 10 Pro
Version	22H2
CPU	AMD Ryzen 7 3700X
Memory	16 GB

Table 3 The kind and volume of information that will be sent to the registry file

Data of value	Quantity
A. 'a' * 31	70
B. powershell -executionpolicy bypass -windowstyle hidden -command "a = Get-ItemProperty -Path HKLM:\System\%a %{\$_.v}; powershell -executionpolicy bypass -windowstyle hidden -encodedcommand \$a"	56
C. powershell -executionpolicy bypass -windowstyle hidden -command "a = Get-ItemProperty -Path HKLM:\System\%b %{\$_.v}; powershell -executionpolicy bypass -windowstyle hidden -encodedcommand \$a"	74

program, not the same as the trip filtering rules in the real world as mentioned above. This study uses this additional thread to obtain the ground truth for evaluating the correctness of the system. This research attempted to replicate the high-frequency writing condition in the real world as closely as possible by writing the login file 200 times, 200 times every 3 s, over 10 min. Two codes and four values will be written at random. Table 3 has a list of the four numbers. The first aims to test whether the ML model can filter out these kinds of meaningless strings. The data length is set to 31 to pass the string length threshold. The second is a PowerShell command that retrieves the data from the registry code HKLM, Systema, and runs the command that corresponds to the data. The PowerShell command is the second.

6 Conclusion

In this study, we create RegGuard, a system that uses ETW (Event Tracing for Windows) technology to identify harmful registry activity instantly. RegGuard is a program that, in real-world tests, effectively stops adversaries from misusing the registry, removing it from the list of places where attackers can play. It does this by detecting malicious activity in the registry in real-time using the ETW (Event Tracing for Windows) technology. At the same time, the community now has a more excellent grasp of the register due to our study on registry threats and remedies.

References

1. Hensley B (2021) Identity is the new perimeter in the fight against supply chain attacks. *Netw Secur* 2021(7):7–9
2. Pitney AM, Penrod S, Foraker M, Bhunia S (2022) A systematic review of 2021 microsoft exchange data breach exploiting multiple vulnerabilities. In: 2022 7th international conference on smart and sustainable technologies (SpliTech). IEEE, pp 1–6
3. Mariani B (2011) Userland hooking in windows, vol 33. High-Tech Bridge SA
4. M. ATT&CK. <https://attack.mitre.org/techniques/T1112/>
5. M. ATT&CK. <https://attack.mitre.org/techniques/T1027/011/>
6. Microsoft. <https://learn.microsoft.com/zh-tw/windows/win32/sysinfo/registry-value-types>

7. Saidi RM, Ahmad SA, Noor NM, Yunus R (2013) Windows registry analysis for forensic investigation. In: 2013 the international conference on technological advances in electrical, electronics and computer engineering (TAECE)
8. Zhang S, Wang L, Zhang L (2011) Extracting windows registry information from physical memory. In: 2011 3rd international conference on computer research and development, Shanghai
9. Chirath WDA, Rupasinghe L (2019) Comprehensive forensic data extraction and representation system for windows registry. In: 2019 international conference on advancements in computing (ICAC)

Device Identification Technology Within the Network Address Translation Devices



Jia-Ning Luo  and Ying-Yi Yeh

Abstract Due to the rapid growth of the Internet, the last IPv4 address was exhausted by the end of 2019. Using Network Address Translation (NAT) technology is popular because it allows multiple network devices to share one or more IPv4 addresses without migrating to IPv6. However, when conducting black-box penetration testing outside of NAT networks, the original IP addresses have already been changed, which makes it hard to identify the source address of a device with potential threats. Therefore, it is necessary to detect whether a device with an IPv4 address uses NAT technology and further probe its internal network devices. This paper proposes a mechanism using deep packet inspection to detect the NAT box and the devices behind it. Penetration testers can use the method we propose to accurately identify potential Internet of Things (IoT) devices within the network, facilitating precise penetration testing.

Keywords Network address translation · Deep packet inspection · Internet of Things

1 Introduction

With the fast growth of the Internet, Internet of Things (IoT) has emerged as a significant trend [1]. According to Aruba, a subsidiary of Hewlett Packard Enterprise, it predicts that each user will have ten internet-connected devices by 2025 [4]. Additionally, the United Nations estimates that the world's population will reach 8 billion on November 15, 2022 [5]. With the exhaustion of IPv4 addresses and the limited

Support by the National Science and Technology Council, Taiwan, under grant number NSTC 113-2634-F-004-001-MBK and NSTC 113-22231-E-606-012.

J.-N. Luo (✉) · Y.-Y. Yeh

Department of Computer Science and Information Engineering, National Defense University, Taoyuan, Taiwan

e-mail: deer@ccit.ndu.edu.tw

adoption of IPv6, coupled with the extensive use of IoT devices, the deployment of NAT devices has become commonplace. NAT's inherent feature of hiding internal devices has made it a simple firewall substitute.

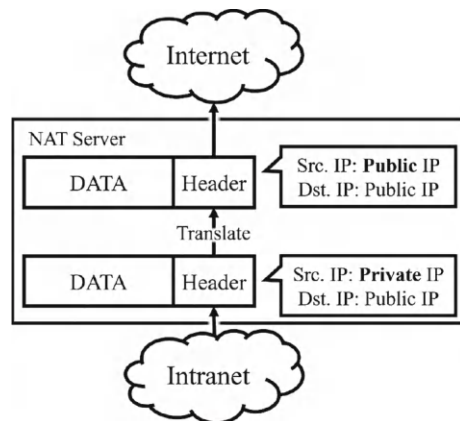
Because IoT devices pose security concerns and are often used in conjunction with NAT technology, the volume of attacks targeting IoT devices is expected to increase as their numbers grow. Therefore, it is crucial to identify specific IoT devices within the NAT environment. Given the vast number of IoT devices, it may be impossible to have complete control over them during deployment or replacement. Remaining vulnerable or outdated IoT devices could potentially serve as security vulnerabilities, posing threats to overall network security.

When network packets are transmitted in an encrypted manner, only the headers remain for identification. However, when network packets pass through NAT devices, the Private IP used for device identification within NAT's internal network is translated into a Public IP. As Public IPs are now exhausted, most NAT devices map a multitude of Private IPs to a single Public IP. When capturing packets from outside NAT, all data streams appear to come from the same source-translated Public IP (see Fig. 1). In NAT's internal network, if Private IPs use a Class A configuration, they can use all IPs between 10.0.0.0/8 and 10.255.255.255/8, accommodating over 16.7 million devices. If the number of network devices within NAT is significant, and all data streams are mapped to the same source, identifying the actual source of data streams becomes a very challenging problem.

2 System Model

In recent years, most of the research has focused on identifying the type of operating system, determining whether it's behind NAT, or whether it's an IoT device. However, to identify potential IoT devices, it's not just about recognizing if it's an

Fig. 1 Address translation mechanism



IoT device; it's also crucial to accurately determine the number of IoT devices and further distinguish devices of the same type or model.

Passive network traffic analysis methods can be executed without the need to adjust network architecture or deploy specific software. This approach is similar to black-box penetration testing and offers lower deployment costs and easier scalability compared to methods involving network architecture adjustments, additional network equipment, or the installation of specific programs. Therefore, this paper employs passive analysis of NAT network data flows, combined with deep packet inspection technology. It records packets simultaneously from both the internal and external networks of NAT. After identifying the device type in the external network, it cross-references with devices in the internal network to validate the accuracy of the identification.

Given the large number of IoT devices and their high homogeneity in operating systems, identifying potential IoT devices during black-box penetration testing is challenging. This paper takes into account methods proposed in recent years and conducts device identification in a two-stage manner. In the first stage, it uses operating system characteristics, DNS requests, and IP header information to identify specific IoT devices. In the second stage, it further distinguishes devices of the same model. If an attacker installs malicious devices of the same model within the NAT network, this method can be used to identify them.

IoT devices are used differently in various fields, but most of them are sensors used to collect sensor data such as pressure or temperature and smart home appliances in home environments. The purpose of sensor deployment is to gather various sensor readings, while home smart appliances are used for remote control or monitoring, such as remotely unlocking electronic locks, real-time monitoring of home security cameras, or controlling the on/off status of home appliances.

Typically, different types of IoT devices connect to different servers. This is why, in recent research, destination IP addresses are often considered important features. However, server IP addresses may change, making destination IP addresses less suitable as recognition features for widespread deployment. Therefore, it's recommended to combine domain names and DNS query results for identifying device types and to use other connection features for further distinguishing devices of the same type. This approach is essential to achieve the goal of visualizing NAT networks.

3 Device Identification

To identify the type of device, it is essential to find unique features. In recent years, many studies have used operating system features, such as a TTL value of 128 indicating a Windows operating system, or extracting user information from HTTP headers. However, Windows TTL values can be manually adjusted, and various systems or applications have increasingly defaulted to using encrypted connections. Using TTL values as the sole feature for identifying a Windows operating system can be deceived, and methods like extracting user information from HTTP headers

are no longer applicable in today's network environments. Therefore, to effectively identify device types, it is necessary to focus on parts of the current network packets that are not affected by encryption mechanisms.

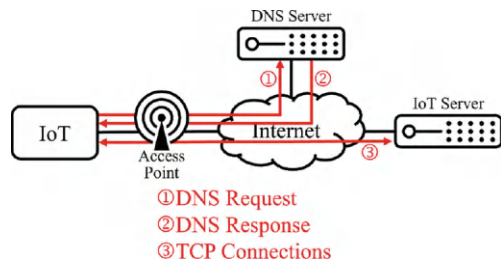
IoT devices frequently connect to specific servers for data collections and receiving commands. Therefore, most IoT devices use DNS queries to find the destination server IP address. In the current DNS service mechanisms, there have been emerging encryption mechanisms like DNS over HTTPS (DoH) and DNS over TLS (DoT) [2]. However, their adoption rates are still low, making them suitable for device type identification work in today's network environments.

In today's network environments where NAT devices are commonly deployed, users and IoT devices are typically in different NAT networks. To establish connections quickly, the fastest way is to deploy servers (or server clusters) with public IPs on the internet. Users and IoT devices actively connect to these servers to request services or transmit data. To retain flexibility for server adjustments or expansion, IoT device connection designs often rely on domain names as the basis for connections. This involves sending specific DNS requests and obtaining the desired server's IP address through DNS servers before establishing a connection with the server (see Fig. 2).

As different types of IoT devices need to establish connections with different servers, they generate distinct DNS requests before establishing connections. To identify specific IoT devices, you can analyze the domain names in the DNS requests. For example, if a domain name contains content like "apple.com," it can be determined that the DNS request was sent by a device produced by Apple. However, NAT services often have DNS proxy functionality. Once a NAT device obtains the DNS query result, it will not perform another DNS query until the TTL (Time To Live) of the result expires. Instead, it will immediately respond to DNS requests from the internal network with the previous query's result. This means that DNS requests captured outside the NAT will be fewer than those inside.

To avoid being limited by long waiting times and to expand the applicability of the method, this research proposes an approach of actively initiating DNS requests. Based on the results of DNS queries, the method filters data flows generated with specific destination IPs and attempts to capture DNS requests. By simultaneously detecting both DNS request domain names and specific destination IPs, this approach enhances the usability of the identification method.

Fig. 2 The IoT device connection flow



3.1 Identify the Same Device

For devices of the same type or model, identifying the source of data flows is quite challenging because they often share identical characteristics such as the operating system, communication protocols, and destination IP addresses. If these devices use encrypted transmission, the identification becomes even more difficult.

As network infrastructure improves, IoT and other network devices increasingly adopt TCP (Transmission Control Protocol) for enhanced reliability. TCP requires a three-way handshake before initiating transmission. During the connection setup, the client (i.e., the NAT side) generates a random source port number. Since this source port number is generated randomly (or pseudorandomly), the probability of generating duplicates is extremely low. Additionally, due to the large number of IoT devices being produced and deployed, there are typically multiple IP addresses associated with the DNS query results, often with load balancing mechanisms in place. Therefore, after identifying a specific type of device, it is possible to use a combination of source port number (referred to as source port) and destination IP as two distinguishing features to identify the source of data flows under the same communication protocol. The quantity of combinations of source ports and destination IPs can also be used to estimate the number of devices of the same model.

3.2 Proposed Model

This paper proposes a device identification process based on the research methods described above. Since IoT devices typically perform a DNS query to confirm the server's IP address before establishing a connection, it is essential to execute a DNS query to obtain the server IP before beginning the device identification process. However, in cases where the packet capture duration is insufficient, it may not be possible to capture DNS query packets. Additionally, the mapping between domain names and IP addresses may change due to manufacturer adjustments. Therefore, prior to identification, it is necessary to perform a DNS query to obtain the server's IP address.

IoT devices maintain continuous connections with servers, and with knowledge of the domain names used by IoT devices, it is possible to verify during identification whether the captured DNS Request contains specific domain names. Furthermore, one can check if the server's IP address, established through the DNS query based on the domain name, matches the IP obtained through the connection. This confirmation helps ascertain the presence of the desired IoT device. Subsequently, by combining the "source port number" and "destination IP" as two distinguishing features, data flow sources and device quantities can be identified.

Penetration testers can use the identification process proposed in this paper to calculate the number of specific devices. If the calculated device count does not match the known quantity, it may indicate the deployment of malicious devices, requiring

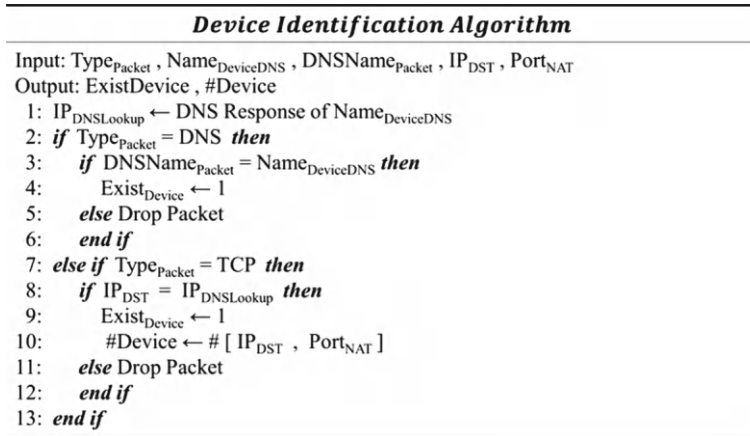


Fig. 3 The IoT identification algorithm

prompt investigation. This paper also presents an algorithm for device identification (see Fig. 3). During the identification process, it is necessary to extract five key features: Packet Type ($Type_{Packet}$), Domain Name of the Device's Connection Server ($Name_{DeviceDNS}$), Name in DNS Request ($DNSName_{Packet}$), Server IP (IP_{DST}), and Source Port Number ($Port_{NAT}$). After confirming the presence of the desired IoT device by matching the name in the DNS Request with the server IP, device quantities can be calculated using the combination of "destination IP" and "source port number" as features. Subsequently, if one wishes to develop software or utilize this method for machine learning, these features can be employed.

4 Experimental Design and Results

To capture network packets for analysis, this study used a Raspberry Pi (RPI) 3B+ model running Raspberry Pi OS configured as a wireless router with NAT functionality enabled. The RPI utilized its built-in wired network card for external connections (WAN) and its built-in wireless network card for internal connections (WLAN).

To facilitate the comparison of network packets before and after NAT translation, this research simultaneously captured network packets on both the RPI's wired and wireless network interfaces, saving them as Pcap files. This allowed for the external recognition of device types and subsequent comparison with the captured results from within the NAT, validating the accuracy of the recognition.

Since this study employed a two-stage data identification method, two separate experiments were conducted to confirm the feasibility of the experimental approach: one for "identifying device types" and the other for "identifying devices of the same type."

4.1 Identifying Device Types

The primary objective of the first experiment was to verify the feasibility of identifying IoT device types. To achieve this, five different devices were selected for the experiment: Xiaomi Desk Lamp, iPhone 13, Xiaomi Robotic Vacuum Cleaner, a laptop (running Windows OS), and Xiaomi Mi Box S. Each of these devices was connected to the internet via wireless networks through the RPi, and network packets were captured according to the previously described method (as illustrated in Fig. 4).

The experiment spanned 15 days, resulting in a captured file size of 20 GB. The goal was to assess whether, after undergoing network address translation (NAT), it was possible to identify the presence of various types of devices, including straight-forward IoT devices (the desk lamp and robotic vacuum cleaner), relatively complex IoT devices (the Android TV set-top box), a smartphone (iPhone 13 running iOS), and a laptop.

4.2 To Recognize Similar Devices

The purpose of the second experiment is to attempt to distinguish between two identical models and configuration IoT devices based on the first experiment. Therefore, two Xiaomi desk lamps and two Xiaomi Mi Box S devices of the same model were selected. The same iPhone 13 used in the first experiment was used, along with the

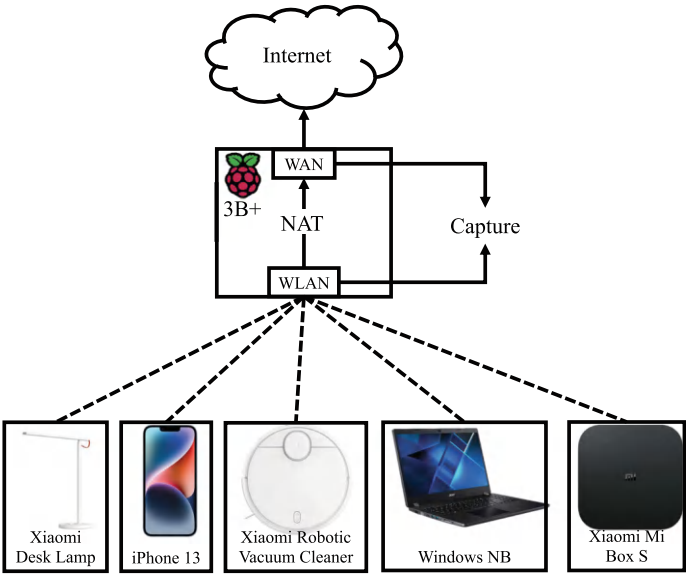


Fig. 4 The experimental framework

application settings and control for the Xiaomi desk lamps available on the Apple App Store. Once the iPhone 13 was set up, it was no longer connected to the Raspberry Pi box. The Xiaomi Mi Box S was set up in a manner similar to a regular television set-top box, used for streaming video content on a daily basis, and configured to connect to the internet via a wireless network through the Raspberry Pi box (see Fig. 4).

Packet capture was performed in the same manner as the first experiment, with a total capture time of 15 days resulting in a file size of 7 GB. The devices selected for this experiment included the simple Xiaomi desk lamp and the relatively complex Xiaomi Mi Box S. The goal was to identify the two Xiaomi desk lamps and two Xiaomi Mi Box S devices within the translated data streams over the network addresses.”

4.3 Experimental Results

In network packets before passing through NAT, apart from network card addresses and IPs, there are also Multicast Domain Name System (MDNS), Internet Group Management Protocol (IGMP), and numerous DNS requests that can be used for device identification. After passing through NAT, network card addresses and IPs have been translated. MDNS and IGMP packets are only broadcast within the local network. Additionally, due to the DNS proxy functionality provided by NAT devices, DNS requests are no longer re-queried within their time-to-live period, resulting in a decrease in the number of DNS requests after passing through NAT devices. This outcome demonstrates that the capture or non-capture of DNS requests should not be the sole criterion for determining the presence of a device.

4.4 Identifying Device Types

In the first experiment, devices like the Xiaomi Mi Box S, which are relatively complex IoT devices, were included. These IoT devices don't simply connect to a single server. Before each connection is established, they send corresponding DNS requests. Moreover, their connection behavior varies depending on user actions, resulting in different connection patterns. The majority of DNS requests generated by these user-driven behaviors could potentially be used for device identification. However, the timing of these characteristics is unpredictable and not easily applicable to various research scenarios.

In this experiment, DNS domain names contained in DNS requests were captured using the Tshark tool. The types and quantities of DNS requests captured from within and outside of NAT were tabulated (see Table 1). The results demonstrate that the DNS proxy functionality provided by NAT services reduces the number of DNS requests captured from the external network. While the decrease in the proportion

Table 1 The captured DNS queries

Device	DNS request	LAN-count	WAN-count
Xiaomi Desk Lamp	sg.ot.io.mi.com	5	5
Xiaomi Robotic Vacuum Cleaner	alsgp0.fds.api.xiaomi.com	985	53
Xiaomi Mi Box S	us.tv.global.mi.com	240	203
	Related DNS query such as	120	95
	animation.tv.global.mi.com		
	Unrelated DNS queries such as	2,967	2,931
	youtubei.googleapis.com		

is not significant, this outcome underscores that the capture or non-capture of DNS requests should not be the sole criterion for device identification. The extent of this decrease is not analyzed in this experiment as it involves factors such as IoT device behavior patterns, DNS query frequencies, TTL, etc., which are beyond the scope of this study.

Regarding the comparison of DNS request behavior, Xiaomi Smart Desk Lamp and Xiaomi Roborock Robot Vacuum generate simple DNS requests, each producing only one type. They also establish stable connections with the IP results of their DNS queries. In contrast, Xiaomi Mi Box S generates a wide variety of DNS requests, with up to 81 different types. This diversity is due to the multitude of application choices available to users when watching TV, such as selecting apps like YouTube or Netflix. Each execution of an application can generate more than one DNS request and connection. Additionally, the Xiaomi Mi Box S runs on the Android TV operating system, which generates data flows for Google account authentication and connections to Google servers during its operation, further complicating the types and modes of connections.

To enhance the reliability of device identification methods, this experiment chose “us.tv.global.mi.com” as a distinguishing feature for Xiaomi Mi Box S due to its higher frequency of DNS query behavior and the generation of stable data flows during the process, making it effective for identifying the presence of the device.

Each IoT device serves a different purpose, and in cases where DNS requests are not generated solely based on behavior patterns, they can be identified through DNS requests or destination IP addresses. This study filtered out the DNS request names and corresponding destination IPs generated by the Xiaomi Smart Desk Lamp, Xiaomi Roborock Robot Vacuum, and Xiaomi Mi Box S when connecting to the manufacturer’s servers, IPs may vary due to manufacturer device adjustments). By detecting these DNS requests and destination IPs, it is possible to confirm the presence of the corresponding devices. However, due to the wide variety of IoT devices, obtaining a sufficient variety and quantity of IoT devices in the future will allow for the creation of an extensive feature database, expanding the number of identifiable device types.

4.5 To Recognize Similar Devices

Building on the results of the first experiment, this study continued to use DNS request or destination IP features for device type identification. However, it was found that devices of the same model with the same settings generated different DNS requests.

Using the Tshark [3] tool, this experiment also captured and tabulated the types and quantities of DNS requests captured from within and outside the NAT. For the Xiaomi Smart Desk Lamp, in addition to “sg.ot.io.mi.com” generated in the first experiment, the second experiment added “sg.ots.io.mi.com,” and both of these DNS requests were generated by both Xiaomi Smart Desk Lamps. For the Xiaomi Mi Box S, apart from “us.tv.global.mi.com” generated in the first experiment, the second experiment added “tw.tv.global.mi.com.” Unlike the Xiaomi Smart Desk Lamp, the two Xiaomi Mi Box S devices did not generate the same DNS requests. Furthermore, the number of DNS requests captured from the external network also showed a decreasing trend in this experiment. This result once again confirms that capturing or not capturing DNS requests should not be the sole criterion for device identification.

In addition to DNS domain name statistics, this experiment also consolidated the DNS query results. It was confirmed in this experiment that the speculation regarding possible changes in server IP addresses was valid. In the DNS requests generated by the Xiaomi Smart Desk Lamp, the domain name “sg.ots.io.mi.com” fig received different query results at different times. Besides having different corresponding IP addresses, the quantity also increased from 2 to 4. This result validates the notion that destination IP should not be used as a fixed feature for identification.

To validate the feasibility of the method proposed in this experiment, source port numbers and destination IPs were also captured and compared to the connection behavior within and outside the NAT. It was observed that the data flows from the Xiaomi Smart Desk Lamp had a longer duration. Over the course of a 15-day experiment, the two Xiaomi Smart Desk Lamps generated only a total of 5 data flows, while the two Xiaomi Mi Box S devices generated a total of 572 data flows. All of these data flows could be successfully identified from the external network using a combination of “destination IP” and “source port number” as features, achieving a 100% recognition accuracy (see Fig. 5). However, in this experiment, due to the limited availability of the same model devices, only preliminary results could be obtained. Whether the same research method can be applied effectively in a situation with a large number of identical devices remains to be validated once a sufficient number of devices are acquired for further experimentation.

Device	# of Data streams		identifiable characteristics		identifiable data stream	Accuracy
	LAN	WAN	Destination IP	Source Port		
Xiaomi Desk Lamp #1	3	5	161.117.198.4	49385	3	100 %
			47.241.121.134	50082		
			8.214.10.144	50078		
Xiaomi Desk Lamp #2	2		161.117.194.68	62012	2	100 %
			24.241.49.4	62016		
Xiaomi Box S #1	233	572 st	47.74.175.105	43192 and 66 others	233	100 %
			47.74.172.19	34538 and 167 others		
Xiaomi Box S #2	338		47.74.175.105	38740 and 304 others	338	100 %
			47.74.172.19	42802 and 34 others		
Note	Only count the data streams relative to us.tv.global.mi.com and tw.tv.global.mi.com					

Fig. 5 Number of DNS requests generated by the second experiment

5 Conclusion

With the growing awareness of cybersecurity, there is increasing emphasis on black-box penetration testing. The method proposed in this paper involves passive network data flow analysis and can be executed without the need to modify the network architecture. It can be complemented with deep packet inspection (DPI) techniques, making it effective even in today's world of widespread encrypted transmissions.

This paper presents an algorithm for querying devices within a NAT and adopts a method that first distinguishes device types and then identifies devices of the same type or model. It achieves this by using DNS requests and destination IPs to differentiate between device types and then using source port numbers and destination IPs to differentiate between devices of the same type or model. However, the applicability of this method in situations with a large number of devices of the same type or model remains to be verified due to the limited availability of such devices in the current study.

Due to the vibrant development of IoT-related technologies and the proliferation of IoT devices, the field of device identification currently lacks refined methods for distinguishing between devices of the same type or model. While there has been

research focused on detecting IoT devices or classifying them using machine learning algorithms, there is still room for improvement in methods for identifying devices of the same type or model.

Furthermore, machine learning techniques have rapidly advanced, and when combined with the development of deep packet inspection technologies, they hold great promise for the future. With access to large feature databases, tools like nDPI [6] can be used to quickly analyze network data flows, and machine learning techniques can be employed to automatically identify potential IoT devices. This will enhance the precision and efficiency of black-box penetration testing.

References

1. Atzori L, Iera A, Morabito G (2010) The internet of things: a survey. *Comput Netw* 54(15):2787–2805. <https://doi.org/10.1016/j.comnet.2010.05.010>. <https://www.sciencedirect.com/science/article/pii/S1389128610001568>
2. Badhwar R (2021) Domain name system (dns) security. In: *The CISO's next frontier: AI, post-quantum cryptography and advanced security paradigms*. Springer, pp 207–212
3. Brokker.net (2018) Inspecting http headers with tshark. <https://brokkr.net/2018/09/02/inspecting-http-headers-wi>
4. Hughes D (2022) 5 networking predictions for 2022. <https://www.arubanetworks.com/resource/5-networking-predictions-for-2022/>
5. United Nations: World population prospects 2022: summary of results (2022). <https://www.un.org/development/desa/pd/content/World-Population-Prospect-2022>
6. Zalewski M (2023) ndpi - open and extensible lgplv3 deep packet inspection library. <https://www.ntop.org>

Fuzz Testing and Analysis

Prediction of Processing Parameters Based on Multi-objective Bayesian Optimization Combined with Machine Learning Model for High Frequency Heat Treatment of Metals



Chih-Hsueh Lin, Yu-Hsiang Tseng, Kuo-Hsin Hu, and Chia-Wei Ho

Abstract In the process of metal processing, heat treatment is a common metal processing method, which is usually used to change the mechanical properties of metal alloys to control such as hardness, strength, toughness, ductility, and elasticity. The high-frequency heat treatment is to rapidly heat the surface of the steel with high frequency, stop the heating when it reaches the quenching temperature, and use an appropriate coolant to implement quenching. In the process of processing, it is often necessary to set processing parameters for the processed products, so as to ensure that each processed product meets the requirements. This manual setting method is not only inefficient but also needs to rely on the experience of field personnel to set parameters. Even so, the problem of artificially setting wrong parameters still occurs. Therefore, this paper applies machine learning technology to analyze the correlation between processing parameters and establishes a predictive evaluation model in combination with hyperparameter optimization, and then predicts the best processing parameters. In order to achieve the concept of intelligent production line, and can reduce the problems caused by human setting errors, thereby improving efficiency.

Keywords Machine learning · Hyperparameter optimization · Processing parameters · High frequency heat treatment

C.-H. Lin (✉) · Y.-H. Tseng · K.-H. Hu · C.-W. Ho
Department of Electronic Engineering, National Kaohsiung University of Science and Technology, Kaohsiung City, Taiwan
e-mail: cslin@nkust.edu.tw

K.-H. Hu · C.-W. Ho
Knowledge Application Development Group, Metal Industry Research and Development Center, Taguig, Philippines

1 Introduction

Heat treatment [1] is a set of industrial processes used to modify the physical properties of materials, and sometimes even their chemical characteristics. Its most prevalent application is in metallurgy. Heat treatment can be applied to the production of various other materials, such as glass, metals, composite materials, and more. Heat treatment involves the use of heating or cooling to achieve extreme temperatures, thus obtaining the desired outcomes. For instance, metallic materials are composed of grains or microcrystalline structures. The properties of these grains are one of the determining factors of the metal's performance. Heat treatment provides a means to control the properties of metals by regulating diffusion and cooling rates within the microstructures. Heat treatment is commonly employed to alter the mechanical properties of metal alloys, thereby controlling factors such as hardness, strength, toughness, ductility, and elasticity [2]. Certain mechanisms may alter the properties of alloys during the heat treatment process. For example, the formation of martensite can lead to intrinsic crystal deformation, while its diffusion mechanism results in changes in alloy [3] homogeneity. Metal heat treatment processes can be broadly categorized into three types: bulk heat treatment, surface heat treatment, and chemical heat treatment. Furthermore, the specific heat treatment processes within each category can vary based on factors such as heating material, temperature, and cooling methods. Using different heat treatment processes for the same type of metal can yield distinct microstructures, thereby leading to different properties.

High-frequency induction hardening [4, 5] is a process that rapidly heats the surface of steel [6] using high-frequency induction, halting the heating the moment the austenitizing temperature is reached, and then employing an appropriate quenching medium for the quenching step. This technique is most suitable for steel with a carbon content of 0.4–0.5% [7], though the complexity of the part's geometry must be considered as intricate shapes might be unsuitable for high-frequency induction hardening. Bulk heat treatment involves uniformly heating the entire workpiece and subsequently cooling it at an appropriate rate, providing a method to alter the overall mechanical properties. The bulk heat treatment of steel can be classified into four categories: annealing, normalizing, quenching, and tempering [8]. Annealing, as the first step, involves heating the workpiece to an appropriate temperature and allowing it to slowly cool based on the material and dimensions, aiming to achieve an equilibrium state within the metal. This process is carried out to attain favorable processability and performance in use. Normalizing involves heating the workpiece to an appropriate temperature and then cooling it, similar to annealing. However, normalizing results in a finer microstructure. As a result, it is often used to enhance the material's machinability. It's sometimes employed as the final heat treatment for components with less stringent requirements. Quenching, on the other hand, involves heating the workpiece and then rapidly cooling it in a quenching medium like organic water solutions, inorganic salts, water, or oil. This process results in hardening of the workpiece but may also render it brittle [9]. To mitigate brittleness, the quenched

workpiece is often subjected to a pro-longed period of heat preservation at temperatures higher than room temperature but below 650 °C, followed by cooling. This technique is known as tempering. Annealing, normalizing, quenching, and tempering are crucial steps in bulk heat treatment, with tempering and quenching closely linked and frequently used in combination. However, in the metal heat treatment industry, enhancing the control of the heating process is a challenge faced by all manufacturing facilities.

For example:

- (a) Currently, due to the prevailing lack of incorporation of intelligent production line concepts in hardware planning, many issues stem from improper system operation and control. This leads to an inability to precisely manage critical process parameters like hardening depth and step differentials, as well as hindered high-efficiency production.
- (b) Considering factors of cost, time, and yield, it has become increasingly challenging to cope with the market's shift from mass production to small-scale, diversified manufacturing. Achieving delivery goals with costs and timelines comparable to those of mass production remains one of the foremost challenges at present.
- (c) According to the Energy Bureau's monthly statistical report, industrial activities account for the highest percentage in the Taiwan region at 51.2%, with basic metal industries constituting the second-highest at 15%. Within the metal heat treatment sector, the heating process poses a challenge that all manufacturing facilities must confront.

In the era of smart manufacturing, artificial intelligence, and the Internet of Things, many companies are leveraging advanced data analytics techniques to integrate these technologies into product production fore-casting, thereby enhancing quality and efficiency. However, currently relying solely on process technology enhancement is insufficient to address the diverse array of product shapes, sizes, weights, materials, and characteristic requirements within the constraints of cost and time. Consequently, many companies are adopting visual platforms coupled with the Internet of Things [10, 11] to establish an integrated system, achieving the goal of an intelligent production line [12]. The Internet of Things refers to the integration of sensors and computational capabilities, along with the collection of data from identifiable objects, facilitated by the Internet for data gathering [13]. Gartner predicts that by 2020, there will be over 20 billion devices (largely from the manufacturing sector) connected to the IoT [14], generating a substantial amount of data, including structured, semi-structured, and unstructured data.

In today's metal processing facilities, there is a limited utilization of artificial intelligence methods to predict machining parameters for products. Instead, most adjustments are made based on the experience of on-site personnel, and often, it is necessary to wait until production is complete to determine whether the requirements have been met. Hence, we aspire to employ machine learning to achieve the goal of predicting machining parameters for products. This approach would aid on-site

personnel in swiftly determining whether the machining of a product meets the required specifications.

2 Machine Learning Models and Hyperparameter Optimization

2.1 *Decision Tree* [15]

A decision tree generates a tree structure based on the training data and uses the learned results to make predictions for new samples. Decision trees are divided into classification and regression, with the difference lying in how the quality of branching is assessed. When the output of the dataset consists of continuous numerical values, the decision tree algorithm falls under the category of regression trees. In the regression tree, as branches expand, the prediction value is determined by the average value of the leaf nodes. The algorithm tests a specific feature of the dataset and assigns data samples to its child nodes after evaluation. At this point, each child node corresponds to a specific feature value within the dataset. Continuing in this manner, the process is repeated until reaching the leaf nodes. When the error value approaches zero, it indicates a better performance.

Using Figs. 1 and 2 as examples, let's assume that the input is represented by x and the output by y , plotting the distribution of data and correct answers on a plane. Assuming a maximum tree depth of 2, in the first layer, the data is split into two parts based on a threshold, let's say $x = 0.496$. When the data points are greater than 0.496, they will extend towards the right, and conversely, when they are less than 0.496, they will extend towards the left. At this point, a branch will emerge and continue to expand downwards, forming the next layer of decision branches. Following this method, the expansion continues until the set maximum depth is reached. At this point, the result will be the final output value of the model.

2.2 *Random Forest* [16]

Random Forest is a supervised machine learning algorithm that falls under Ensemble Learning, as shown in Fig. 3. It combines Decision Trees and utilizes Bagging. Due to its simplicity, accuracy, and flexibility, Random Forest is one of the most commonly used algorithms. Random Forest can be used for both classification and regression analysis. Its non-linear nature makes it highly adaptable to various types of data and scenarios. A random forest is a collection of decision trees, yet there are still some differences between the two. Decision trees tend to create rules for making decisions, while a random forest randomly selects certain features and observes them to construct a forest of decision trees. Then, it derives an average from the results of

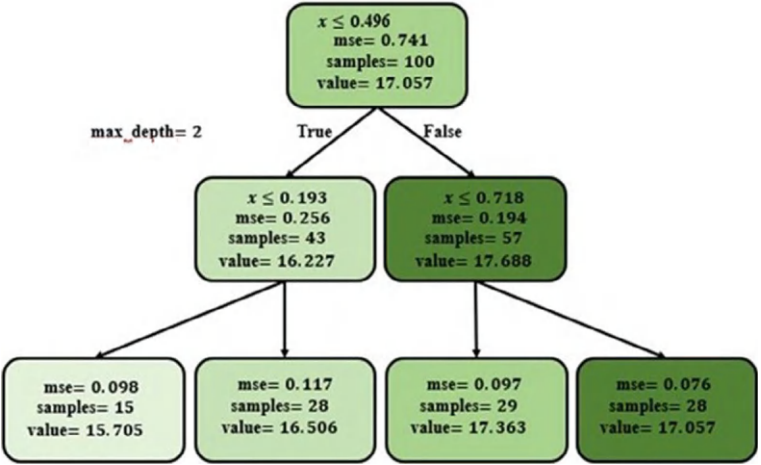


Fig. 1 Example illustration of a regression tree—1

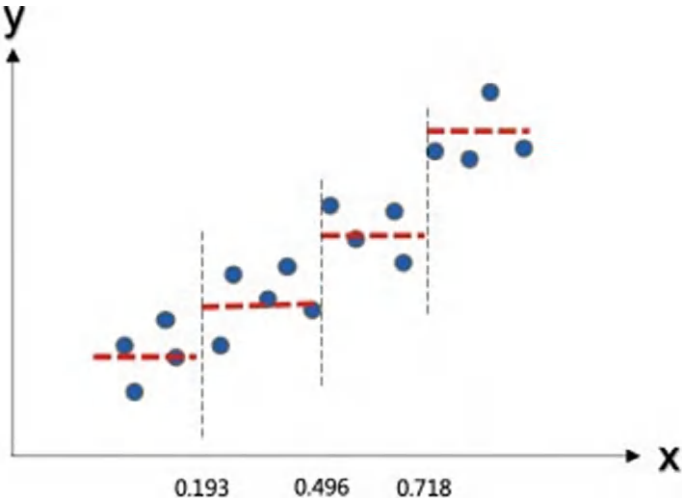


Fig. 2 Example Illustration of a regression tree—2

these individual trees. This theory suggests that a large number of unrelated decision trees can produce more accurate predictions than a single decision tree.

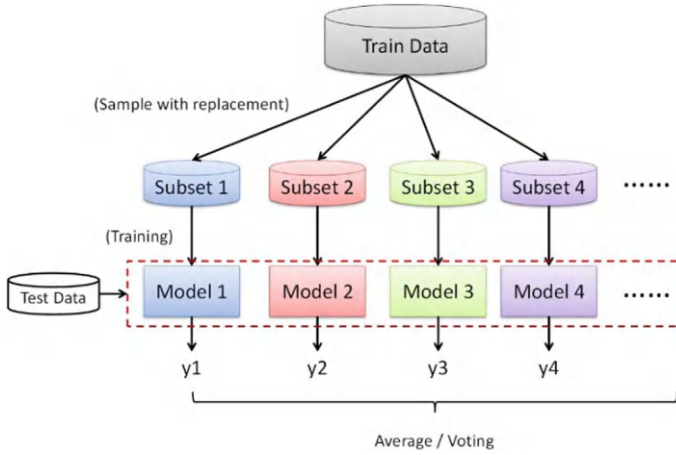


Fig. 3 Illustration of Bagging Bootstrap Aggregation Method

2.3 XGboost (*eXtreme Gradient Boosting*) [17]

XGBoost, short for *eXtreme Gradient Boosting*, is a model proposed by Tianqi Chen, a Ph.D. student at the University of Washington. XGBoost is an enhancement of Gradient Boosting, as shown in Fig. 4. Reference [18] built on the foundation of boosting. The core idea is to utilize each decision tree for predictions. First, a decision tree is built, and the residual between this tree’s predictions and the target answers is calculated. Then, a second decision tree is constructed to predict this residual, and this process continues iteratively. The final result is achieved by summing up the predictions of each decision tree to minimize the residual with the target answers. Hence, XGBoost can be seen as a fusion of the advantages of both Boosting and Bagging techniques. In XGBoost, the original approach of Gradient Boosting is retained, allowing each tree to have mutual correlations, similar to the concept of residual networks. The goal is to ensure that the subsequent trees can avoid making the same mistakes as the previous ones. In addition, XGboost employs a technique called feature random sampling, similar to Random Forest, where during the construction of each tree, a random subset of features is selected. This ensures that each tree’s construction does not involve all the features in making decisions. To achieve more complex models, XGboost incorporates regularization methods into the objective function. Because during training, the model tends to fit the training data and generate many high-degree polynomial functions, which can make it susceptible to noise interference and lead to overfitting. Therefore, to avoid the aforementioned issues, L1/L2 Regularization is used to make the loss function smoother and optimize its resistance to noise interference. Finally, XGBoost uses first and second-order derivatives to generate the next tree. Where the first-order derivative is the Gradient, and the second-order derivative is the Hessian.

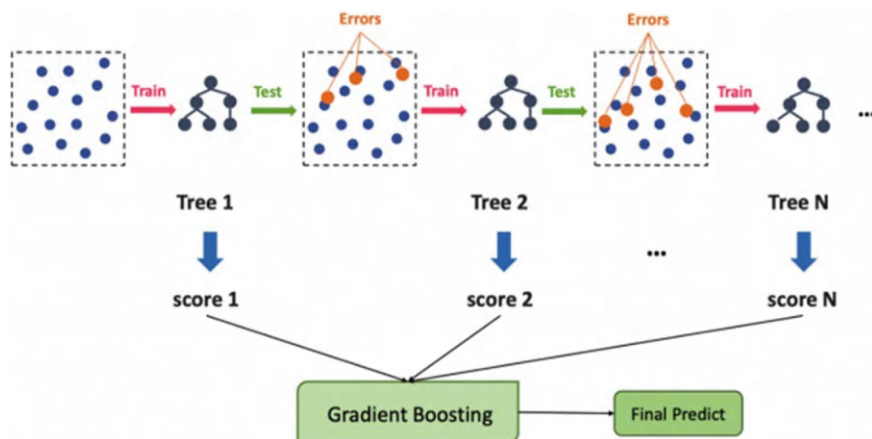


Fig. 4 Boosting illustration [19]

2.4 Multi-objective Bayesian Optimization for High-Dimensional Search Spaces [20]

Bayesian optimization can be considered an efficient, versatile, and sample-efficient optimization method, particularly effective for tuning hyperparameters in machine learning. However, the fundamental form of Bayesian optimization is subject to the following limitations:

- In previous applications, it was common to consider low-dimensional search spaces, but this also limited the number of tunable parameters, typically with fewer than 20 adjustable parameters available.
- Using a typical Gaussian process to substitute for model-based regression, however, results in a cubic-time complexity in the number of sampling points.
- Most methods focus on single-objective unconstrained problems.

Due to the aforementioned issues, Samuel Daulton and others proposed the algorithm of Multi-Objective Regionalized Bayesian Optimization (MORBO). The algorithm utilizes a set of coherent Trust Regions (TR) and optimizes different parts of the Global Pareto (GP) frontier.

MORBO is a coordinated multi-Trust Region (TR) approach for constrained high-dimensional multi-objective Bayesian optimization. Unlike the high-dimensional Bayesian optimization (TuRBO) approach that utilizes multiple independent Trust Regions (TRs), MORBO employs a collaborative strategy where observations across TRs are shared, providing each TR with all available information about constraints within the TR and relevant objectives of local optimization (see Fig. 5).

Algorithm 1: Summary of MORBO

Input: Objective functions f , Number of trust region n_{TR} , Initial trust region length L_{init} , Maximum trust region length L_{max} , Minimum trust region length L_{min} .

Output: Approximate Pareto frontier $\mathcal{P}_n$

```

1 Evaluate an initial set of points and initialize the trust regions  $\mathcal{T}_1, \dots, \mathcal{T}_{n_{TR}}$  using the center selection procedure described
  in Section 3.2 and mark center points as unavailable for other trust regions.
2  $X_0 \leftarrow \emptyset, Y_0 = \emptyset, t \leftarrow 1$ 
3 while budget not exhausted do
4   Fit a local model within each trust region.
5   Select  $q$  candidates using the sequential greedy HVI procedure described in Section 3.1.
6   Evaluate candidates on the true objective functions and obtain new observations.
7   for  $j = 1, \dots, n_{TR}$  do
8     Update trust regions with new observations as described in Section 3.
9     Increment success/failure counters as described in Section 3 for observations from  $\mathcal{T}_j$ .
10    Update edglength  $L_j$  for  $\mathcal{T}_j$ .
11    if  $L_j < L_{min}$  then
12      Terminate  $\mathcal{T}_j$ .
13      Fit GP to restart points  $\mathcal{D}_{t-1} = (X_{t-1}, Y_{t-1})$ :  $f_{t-1} \sim P(f|\mathcal{D}_{t-1})$ .
14      Sample  $\lambda \sim S_+^{M-1}$  and  $\tilde{f}_{t-1} \sim P(f|\mathcal{D}_{t-1})$ , where  $S_+^{M-1} = \{w \in \mathbb{R}_+^M : \|w\|_2 = 1\}$ .
15       $x_t \leftarrow \arg \max_{x \in X} s_\lambda[\tilde{f}_{t-1}(x)]$ , where  $s_\lambda[y] = \min_m (\max(\frac{y_m}{\lambda_m}, 0))^M$  and  $\cdot_i$  denotes the  $i$ th element.
16      Evaluate  $x_t$  on the true objective functions and obtain new observation  $y_t$ .
17      Reinitialize  $\mathcal{T}_j$  with edglength  $L_{init}$  centered at the  $x_t$ .
18      Set  $X_t \leftarrow X_{t-1} \cup \{x_t\}, Y_t \leftarrow Y_{t-1} \cup \{y_t\}, t \leftarrow t + 1$ .
19    Update center to the available point with maximum HVC (globally if  $\mathcal{T}_j$  was terminated otherwise within  $\mathcal{T}_j$ ).
20 return Approximate PF across observed function values.

```

Fig. 5 Multi-objective Bayesian optimization [20]

3 Research Results and Comparisons

This paper will follow the workflow in Fig. 6 for the research. Firstly, it will collect the necessary processing parameter data for high-frequency heat treatment. Next, it will perform data preprocessing, including handling missing values, removing outliers, and eliminating blank fields, to prepare the data for subsequent model training. Finally, the models will undergo initial testing and performance evaluation.

After preprocessing, there were 112 remaining data points for the Long Axis, 443 for the Gear, and 555 for the Tempering Furnace. Subsequently, the processed dataset was divided into training and testing sets. As this study aimed to predict different machining parameters for the three types of products, three separate models were trained, each targeting specific machining parameters, as outlined in Table 1.

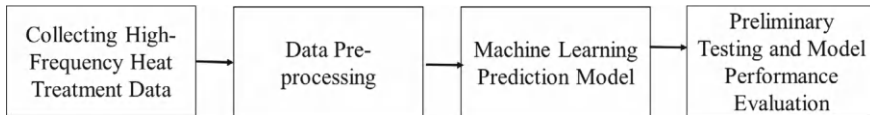


Fig. 6 Research workflow diagram

Table 1 Parameters for predicting each machining item

Machining items	Input parameters	Predictive parameters
Main shaft	Dimensions material, hardness, depth, power	Speed
Gear cutting	Dimensions, material, hardness, power	Speed
Tempering furnace	Processing hardness, tempering hardness, dimensions, material, time	Temperature

4 Experimental Results

This study will compare the performance of various machine learning models with different hyperparameter optimizations. The models under consideration include KNN, AdaBoost, Decision Tree, Random Forest, Gradient Boosting (GB), and XGBoost. The best-performing models among these will be selected, and the study will conclude by determining which model yields the most optimal predictive results.

The evaluation will be conducted using a tenfold cross-validation method to assess the model’s performance. Three evaluation metrics, namely R2 (R-squared), MSE (Mean Squared Error), and MAE (Mean Absolute Error), will be employed to assess the models.

To confirm the effectiveness of hyperparameter optimization, this study will first train models without any hyperparameter optimization, using them as baselines. The results are presented in Tables 2, 3, and 4 as references.

Based on the results, it can be observed that before undergoing hyperparameter optimization, all models showed decent performance, with XGBoost outperforming

Table 2 Results of various models used for predicting the speed of the main shaft are shown below

Method	R ² score (%)	MSE (%)	MAE (%)
KNN	78.34	59.11	35.4
AdaBoost	74.1	60.15	36.28
Decision tree	87.58	45.79	24.83
Random forest	88.58	31.57	15.29
Gradient boosting	88.45	40.31	18.26
XGBoost	89.37	26.84	13.07

Table 3 Results of various models used for predicting the speed of the gear teeth are as follows

Method	R ² score (%)	MSE (%)	MAE (%)
KNN	78.07	11.15	6.09
AdaBoost	79.32	10.52	6.44
Decision tree	81.75	6.48	4.01
Random forest	89.47	3.87	2.11
Gradient boosting	86.27	6.98	4.72
XGBoost	90.71	2.67	1.51

Table 4 Results of various models used for predicting the temperature of the tempering furnace are as follows

Method	R ² score (%)	MSE (%)	MAE (%)
KNN	78.29	51.06	25.21
AdaBoost	77.98	52.02	27.56
Decision tree	83.12	40.27	18.36
Random forest	88.67	15.81	8.42
Gradient boosting	85.79	38.5	15.61
XGBoost	88.73	15.76	8.31

the other models in all evaluation metrics. However, the performance of KNN and AdaBoost was not satisfactory. Therefore, in the subsequent steps, hyperparameter optimization will be conducted separately for the remaining four models to compare their optimized performance.

After obtaining the unoptimized results as mentioned earlier, the next step involves optimizing the better-performing models using multi-objective Bayesian optimization. The results of this optimization process are presented in Tables 5, 6, and 7.

Table 5 The results of each model after undergoing multi-objective Bayesian optimization and their use in predicting the speed of the main shaft are shown in the tables below

Method	R ² score (%)	MSE (%)	MAE (%)
Decision tree+ MOBO	93.17	42.2	20.36
Random forest+ MOBO	94.88	25.1	12.07
Gradient Boosting+ MOBO	93.48	30.04	13.24
XGBoost+ MOBO	97.15	20.07	10.38

Table 6 The results of each model after undergoing multi-objective Bayesian optimization and their use in predicting the speed of the gear cutting are shown in the tables below

Method	R ² score (%)	MSE (%)	MAE (%)
Decision tree+ MOBO	86.81	5.04	2.87
Random forest+ MOBO	96.17	2.91	1.38
Gradient Boosting+ MOBO	92.17	4.34	2.9
XGBoost+ MOBO	97.34	1.35	0.81

Table 7 The results of each model after undergoing multi-objective Bayesian optimization and their use in predicting the temperature of the tempering furnace are shown in the tables below

Method	R ² score (%)	MSE (%)	MAE (%)
Decision tree+ MOBO	90.11	30.83	13.41
Random forest+ MOBO	95.04	10.48	4.09
Gradient boosting+ MOBO	91.54	33.42	10.58
XGBoost+ MOBO	95.96	10.17	4.04

5 Conclusion

This study establishes a systematic process to effectively enhance the prediction of processing parameters for manufactured products. It begins with data pre-processing, involving data cleaning and feature extraction, and culminates in multi-objective Bayesian optimization to improve the overall model performance. In comparison to the initial baseline, this research achieves approximately a 7.78% improvement in the final performance (R2) for the longitudinal axis, approximately a 6.63% improvement in the final performance (R2) for the gear, and approximately a 7.23% improvement in the final performance (R2) for the tempering furnace.

References

1. Heat Treating Society, ASM International. What is heat treating. The Hearth of Industry
2. ZIA AW, Zhou Z, Po-wan S, Li L, Yan K (2017). The effect of two-step heat treatment on hardness, fracture toughness, and wear of different biased diamond-like carbon coatings. *Surf Coat Technol* 320:118–125. <https://doi.org/10.1016/j.surfcoat.2017.01.089>
3. Gupta SP (2002). Solid state phase transformations. Allied Publishers Private Limited, pp 28–29
4. Department of Mechanical Engineering and Engineering Research Institute, Gyeongsang National University, 501, Jinju-daero, Jinju-si, Gyeongsangnam-do 52828, Korea Metals 2020, 10(11):1484
5. High frequency heat treatment method and device. <https://patents.google.com/patent/JP2005325408A/en>
6. Vanpaemel J (1982) History of the hardening of steel: science and technology. *J Phys Colloq* 43(C4):C4-847–C4-854
7. Abou-Jahjah J, Dobránszky J Combined heat treatments. <https://core.ac.uk/download/pdf/11857591.pdf>
8. Dossett JL, Boyer HE (2006) Practical heat treating. ASM International, pp 17–22
9. Najdahmadi A, Zarei-Hanzaki A, Farghadani E (1 Feb 2014) Mechanical proper-ties enhancement in Ti–29Nb–13Ta–4.6Zr alloy via heat treatment with no detrimental effect on its biocompatibility. *Mater Des*
10. Weiser M (1991) The computer for the 21st CENTURY. *Scientific American*. <https://www.scientificamerican.com/article/the-computer-for-the-21st-century/>
11. Yang C, Shen W, Wang X (2018) The Internet of Things in manufacturing: key issues and potential applications. *IEEE Syst Man Cybern Mag*
12. Raji R (1994) Smart networks for control. *IEEE Spectrum*

13. Ning H, Liu H (2012) Cyber-physical-social based security architecture for future internet of things. *Adv Internet Things* 2:1–7
14. Hung M (2017) Leading the IoT: Gartner insights on how to lead in a connected world. https://www.gartner.com/imagesrv/books/iot/iotEbook_digital.pdf
15. Menzies T, Hu Y (Oct 2003) Data mining for very busy people. *IEEE Computer*, pp 18–25
16. Ho TK (1995) Random decision forests. In: *Proceedings of 3rd international conference on document analysis and recognition (Montreal, Que., Canada: IEEE Comput. Soc. Press)*, vol 1, pp 278–282
17. XGBoost: a scalable tree boosting system. Tianqi Chen, Carlos Guestrin University of Washington. <https://www.kdd.org/kdd2016/papers/files/rfp0697-chenAemb.pdf>
18. Bergstra J, Bengio Y Random search for hyperparameter. <https://www.jmlr.org/papers/volume13/bergstra12a/bergstra12a.pdf>
19. Gradient boosting. <https://machine-learning.paperspace.com/wiki/gradient-boosting>
20. Daulton S, Eriksson D, Balandat M, Bakshy E (2022) Multi-objective Bayesian optimization over high-dimensional search spaces . To appear at UAI 2022, 24 pages
21. Cover TM, Hart PE (1967) Nearest-neighbor pattern classification. *IEEE Trans Inf Theory* 13(1):21–27
22. Snoek J, Larochelle H, Adams RP (2012) Practical Bayesian optimization of machine learning algorithms. *NIPS*
23. Dewancker I, Mccourt M, Clark S (2018) Bayesian optimization primer
24. Nogueira F Bayesian optimization. <https://github.com/fmfn/BayesianOptimization>

Protocol Security Fuzz Testing with Data Augmentation



Nain-Fang Tu, Chia-Ling Lu, and Chi-Yuan Chen

Abstract Fuzz testing is a crucial technique for uncovering system vulnerabilities, particularly focusing on security. However, its effectiveness is challenged when testers are unfamiliar with communication protocols. This research addresses these challenges, taking the Bluetooth Low Energy (BLE) protocol as a case study. We introduce an approach based on Generative Adversarial Networks (GANs) to create diverse data across varied protocol categories, significantly improving the quality of inherently imbalanced protocol testing datasets. Our methodology allows for application to different testing targets, enabling efficient generation of fuzz testing data without prerequisite knowledge of target specifications, thereby simplifying the process for testers and advancing fuzz testing methodologies in security testing domains.

Keywords Fuzz testing · Artificial intelligence · Bluetooth low energy

1 Introduction

Fuzz Testing is generally acknowledged as a method grounded in black-box testing, proving its efficacy in probing and identifying vulnerabilities within protocols by generating input data packets. These packets are dispatched to various communication layers of the target device, assessing its reactions and security robustness.

Bluetooth technology, as the target protocol example for this study, has been prevalent for 25 years. In 2010, the introduction of version 4.0 officially bifurcated Bluetooth development into two primary trajectories: Basic Rate/Enhanced Data Rate (BR/EDR) and Low Energy (LE), differentiated primarily through chip and system design. The fundamental construct of the Bluetooth core system involves a Host end, typically the software component, and one or multiple Controller ends, the

N.-F. Tu · C.-L. Lu · C.-Y. Chen (✉)
National Ilan University, I-Lan, Taiwan
e-mail: chency@niu.edu.tw

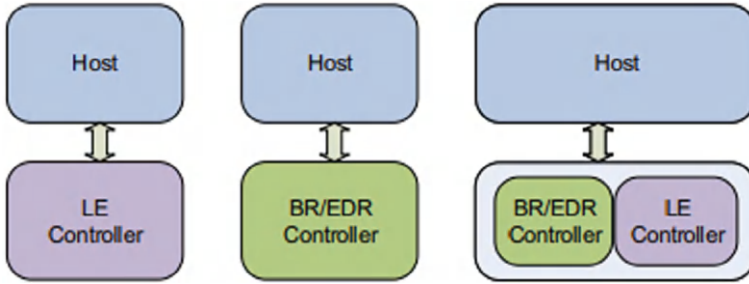


Fig. 1 Bluetooth host and controller combinations [1]

hardware. Figure 1, taken from the Bluetooth Core Specification [1], delineates the composition and categories of the Bluetooth core system in its latest 5.4 version.

Despite Bluetooth Low Energy (BLE) technology emerging as one of the most popular communication technologies in IoT applications, its inherent objective to minimize energy consumption has constrained its security design, making it susceptible to inherent protocol vulnerabilities. For testers unfamiliar with Bluetooth protocols, generating a number of test packets through existing knowledge poses a considerable challenge. This study proposes a generative adversarial network approach to ensure the produced data meets specific categories and exhibits diversity, aiming to augment the originally imbalanced and sparse packets in BLE.

In subsequent sections, we will present a concise review of the relevant literature on fuzz testing and Generative Adversarial Networks in the Related Works section. We will explore the causes of dataset imbalance through the collection of real BLE packets. Following that, we will introduce the framework proposed in this study. Finally, discussions on the research conclusions and prospective research directions will be conducted.

2 Related Works

2.1 Bluetooth Low Energy (BLE)

This study concentrates its testing scope on BLE. Figure 2 illustrates the architecture of BLE [1], segmented into three components: LE Host, Host Controller Interface (HCI), and LE Controller, each enveloping the content of various protocol layers. When a Bluetooth device receives signals, modulation occurs at the Physical Layer, with the modulated signals transmitted to the Link Layer. Here, the HCI primarily facilitates data transfer between layers and manages connections. The L2CAP Layer is responsible for packet flow management, while the Security Manager Protocol (SMP) Layer handles encryption for inter-device connections. The Generic Attribute Profile (GAP) is not a protocol layer in a strict sense; it primarily defines the types of

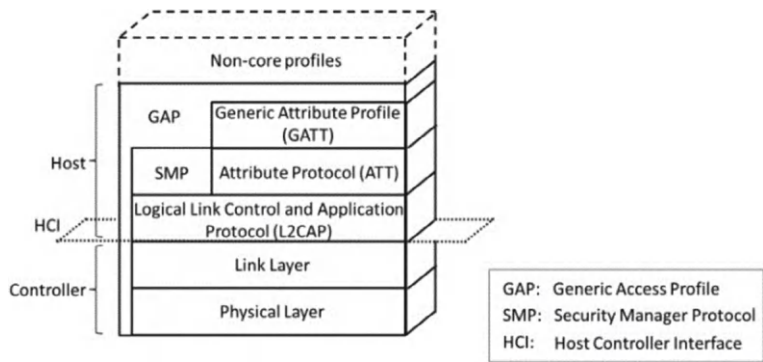


Fig. 2 Architecture of bluetooth low energy

data and services between devices, its definitions penned in the Applications Layer, and the Attribute Layer (ATT) utilizes GATT definitions to generate actual data packets.

Collecting packets are classified based on the fundamental architecture of BLE and according to the communication conditions between devices. The independence of the Controller side implies our inability to comprehend the system’s operational conditions; control is only feasible once the packets are on the Host side. Table 1 demonstrates the categorization of testable ends within Bluetooth Low Energy packets. Generally, Bluetooth packets are encapsulated through layering, resulting in packets that are both L2CAP Layer with SMP Layer content. The categorization is based on the primary function of packets and classifications by packet analyzers like Wireshark. Figure 3 illustrates the protocol packet categories listed by the Protocol Hierarchy feature in Wireshark, in which HCI_ACL data packets can be divided into L2CAP sub-categories, and L2CAP can be further broken down into different protocol data packets like ATT and SMP.

Bluetooth devices are divided into Central Devices and Peripheral Devices. When not connected, peripheral devices continually broadcast information about themselves. A whitepaper [2] addressed man-in-the-middle attacks on Bluetooth, outlining how the deployment of forged nodes can compromise targets and intercept data transmissions between devices. Furthermore, another study [3] explored existing vulnerabilities in BLE protocols and introduced respective attack methodologies. By

Table 1 Categories of testable bluetooth low energy packets

Type	Packet type
Host controller interface	HCI_EVT
	HCI_CMD
Host	L2CAP
	SMP
	ATT

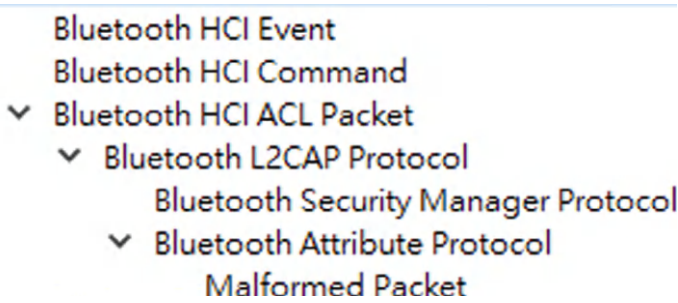


Fig. 3 Protocol hierarchy for bluetooth

analyzing each layer of the protocol, the authors identified design flaws leading to security vulnerabilities, including authentication mechanisms, encryption methods, and Application Programming Interfaces (APIs). They implemented attacks by continually sending specific packets, utilizing open-source Proof-of-Concept (POC) packet injectors for realization.

2.2 Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GAN) were first introduced by Goodfellow et al. [4], generating data that closely mirrors real samples through adversarial training. Presently, GAN has been extensively applied across various domains, such as images, sound, and text. In 2017, Yu et al. [5] proposed a variant of GAN called Sequence Generative Adversarial Networks (SeqGAN), with the main goal of generating sequential data that meet specific conditions. This variant holds significant applicative value for applications requiring extensive data training, such as speech recognition and natural language generation.

As illustrated in Fig. 4, the overall structure of SeqGAN adheres to the fundamental principles of typical adversarial training networks. The Generator (G) is tasked with producing sequence samples composed of words, while the Discriminator (D) differentiates whether the generated sequence is real or generated. Furthermore, SeqGAN integrates several core concepts from reinforcement learning: State (S), Action (A), and Reward (r). State (S) represents the number of words generated at a certain point in time; Action (A) decides the word to be generated based on the previous word's state and model parameters; and Reward (r) evaluates the generated sequence, awarding points if the sample successfully deceives the Discriminator (D) and none if it fails.

SeqGAN, leveraging concepts of reinforcement learning, scores the generated samples during each generation process instead of awarding points only after the completion of the entire sequence. This design overcomes the traditional GAN's

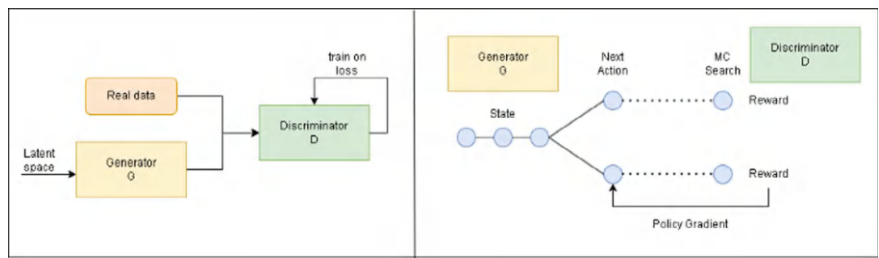


Fig. 4 SeqGAN architecture

challenge of using gradient updates on discrete data. Figure 5 summarizes the main workflow of SeqGAN:

1. Pre-training the Generator (G) using Maximum Likelihood Estimation (MLE).
2. Pre-training the Discriminator (D) using faked samples from Generator (G) and real data samples as inputs.
3. Adversarial Training Phase for Generator (G): For each time step, calculate the reward values and use Discriminator (D) along with Monte Carlo Search (MCS) to compute the current reward, then update the policy gradient.
4. Adversarial Training Phase for Discriminator (D): Train with real data as positive samples and samples from Generator (G) as negative samples.

In 2016, Odena et al. [6] unveiled the Auxiliary Classifier GANs (ACGAN), which can generate images with specific labels through an auxiliary classifier. However, as it adopts a supervised model, it necessitates substantial and balanced labeled data to manifest stable generation quality, primarily addressing image generation tasks during data balance.

To mitigate the negative impact on model accuracy due to insufficient sample data in data classification or prediction tasks, Mariani et al. [7] introduced BalancingGAN (BAGAN). While data augmentation techniques are common enhancements in original datasets, their usage in unbalanced sample classes can result in uneven class distribution, ultimately impairing the model’s generative performance on minority classes. BAGAN serves as a solution in image classification tasks, addressing the issues faced by ACGAN.

As depicted in Fig. 6, BAGAN primarily integrates the techniques of GAN and Autoencoder. BAGAN’s methodology is categorized into three phases: Autoencoder Training, GAN Initialization, and GAN Training. Initially, GAN is initialized using an autoencoder to attain optimal initial solutions and avoid mode collapse, with the encoder portion inferring the distribution of latent space among different classes. Concurrently, BAGAN employs the same network topology to design these modules, aligning the decoder with the generator and the initial layers of the encoder with the discriminator. In BAGAN, knowledge from the autoencoder is transferred to GAN through the appropriate initialization of parameter weights.

The subsequent phase involves constructing a class-conditional latent vector generator, which is realized by learning the probability distribution of different

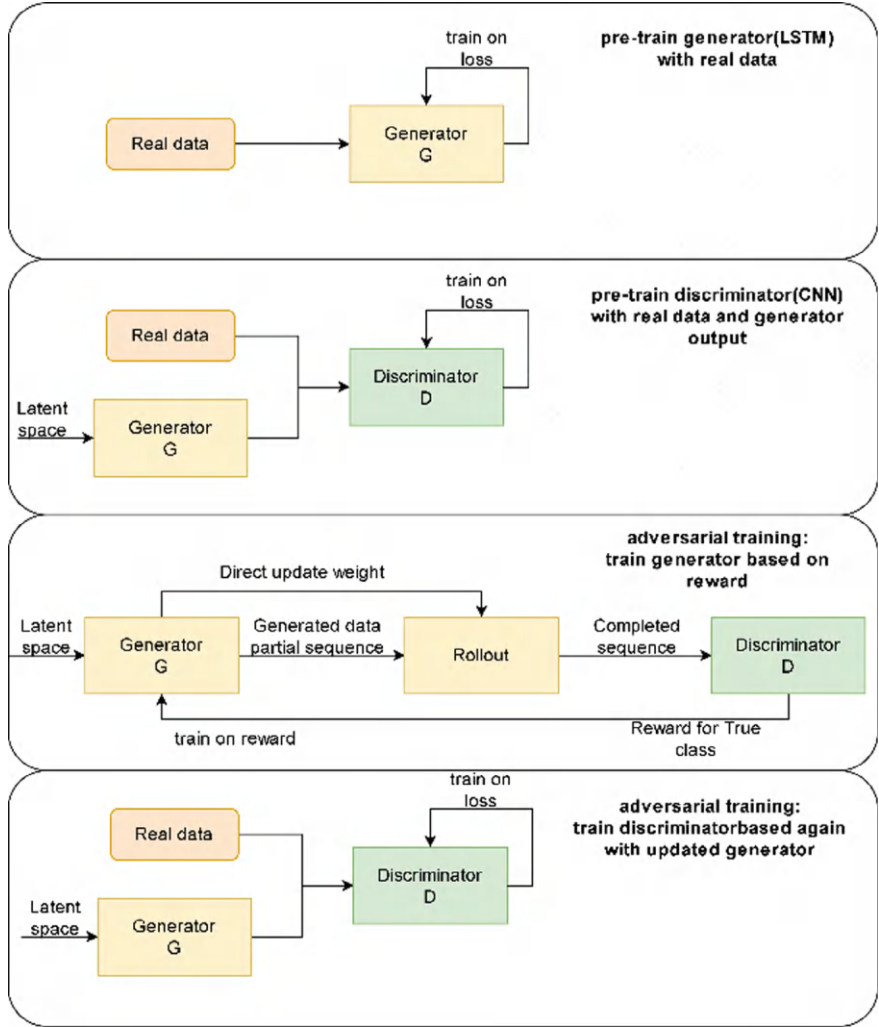


Fig. 5 SeqGAN training process

classes in the latent space. Finally, the weights of the generator and discriminator are fine-tuned in the conventional adversarial training process.

The primary stages of BAGAN are as follows:

1. Autoencoder Training Phase: The autoencoder is trained using all training set samples (including minority and majority classes) without explicit class knowledge.
2. GAN Initialization Phase: The Generator (G) and Discriminator (D) possess explicit class knowledge, with G responsible for generating samples of different classes and D determining whether the samples are fake or of the specified class.

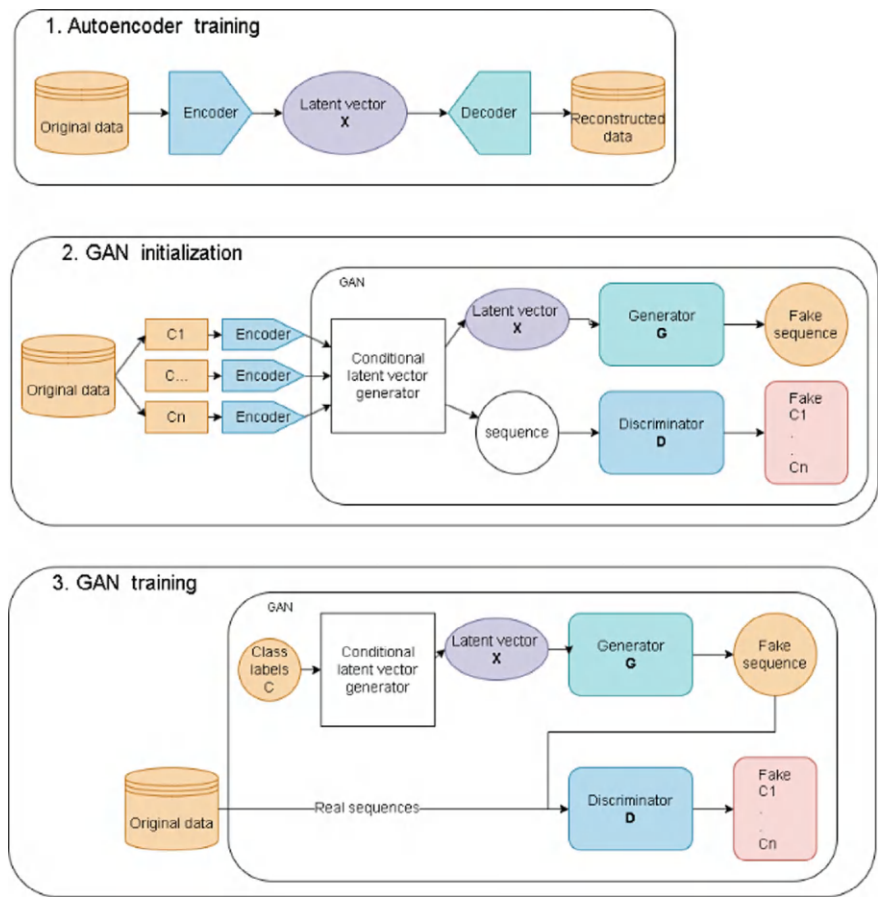


Fig. 6 BAGAN process

3. GAN Training Phase: Data batches flow through Generator (G) and Discriminator (D) for weight fine-tuning to optimize the loss function, with $1/(n + 1)$ samples being fake in each batch to balance the fake class. The Generator (G) extracts latent vectors from the class-conditional latent vector generator as inputs, which receives uniformly distributed class labels as inputs. The Discriminator (D) uses the real dataset and forged samples processed by the Generator (G) as inputs.

3 Evaluation of Bluetooth Low Energy Packets

To better comprehend the challenges associated with collecting BLE data in real-world scenarios, this study utilizes a dataset composed of an Android phone representing central devices and a Xiaomi Mi Band 6 symbolizing peripheral devices.

Bluetooth packets are captured by enabling the corresponding option in the developer settings of the mobile phone and connecting to the test device. Traffic packets are then generated through the existing features of the test device and the corresponding mobile application, Zepp Live APP. Subsequently, based on the contents of Table 1, the captured packets are categorized into five classes. Lastly, Wireshark is used to convert the collected samples into CSV files for analysis and TXT files for packet storage, forming a classified dataset after data preprocessing.

Initially, defining what constitutes a successful sample is imperative. The foundational criterion is that generated samples must conform to Bluetooth specifications. Although the encapsulation of Header and Payload is generally similar across different layers, minor discrepancies do exist. A packet is deemed qualified if, after analysis via Wireshark, it is displayed under the correct protocol, belongs to one of the categories in Table 1, and doesn't present significant error messages. Conversely, samples not conforming to any category in Table 1 or those displaying no information or severe error messages in Wireshark's Info field are considered unqualified.

Figure 7 illustrates the distribution of sample categories in the initial dataset. It reveals a palpable imbalance in dataset classes. The categories, in descending order of packet count, are ATT (56,894), EVT (29,150), CMD (250), L2CAP (4), and ACL (2).

The number and types of packets collected during BLE connections may vary due to several factors, such as the amount of data transmitted during the connection, the distance between the devices, and signal strength. This study captures packets directly from device connections and provides a preliminary explanation of the distribution and quantity of packet categories, broadly aligning with the normal conditions of Bluetooth communications.

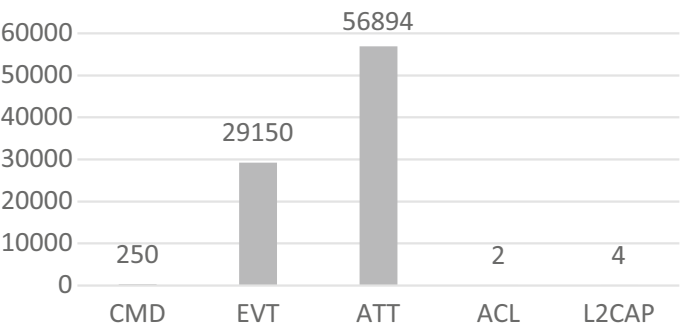


Fig. 7 Distribution of BLE packet categories

4 SEQ-BAGAN

Our research introduces the SEQ-BAGAN model, an architecture derived from the foundational principles of SeqGAN and BAGAN. Its core focus lies in generating sequence-based BLE data packets, addressing the imbalance in the original dataset. Due to a paucity of certain sequence categories in the primary dataset, our approach first harnesses the GAN to learn latent features specific to classification problems from all sequences. These features then facilitate the creation of novel sequences within underrepresented categories. For instance, once the GAN grasps that all defined packet categories share the same Header and Packet Type fields, it can leverage this knowledge to generate other categories.

Figures 8 and 9 delineate the architecture and workflow of our proposed model, respectively. Within the GAN framework, both the Generator (G) and Discriminator (D) undergo modifications based on SeqGAN, integrating insights from BAGAN's autoencoder. For the generator G, its input derives from latent space vectors, paired with a decoder (Δ) to produce fabricated sequence samples. Conversely, the discriminator D processes sequence samples, its initial layers aligning with an encoder (E), transforming sequences into latent space vectors for categorical discernment.

The generator G utilizes the length of latent vectors as input. An embedding layer translates integer sequences into vector space, serving as input for the LSTM. Ultimately, the LSTM output vector undergoes a conversion into a probability distribution, trained using stochastic gradient descent (SGD). The discriminator D, based on a convolutional neural network, shares its input and embedding layer with G. After navigating 12 pairs of convolutional and pooling layers, the outcomes concatenate and flatten. A subsequent Highway layer balances input–output ratios, counteracting the vanishing gradient issue and enabling deeper network learning. The final dense layer in D determines whether a sample belongs to a specific category or is counterfeit, employing Categorical Crossentropy as its loss function.

A pivotal component in the model, the Class-conditional Latent Vector Generator (θ), produces latent vectors correlating with specific categories, subsequently generating corresponding fabricated sequence samples. This concept, an evolution from Conditional Generative Adversarial Networks (CGAN) [8], utilizes labels during the generation phase, enhancing sample controllability and interpretability. In Phase 2, θ 's input comprises the original dataset transformed by the encoder E, aligning with the probability distribution vector of various categories in latent space, then fed into G to produce sequences for specific categories. These vectors, during the initialization stage, can train G's initial weights, offering an improved starting point and enhancing generation capacity. In Phase 3, θ ingests specific category labels C to create latent vectors of that category, which then serves as G's input to produce sequences of that particular category. Through this design, we can manipulate θ 's input to produce samples for minority categories, amplifying the data augmentation effect.

The experimental results of SEQ-BAGAN are illustrated in Fig. 10, where a substantial number of samples have been generated for each category to support

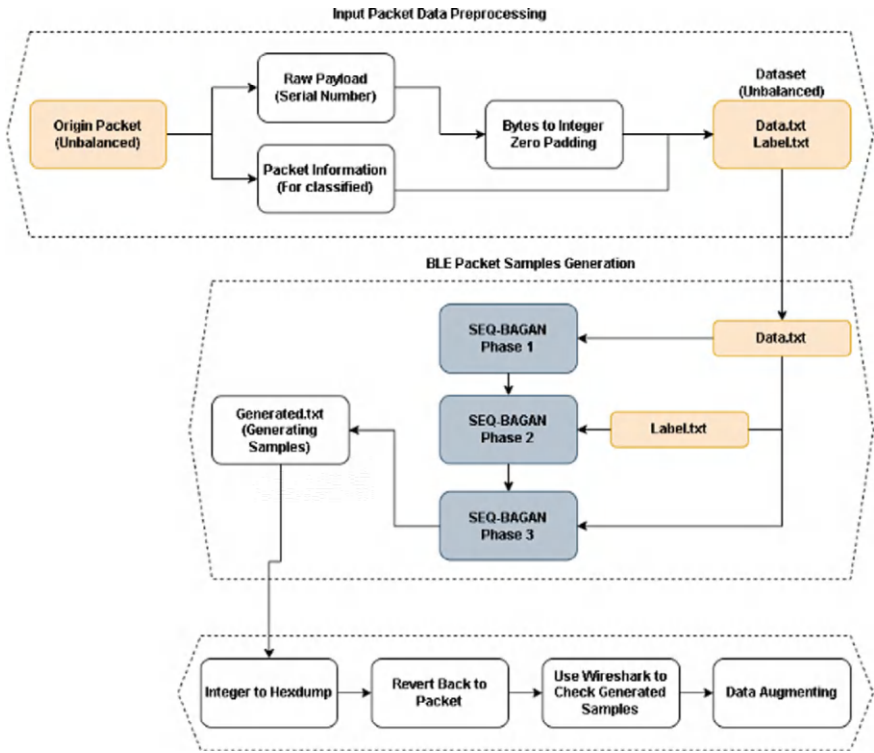


Fig. 8 SEQ-BAGAN architecture

fuzz testing, resulting in an overall balanced class distribution. Notably, the largest category does not occupy more than half of the proportion, indicating the method we proposed possesses tangible enhancement capabilities in addressing imbalanced datasets. Our research proposes a generative adversarial network-based framework specifically tailored for sequence sample generation of BLE data packets. This structure can generate samples for specified categories, fostering dataset balance and enhancing the generalization capacity of trained models. Furthermore, these generated samples can be employed for fuzz testing, pinpointing vulnerabilities in the BLE protocol, and addressing the challenges of data category imbalance and scarcity in this domain.

5 Conclusions

This study addresses the issue of class imbalance within Bluetooth Low Energy protocol by proposing a novel adversarial generative network architecture named SEQ-BAGAN. This architecture amalgamates the advantages of SeqGAN and

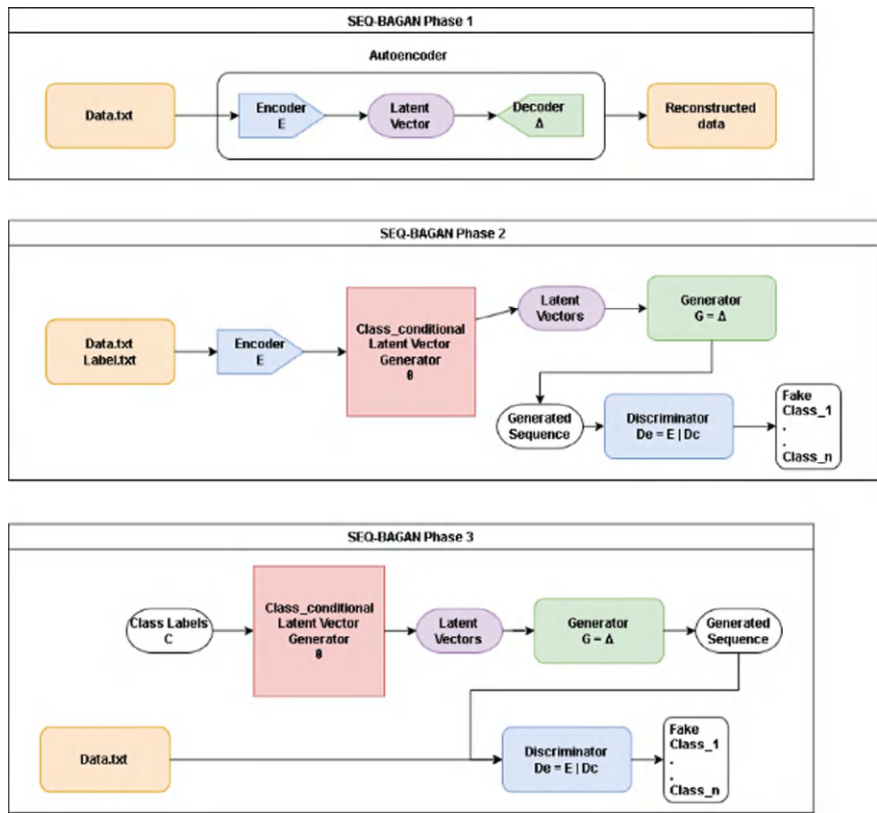


Fig. 9 SEQ-BAGAN process

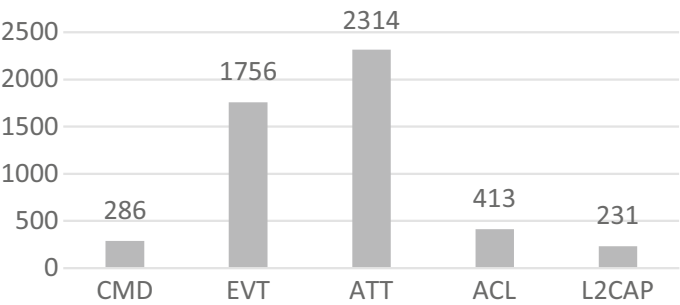


Fig. 10 Distribution of generated BLE packet categories

BAGAN to enhance the data packets in BLE and resolve the issue of data imbalance. This research also involved the collection and evaluation of datasets from prevalent low-energy Bluetooth devices available in the market. In our future research directions, we plan to:

1. Reassess the simulated scenarios for dataset collection to ensure heightened realism and representativeness.
2. Optimize the architecture of the adversarial generative network model to generate samples of specific categories with improved quality.
3. Investigate and evaluate the resemblance between generated and original samples, using the generated samples as a means for data augmentation, and assess the performance of the augmented dataset in comparison to the original one.
4. Besides evaluating whether the generated samples conform to the specified categories, we will also consider the diversity among the generated samples to ensure more comprehensive and generalized model training.

By further refining and expanding the methodology proposed in this study, we aspire to address the problem of class imbalance in BLE more effectively and enhance the generalization capability of the associated models.

References

1. Bluetooth SIG (2023) Bluetooth core version v5.4 specification
2. Jasek S (2016) Gattacking bluetooth smart devices. In: Black Hat USA conference
3. Garbelini ME, Clark SS, Kumar RBS (2020) Sweyntooth: unleashing mayhem over bluetooth low energy. In: Proceedings of the 2020 USENIX conference on USENIX annual technical conference (ATC)
4. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2020) Generative adversarial networks. *Commun ACM* 63(11):139–144
5. Yu L, Zhang W, Wang J, Yu Y (2017) Seqgan: sequence generative adversarial nets with policy gradient. In: Proceedings of the AAAI conference on artificial intelligence, vol 31, no 1
6. Odena A, Olah C, Shlens J (2017) Conditional image synthesis with auxiliary classifier GANs. In: Proceedings of the international conference on machine learning (ICML). PMLR
7. Mariani G, Scheidegger F, Istrate R, Bekas C, Malossi C (2018) Bagan: data augmentation with balancing GAN. [arXiv:1803.09655](https://arxiv.org/abs/1803.09655)
8. Mirza M, Osindero S (2014) Conditional generative adversarial nets. [arXiv:1411.1784](https://arxiv.org/abs/1411.1784)

Preserving the Higher Coverage Guided of an ICS-Oriented Fuzzy Testing Technique



Ying-Chin Chen, Yang-Hong Yi, Chit-Jie Chew, and Jung-San Lee

Abstract Reacting to the industrial 4.0 era, the information integration capabilities of smart manufacturing and industrial Internet of things (IIoT) have become a cornerstone of transformation in the traditional manufacturing industry. The industrial control system (ICS) is the significant communication of the factory environment, which is the main attacking target. Therefore, how to discover the potential vulnerability is the most important issue. Fuzzy testing is one of the vulnerability exploration techniques, however, the classical one is not suitable for ICS. The coverage guided is lower than that of the information technology situation. To address this challenge, the authors propose an ICS-oriented fuzzy testing (ICS-FT) technique preserving the higher coverage guided. The algorithm of dynamic link database injection is used to improve the coverage rate within the Windows system. Especially, the authors divide the zones and conduits for the testing bed based on IEC 62,443 and Purdue model. In the experiments, the proposed approach consistently exhibits substantially improved coverage rates compared to conventional fuzz testing, such as AFL and AFLFast. Moreover, ICS-FT incorporates a significantly greater number of testing inputs. This can afford a comprehensive understanding of the testing condition.

Keywords ICS · Fuzzy testing · IEC 62443

1 Introduction

Reacting to the industrial 4.0 era, the information integration capabilities of smart manufacturing [1] and industrial Internet of things (IIoT) [2] have become a cornerstone of transformation in the traditional manufacturing industry, as shown in Fig. 1. The industrial control system (ICS) acts as a bridge between physical device and virtual system. Conversely, ICS is the main attacking target [3]. Most of the devices in the factory have the demands of network requirements due to the IIoT. As to

Y.-C. Chen · Y.-H. Yi · C.-J. Chew · J.-S. Lee (✉)

Feng Chia University, No. 100 Wenhwa Rd., Taichung 40724, R.O.C, Taiwan

e-mail: leejs@fcu.edu.tw

network communication, the insecure protocol might lead to an exponential increase in the number of malicious attacks against ICS each year [4]. In 2010, the virus Stuxnet attacked Iran’s Bushehr Nuclear Power Plant [5], which is suspected jointly developed by the United States and Israel. The malicious instruction was written into the programmable logic controller (PLC) of the nuclear power plant, causing the centrifuge to self-destruction. PLC is a normal component in ICS. After that, the revealed vulnerability of ICS dramatically increases. During the year 2018, the global leading enterprise of semiconductor industry, Taiwan Semiconductor Manufacturing Co., Ltd. (TSMC), suffers from the ransomware attack of Wannacry [6]. The production line is crashed seriously by the virus. After two years, the American factory of the world technology manufacturing Foxconn Technology Group is attacked by ransomware LockBit 2.0 so that the internal business suspends [7]. From the above incidents, the hacker might transfer the target of cyber attack from information technology (IT) to operation technology (OT).

In fact, Williams has proposed Purdue enterprise reference architecture (PERA) for the specific industrial field, which classifies the six levels of communication and three kinds of area [8]. PERA is regarded as the first ICS design for facing cyber-security issues. Later on, the International Society of Automation (ISA) published the ISA/IEC 62443 series of standards, which defines requirements and processes for implementing and maintaining electronically secure industrial automation and control systems (IACS) [9]. The ISA/IECS 62443 standards offer guidance for key stakeholder to ensure the safety, integrity, reliability, and security of ICS. However, the organization bears a considerable cyberattack for IACS even though people have the institutionalized management. The main reason is the zero-day attack [10], in which the attack method is based on unpatched weakness to exploit. Therefore, how to discover the potential vulnerability is the most important issue in ICS. Fuzzing testing

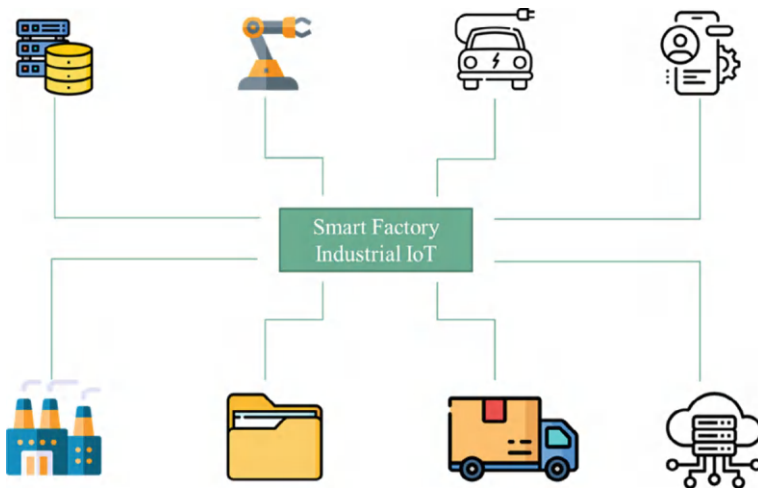


Fig. 1 The diagram of smart factory and IIoT

is one of the general techniques to use to implement the vulnerability discovery. Fuzzing testing automatically examines the input value through iteration until the system crashes. Afterwards, the researcher is able to analyze the input data to find out the potential vulnerability effectively. Zalewski has released the American fuzzy lop (AFL) in 2013, which is determined as the classical fuzzy testing [11]. After that, the researchers have designed the lightweight version AFLFast [12] and specific version AFLSmart based on AFL [13]. Fuzzing testing is a mature technique in the field of information technology (IT). Regarding the operation field (OT) application, the CGFuzzer and V-Fuzz have improved the coverage guided because of adopting deep learning [14]. Unfortunately, the input point of fuzzing testing is seldom considered in detail.

Thus, the authors propose a novel ICS-oriented fuzzy testing (ICS-FT) technique through varied input points. This can enhance the coverage guided. Also, the algorithm of dynamic link database injection (DLL) [15–18] is used to improve the coverage rate within the Windows system, which the outdated Windows server is basically used in ICS. Especially, the authors divide the zones and conduits for the testing bed based on IEC 62443 and Purdue model. In the experiments, the proposed approach consistently exhibits substantially improved coverage rates compared to conventional fuzz testing, such as AFL and AFLFast. Moreover, ICS-FT incorporates a significantly greater number of testing inputs. This can afford a comprehensive understanding of the testing condition.

The rest of the research is organized as follows. Preliminaries are introduced in Sect. 2, and the details of the proposed scheme are explained in Sect. 3. The experiments and analysis are discussed in Sect. 4, followed by the conclusions in Sect. 5.

2 Preliminaries

Here the authors introduce the components and protocol of ICS environment related to the proposed scheme in the following.

2.1 *The Significant Components of ICS*

- **Supervisory control and data acquisition (SCADA) system**

SCADA system is used to monitor the data and alarm of factory immediately to check whether the condition is stable or not for operation members, such as human machine interface (HMI), monitoring software, and remote terminal unit.

- **Control system**

Control system is used to execute the command to make the facilities normally operate, such as programmable logic control (PLC) and distributed control system (DCS).

- **Communication networks**

Communication networks are used to construct the successful communication channel among each system of ICS, such as POWERLINK [19], Modbus [20], and OPC UA [21].

2.2 *The ICS Communication Protocol-Open Platform Communications Unified Architecture (OPC UA)*

OPC UA is a communication standard for data exchange between sensors in an ICS environment [22]. OPC foundation has designed OPC UA to mitigate the flaws of OPC data access (OPC DA), which has the limits of Windows-platform dependency, security issue, and low-portability. OPC UA has the capability of a data model, which can define multiple data structures of device and system. The data model includes four units, as described in the following. The diagram of OPC UA data model is displayed in the Fig. 2.

- **OPC UA meta model**

Meta model is the basic of information model, which is used to define the essential information and the rules of how to execute the object-oriented. The information is composed of variable, method, and event trigger capability.

- **Built-in information model**

Built-in information model is the basic model, which is able to process the data, such as interface of client and debug.

- **Companion information model**

Companion information model is used for standard information model of a specific field, which is developed by professions of specific domain and OPC foundation, such as AutomationML, ISA 95, and BACnet.

Fig. 2 OPC UA data model



- **Vendor specific information model**

The vendor specific information model is defined for special vender, which is used to extend the existed standard model for supporting corresponding function and requirement.

3 The Proposed Framework

The proposed scheme focuses on varied input points to establish a novel ICS-oriented fuzzy testing (ICS-FT) technique preserving the higher coverage guided. ICS-FT has three phases, including seed generation, mutation, and trace. Firstly, ICS-FT acquires the seed from OPC server. Next, the mutation phase generates diverse conditions via the input seed to mimic different situations in reality. Finally, ICS-FT analyzes and records the numerous testing routes through instrumentation in the trace phase. The trace module is executed until the system crashes therefore the author is able to verify the weakness part. The architecture of the proposed scheme is illustrated in Fig. 3.

3.1 Seed Generation Module

ICS-FT acquires the first seed of a round from OPC service in fuzzing testing. Any of the conduits in ICS is determined as input data, such as RTU returning message to SCADA server.

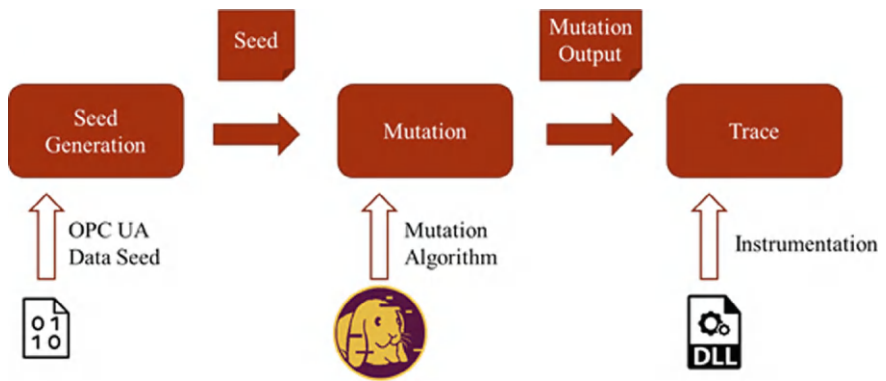


Fig. 3 The architecture of the proposed scheme

3.2 Mutation Module

The mutation module is used to produce diverse test cases based on AFL. ICS-FT mutates the seed using insertion, deletion, substitution, and flipping to modify the content and structure of the input.

3.3 Trace Module

The trace module implements instrumentation according to DLL injection of Windows server. The detail is described below:

Step 1. ICS-FT obtains OPC UA related process by applying OpenProcess function of Windows application programming interface (API), which is called “handle” under the operation system (OS) of Windows.

Step 2. OS allocates a memory space of virtual address space for ICS-FT by using VirtualAllocEx API, which is used to store the DLL of instrumentation. The size and allocation method are based on the testing demand.

Step 3. ICS-FT writes the DLL file included instrumentation into allocated memory space according to WriteProcessMemory API.

Step 4. OS creates a new thread for ICS-FT in the target process by applying CreateRemoteThread API, which the parameters are output from Step 1 to Step 3. If the system occurs the crash test case, it goes to step 4.1; otherwise, it goes to 4.2.

Step 4.1. ICS-FT marks the seed as crashed testing case, such as input data, condition of stack, error message, and timestamp.

Step 4.2. ICS-FT tests the next generated seed from mutation.

4 Experimental Results

In this section, we estimate the practicability of ICS-FT through a serial of simulation results. Here, the C/C++ language is applied to construct ICS-FT with a personal computer running Windows 10 64-bit. The computer is equipped with an Intel Core i7-4770 3.40 GHz with 8G RAM. The tool of AFL is applied by Win-AFL, which is used for Windows system. Regarding to DLL injector and instruction, the authors have adopted Hook API and Dynamorio to implement, respectively.

The experimental outcomes include two sections, the execution status and the coverage guided. The executed interface and data is shown in the Fig. 4, which consists of eight parts, such as process timing, cycle progress, stage progress, fuzzing strategy yields, map coverage, finding in depth, path geometry, and overall results.

Win-AFL+ DynamoRIO	Win-AFL+ ICS-FT
WinAFL 1.10b based on AFL 2.43b (test_gdip.exe) process timing run time : 0 days, 9 hrs, 10 min, 9 sec last new path : 0 days, 0 hrs, 0 min, 26 sec last uniq crash : none seen yet last uniq hang : none seen yet cycle progress now processing : 16 (72.73%) paths timed out : 0 (0.00%) stage progress now trying : arith 8/9 stage execs : 500/144 (35.36%) total execs : 26.1k exec speed : 55.74/sec (slow) fuzzing strategy yields bit flips : 17/1884, 0/1177, 0/1163 byte flips : 0/145, 0/141, 0/127 arithmetic : 5/7036, 0/1887, 0/550 known ints : 0/638, 3/3614, 6/4135 dictionary : 0/0, 0/0, 0/13 haves : 6/4037, 0/0 trials : 2.708/34, 0.00% map coverage map density : 3.32% / 4.51% count coverage : 1.55 bits/triple findings in depth favored path# : 10 (45.45%) new edges on : 12 (54.55%) total crashes : 0 (0 unique) total timeouts : 0 (0 unique) path geometry level# : 5 pending : 16 pend fav : 5 own finds : 21 imported : n/a stability : 71.84% overall results cycles done : 0 total paths : 22 uniq crashes : 0 uniq hangs : 0 [cpu00000]: 29%	WinAFL 1.10b based on AFL 2.43b (test_gdip.exe) process timing run time : 0 days, 0 hrs, 10 min, 0 sec last new path : 0 days, 0 hrs, 0 min, 9 sec last uniq crash : 0 days, 0 hrs, 4 min, 16 sec last uniq hang : none seen yet cycle progress now processing : 42 (93.3%) paths timed out : 0 (0.00%) stage progress now trying : havoc stage execs : 22/460 (4.78%) total execs : 122k exec speed : 223.4/sec fuzzing strategy yields bit flips : 4/4960, 0/4940, 3/4900 byte flips : 0/620, 0/600, 0/560 arithmetic : 5/34.5k, 0/10.1k, 0/6713 known ints : 0/2907, 5/16.2k, 14/19.5k dictionary : 0/0, 0/0, 0/333 haves : 16/14.1k, 0/0 trials : 24.686/186, 0.00% map coverage map density : 2.70% / 3.04% count coverage : 1.58 bits/triple findings in depth favored path# : 13 (28.89%) new edges on : 21 (46.57%) total crashes : 1 (1 unique) total timeouts : 0 (0 unique) path geometry level# : 12 pending : 26 pend fav : 1 own finds : 44 imported : n/a stability : 74.36% overall results cycles done : 0 total paths : 45 uniq crashes : 1 uniq hangs : 0 [cpu00000]: 15%

Fig. 4 The executed outcomes of ICS-FT

The description is depicted below. The authors carried out the experiments of the classical AFL and ICS-FT in Fig. 4. According to Fig. 4, the frame with red color illustrated the total paths of 22 and 45, which represent the classical AFL and ICS-FT, respectively.

- **Process Timing:** Process timing displays the time-related information of current fuzzing testing, such as run time, last crash time, and last hang time.
- **Cycle Progress:** Cycle progress includes information of now processing and the paths of timeout.
- **Stage Progress:** Stage progress includes information of stage execution progress, total execution progress, and execution speed.
- **Fuzzing Strategy Yields:** Fuzzing strategy yields includes information of mutation, such as the amplitude of bit flips and byte flips.
- **Map Coverage:** Map coverage provides the data of coverage, such as map density and count coverage.
- **Findings in Depth:** Findings in depth offers the testing details, such as the total number of paths and favorite path, crash cases, and timeout cases.
- **Path Geometry:** Path geometry includes the path-related data, such as the number of new path, testing level, and stability value.
- **Overall Results:** Overall results obtains the whole testing data, such as total number of discovered paths, done cycles, crashes, and timeout.

Regarding coverage guided, the authors had the experiments of the classical AFL and ICS-FT in Table 1. Both total execution times are 10 min. Obviously, the proposed scheme has a higher value than that of classical AFL, even more than two times. It is because that ICS-FT has employed dynamic instruction through the trace module, as shown in Sect. 3.3. The DLL injection is able to completely execute the instrumentation and analysis via entire memory accessing. Therefore, ICS-FT has properties of better efficiency.

Table 1 Comparisons between classical AFL and the proposed scheme

Method subject	Classical AFL	ICS-FT
Time	10 min	10 min
Path	22	45

5 Conclusions

In this research, the authors propose a novel ICS-oriented fuzzing testing based windows environment. ICS-FT integrates the DLL injection and Win-AFL to implement the dynamic instrumentation. Compared to the traditional AFL method, the proposed scheme has the higher cover guided value in the experiments. As to the future works, the authors will realize the simulation with comprehensive OPC UA server. This has ability to improve the mutation strategy. Moreover, it is feasible to accomplish in the operation field.

References

1. Rosen R, Wichert GV, Lo G, Bettenhausen KD (2015) About the importance of autonomy and digital twins for the future of manufacturing. IFAC-PapersOnline 48(3):567–572

2. Karmakar A, Dey N, Baral T, Chowdhury M, Rehan M (2019) Industrial internet of things: a review. In: 2019 international conference on opto-electronics and applied optics. pp 1–6

3. Yu ZH, Gao HX, Wang D, Alnuaim AA, Firdausi M, Mostafa AM (2022) SEI2RS malware propagation model considering two infection rates in cyber-physical systems. Phys A: Stat Mech Its Appl 597(1)

4. Yu Z, Wang H, Wang D, Li Z, Song H (2022) CGFuzzer: a fuzzing approach based on coverage-guided generative adversarial networks for industrial IoT protocols. IEEE Internet Things J 9(21):21607–21619

5. Lüders S (2011) STUXNET and the impact on accelerator control systems. In: The 13th international conference on accelerator and large experimental physics control systems. Grenoble, France, pp 1285–1288

6. Kumar M (2018) TSMC chip maker blames WannaCry malware for production halt. The Hacker News. <https://thehackernews.com/2018/08/tsmc-wannacry-ransomware-attack.html>

7. Elsad A, Gumarin JR, Barr A (2022) LockBit 2.0: how this RaaS operates and how to protect against it. UNIT 42 Ransomware, Threat Briefs and Assessments. <https://unit42.paloaltonetworks.com/lockbit-2-ransomware/>

8. Williams TJ (1994) The purdue enterprise reference architecture. Comput Ind 24(2–3):141–158

9. Maidl M, Kröselberg D, Christ J, Beckers K (2018) A comprehensive framework for security in engineering projects-based on IEC 62443. In: 2018 IEEE international symposium on software reliability engineering workshops (ISSREW). pp 42–47

10. Wang W, Chen L, Han L, Zhou Z, Xia Z, Chen X (2020) Vulnerability assessment for ICS system based on zero-day attack graph. In: 2020 international conference on intelligent computing, automation and systems (ICICAS). Chongqing, China, pp 1–5

11. Metzman J (2013) Going beyond coverage-guided fuzzing with structured fuzzing. Black Hat USA

12. Böhme M, Pham VT, Roychoudhury A (2019) Coverage-based Greybox fuzzing as Markov chain. IEEE Trans Software Eng 45(5):489–506

13. Pham VT, Böhme M, Santosa AE, Căciulescu AR, Roychoudhury A (2021) Smart Greybox fuzzing. *IEEE Trans Software Eng* 47(9):1980–1997
14. Li Y, Ji SL, Lyu CY, Chen Y, Chen JH, Gu QC, Wu CM, Beyah R (2022) V-Fuzz: vulnerability prediction-assisted evolutionary fuzzing for binary programs. *IEEE Trans Cybern* 52(5):3745–3756
15. Ko B, Choi W, Jeong D (2020) A Study on the tracking and blocking of malicious actors through thread-based monitoring. *J Korea Inst Inf Secur & Cryptol* 30(1):75–86
16. Abrams L (2018) Windows 10 ransomware. *Bleepingcomputer windows 10 ransomware protection bypassed using DLL injection*
17. Klein A, Kotler I (2019) Windows process injection in 2019. *Black Hat USA 2019*
18. Alasiri A, Zaidi MA, Lindskog D, Zavorsky P, Ruhl R, Alasmi S (2012) Comparative analysis of operational malware dynamic link library (DLL) injection live response versus memory image. In: *Conference: international conference on computing, communication and informatics management (ICCCSIM 2012)*
19. Ethernet POWERLINK Standardisation Group (2016) Ethernet POWERLINK communication profile specification
20. Fovino IN, Carcano A, Masera, Trombetta M (2009) Design and implementation of a secure modbus protocol. *Crit Infrastruct Prot* III 311:83–96
21. OPC Foundation (2017) OPC unified architecture-part 1: overview and concepts industry standard specification OPC. 10000–1
22. OPC Foundation (2017) OPC unified architecture-part 4: services. *Industry Standard Specification OPC*, pp 10000–4

Security Application

Video Access Control Aligned with Health Education Classification



Yi-Hui Chen  and Cheng-I Hsieh

Abstract Content authorization is a widely employed practice in health education categorization to prevent children from encountering inappropriate material. The conventional approach involves encrypting sensitive areas as nonsensical content, allowing decryption only when users possess authorized keys. Nevertheless, children should likely access these sensitive areas if their parents forget to log out of the system. To address this limitation, the proposed scheme introduces a real-time access control mechanism that employs online face recognition to determine whether users can access the content. Furthermore, the sensitive regions are encrypted using mask algorithm and decrypted by extracting facial features, reducing the maintenance cost associated with key management. Experimental results affirm the viability of the proposed scheme.

Keywords Visual cryptography · Fine-grained access control · Face recognition

1 Introduction

In today's digital age, an enormous volume of audio-visual content is uploaded daily to social media and streaming platforms, giving rise to security concerns. For instance, malicious actors can exploit segments of these videos to acquire sensitive personal information. To cite an instance, parents need to control violent or inappropriate content to prevent children from viewing it. Despite age-appropriate ratings for many violent videos, these measures often fall short of effectively managing user

Thanks to the technical support by Kingston Ka Ho Sze, Yue-Ling Huang, Yi-Ling Kuo, Li-Chin Lai and Yi-Chun Hsieh.

Y.-H. Chen (✉) · C.-I. Hsieh

Department of Information Management, Chang Gung University, Taoyuan 33302, Taiwan
e-mail: cyh@mail.cgu.edu.tw

Y.-H. Chen

Kawasaki Disease Center, Kaohsiung Chang Gung Memorial Hospital, Kaohsiung 83301, Taiwan

access based on permissions. However, these ratings may not adequately restrict access for underage individuals. To address this issue, a straightforward approach involves encrypting sensitive content segments, allowing people to view them only if they possess the necessary decryption keys.

In this paper, our proposed scheme aims to precisely detect violent scenes and regions within videos to customize content access according to user permissions. We employ the YOLOv8 and YOLOv5 (You Only Look Once) model [2] to identify these violent regions. To improve the efficiency of training the model, we employ transfer learning to aid in recognizing violent areas.

Furthermore, the existing access control mechanism verifies users' identities only once upon logging into the system. This poses a risk, as unauthorized content may still be accessible if parents or guardians forget to log out. We have integrated facial recognition technology based on OpenCV API [1] into our proposed scheme to bolster the system's security measures. To safeguard sensitive regions, we employ visual cryptography [12], a technique that divides secrets into multiple partitions, commonly called "shares" or "shadows." These shares are combined only when needed to reveal the secrets. Consequently, the original content is encrypted based on facial features, ensuring user identities are continuously verified in real time.

In the experiments, three public datasets [4, 5, 9] are conducted for violent image recognition, namely the Hockey Fight Dataset, Crowd Violence Dataset and AIRT-Lab Dataset. These datasets were used to train the model to identify violent areas within images, and the model's ability to detect these areas yielded an impressive detection rate of nearly 93%. This high detection rate undergirds the effectiveness of our violent object detection approach. The efficiency of our system can be attributed to the encryption method, which is enhanced through visual cryptography. Furthermore, our proposed scheme offers fine-grained access control by encrypting only the sensitive regions within the content. We have defined different viewing scopes based on user permissions, allowing for granularity as precise as a single pixel. Access restrictions are determined by the user's permission with the lowest level of access, ensuring that content viewing is tightly controlled and dependent on the user's specific permissions.

2 Related Works

This section briefly describes the concepts of violence detection and visual cryptography.

2.1 Violence Detection

In recent years, with the continual development of deep learning technologies, many models that employ deep learning to recognize violent scenes have achieved commendable accuracy in classifying such images. Among all the classification models

Table 1 The accuracy of learning methods, in detecting instances of violence in hockey fight and crowd violence scenarios [13]

Author	Architecture	Hockey fight (%)	Crowd violence (%)
Sudhakaran and Lanz [16]	2D CNN + ConvLSTM	97.1	94.5
Hanson et al. [8]	2D CNN + ConvLSTM	98.2	96.3
Sernani et al. [13]	ConvLSTM	96.6	84.2
Ding et al. [7]	3D CNN	91.0	–
Song et al. [15]	3D CNN	99.6	94.3
Li et al. [11]	3D CNN	98.3	97.2
Ullah et al. [17]	C3D + Fully connected layers	96.0	98.0
Accattoli et al. [3]	C3D + SVM	98.5	99.2

that have emerged thus far, the primary structures are categorized into 2D CNNs network architecture, 3D CNNs network architecture, and ConvLSTMs network models. Table 1 enumerates prior studies that have leveraged deep learning models to identify violent scenes, specifically conducting experiments on the Hockey Fight and Crowd Violence datasets to test their accuracy.

Furthermore, previous research has indicated that deep learning methodologies can effectively extract spatio-temporal features from video [18, 19], capturing more dynamic information across continuous frames. The publicly available datasets commonly used in past studies include lower-resolution ones such as the Hockey Fight dataset and Crowd Violence dataset, as well as a dataset with various resolutions proposed recently by Cheng et al. [6]. Paolo Sernan et al. [13] introduced a high-resolution dataset with 1920×1080 pixels. Within the Hockey Fight dataset, there are a total of 1000 clips, evenly divided into “fight” and “non-fight” categories, each ranging between 41 and 50 frames [15]. The original image resolution is 720×576 pixels, with the 320×240 pixels version being more prevalent [14]. The clips are extracted from professional hockey match footage. The Crowd Violence dataset consists of 246 pins, with 123 labeled as “violence” and 123 as “non-violence,” all sourced from YouTube. Each clip has a resolution of 320×240 pixels and an average length of 3.6s, classifying these two datasets as low resolution. In addition, a recent author introduced the high-resolution violent image detection dataset AIRTLab, containing 350 clips, divided equally into “violence” and “non-violence” categories. Each clip averages 5.6s in length and has a 1920×1080 pixels resolution. Moreover, to enhance the accuracy of identifying non-violent actions that have previously been easily misclassified as violent, such as hugging, high-fiving, and waving the AIRTLab dataset includes many false-positive clips to aid in researching violent imagery detection. This study seeks to amalgamate previous deep learning-based violent image detection model research with image encryption technology to protect sensitive information within images.

2.2 Visual Cryptography

Visual Cryptography (VC) is a long-established technology called Visual Secret Sharing Scheme (VSS), a form of secret sharing. Naor and Shamir first proposed it in 1994 [12] Fig. 1. The basic concept of visual cryptography is to divide a remote image into two or more image fragments. When viewed separately, these fragments reveal no information about the original image. The original image can only be decrypted and displayed when a sufficient number of these shares are stacked together. Unlike traditional methods that require extensive cryptography calculations for decryption, this method requires only minimal computations for decoding.

Visual cryptography technology has since been widely researched and applied, leading to various developments. For instance, a study published in 2003 by Hou [10] extended visual cryptography techniques to color images. Unlike previous research focused only on black-and-white and grayscale images, the author used color decomposition methods to propose a new encryption algorithm for color images. First, the color image was decomposed to C, M, and Y images, and then halftone technology was used to convert these images into halftone images. Combining these three halftone images generated a color halftone image, which maintained the color quality of the original image, thus enhancing the feasibility of encrypting color images. Finally, a security analysis of the color image encryption algorithm was conducted to ensure the safety of hiding image information.

Pixel	Probability	Share ₁	Share ₂	Share ₁ ⊗ Share ₂
	50%			
	50%			
	50%			
	50%			

Fig. 1 Naor and Shamir’s scheme for encoding a binary pixel into two shares

3 Methodology

3.1 System Design

As mentioned in the previous sections of the introduction and related works, we focus on detecting violent scene areas and controlling user access permissions. We design a system workflow to include these functionalities, as illustrated in the Fig. 2. The system consists of five phases: User Facial Recognition, User Permission Control, Violent Image Area Detection and Extraction, Violent Area Encryption, and Legitimate User Key Matching.

- (1) User Facial Recognition: When a user is viewing any video content, real-time user permission information is needed. We use the OpenCV facial recognition API to identify the user.
- (2) User Permission Control: Different keys are assigned to different users based on the facial recognition results. By having these unique keys, we can control user permissions effectively. For instance, keys owned by adults and minors will have different decryption capabilities on the videos.
- (3) Violent Image Area Detection and Extraction: Initially, we train our object detection model based on the YoloV8 model. We use the Hockey Fight Dataset and the Crowd Violence Dataset for annotation and training, effectively identifying the violent areas. Detected areas are then extracted for further processing.
- (4) Violent Area Encryption: The extracted violent areas are encrypted using visual cryptography.

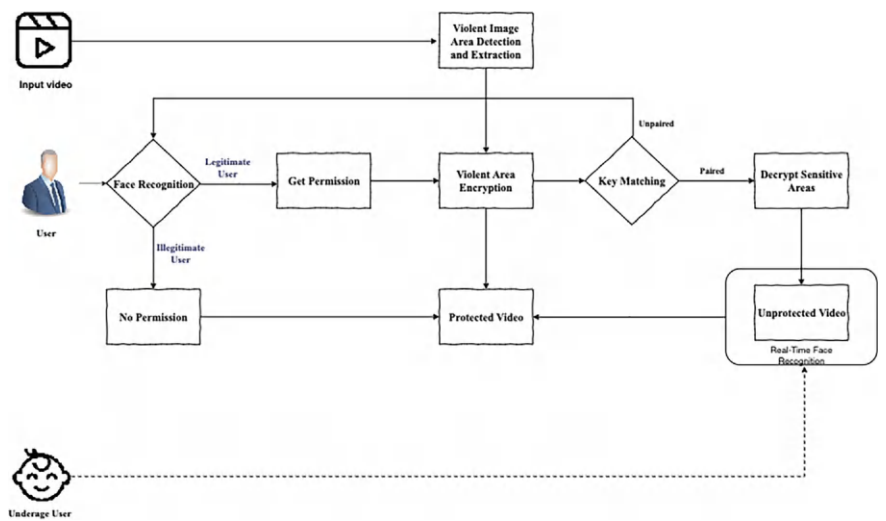


Fig. 2 The proposed system architecture

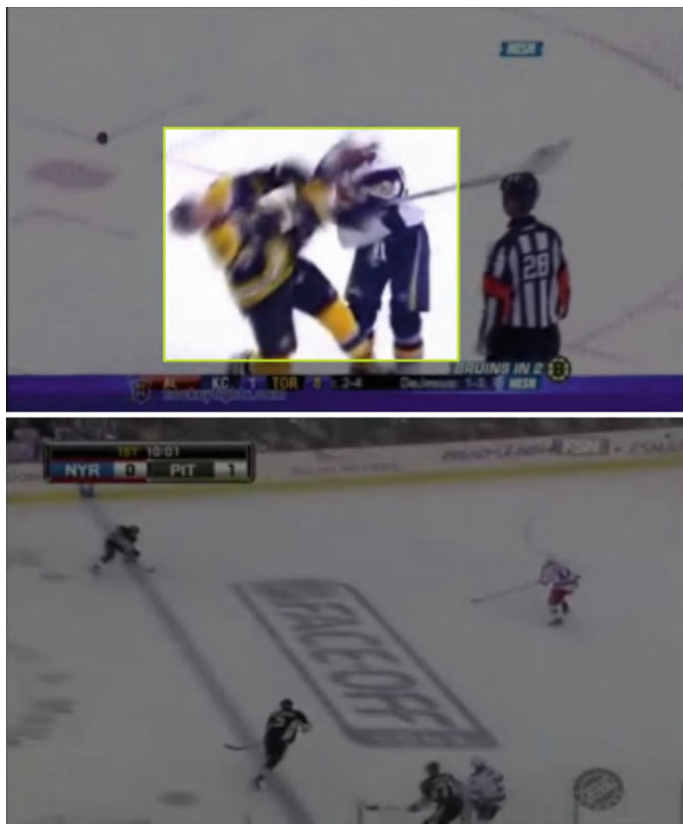


Fig. 3 Images are labeled as “Violence” and “Null” in the dataset

- (5) Legitimate User Key Matching: The keys the users own are used for secret sharing with the encrypted areas. By stacking the images, the original content be restored.

3.2 Violent Area Detection

To accurately extract the Violent Area, we employed the YOLOv8 model to train a classifier capable of distinguishing between violence and non-violence. First, we use Roboflow annotation tools to label three public datasets. In the images, areas featuring violence are tagged as “Violence,” if no violent regions are present, they are marked as “null”. As shown in Fig. 3, images are labeled as “Violence” in the dataset are demonstrated. A total of 6042 images were annotated across the two categories, with 3941 labeled as “Violence” and 2101 marked as “Null.”

Next, we train our model on Google Colab, with the GPU accelerator Nvidia Tesla T4, using Ultralytic. The original dataset is split into training set, validation set, and test sets in a 7:2:1 ratio. To compensate for any potential inadequacies in the dataset, we augment it to 14,459 images using data augmentation techniques. We inspect the loss value during each training epoch (initially set to 100 epochs). If the loss value does not decrease in five consecutive epochs, we stop the training process. The model with the lowest loss value is then selected as the final detection model. By employing these strategies, we aim to develop a robust and accurate model for violent area detection, ensuring that it performs well in real-world applications.

3.3 Image Encryption

For a given image I , the original pixel value, depicted as p_i , can be separated into three bit-planes R, G, and B, which are represented as a bit stream B_R , B_G , and B_B , respectively. Each bit-plane can be described as $B_x = b_1b_2b_3b_4b_5b_6b_7b_8$, where i indicates i th pixel in the x bit-plane in image I and x is mapped to R , G or B plane. The given pixel B_x can be separated into two shares S_1 and S_2 according to the encryption policy as Eq. (1), where the random number r_j is generated with the secret key belonging to the user and $S_1 = \bigcup_{j=1}^8 r_j$, $S_2 = \bigcup_{j=1}^8 s_j$, and j is from 1 to 8.

$$s_j = b_j \oplus r_j. \quad (1)$$

4 Experimental Evaluation

In this chapter, we will utilize the methods mentioned in Chap.3 for training and experimentation. Standard evaluation metrics will be employed to assess the feasibility of the proposed plans.

We trained models based YOLOv8 versions. We then evaluated these models using standard performance metrics such as precision, recall, and mAP (Mean Average Precision). These metrics can be expressed using True Positives (TP), True Negatives (TN), False Positives (FP), False Negatives (FN), Average Precision (AP), and Intersection Over Union (IOU) as follows Eqs. (2)–(5):

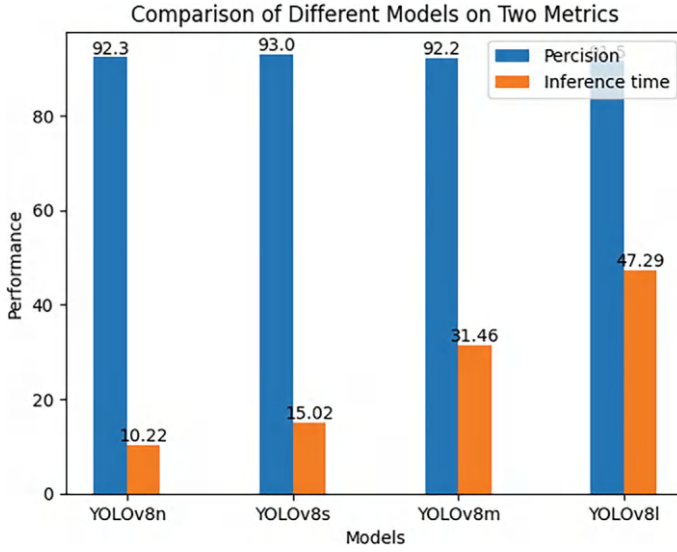


Fig. 4 YOLOv8 model accuracy and inference time

- Precision: The ratio of True Positives among all the predicted positives.
- Recall: The ratio of True Positives among all actual positives.
- AP (Average Precision): Refers to the average precision value across the precision-recall curve. For this experiment, we set the IOU threshold to be greater or equal to 0.5.
- mAP (Mean Average Precision): Refers to the mean of the AP across all categories.
- IOU (Intersection Over Union): Measures the degree of overlap between the bounding box and the ground truth. It is the ratio of the area of overlap to the area of the union of the two regions.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (2)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3)$$

$$\text{mAP} = \frac{1}{N} \sum_{i \in N}^N \text{AP}_i \quad (4)$$

$$\text{IOU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \quad (5)$$

Table 2 Violent detection accuracy without data augmentation for YOLOv8 each model

	Precision (%)	Recall (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
YOLOv8n	89.7	83.3	91.2	62.3
YOLOv8s	90.5	83.3	90.9	62.2
YOLOv8m	87.9	84.7	90.8	62.3
YOLOv8l	88.9	83.1	90.9	60.9
YOLOv8x	87.4	81.8	89.8	60.2

Table 3 Violent detection accuracy with data augmentation for YOLOv8 each model

	Precision (%)	Recall (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
YOLOv8n	92.3	82.2	90.5	64.3
YOLOv8s	93	83.9	91.1	66
YOLOv8m	90.5	85.5	90.7	65.6
YOLOv8l	91.5	82.1	90.2	65.3
YOLOv8x	89.7	85.8	90.8	64.8

In our research, based on the aforementioned evaluation metrics, we compared models that were trained with and without dataset augmentation. We focused on each YOLOv8 models, with Fig. 4 showing the number of parameters in each model. These models, from smallest to largest, are YOLOv8n, YOLOv8s, YOLOv8m, and YOLOv8l. Tables 2 and 3 illustrate that models trained with augmented datasets displayed a minimum 2.5% increase in percision, with the YOLOv8l model showing the most significant improvement of 4.1%. Additionally, there was a general rise in mAP.5:95, signifying that models trained with data augmentation achieved improved accuracy in predicting targets (violent objects). The improvement in mAP indicates a considerable enhancement in model performance at different IOU thresholds (IOU >= 50, 55, 60...95), thereby expanding the model’s generalization. However, due to the YOLOv8x model’s extensive parameter count and computational demands, it requires a significant amount of GPU memory. Unfortunately, our current equipment is not capable of handling the training of this model.

Furthermore, we examined the inference time of each model. Using a GPU T4, we calculated the average time taken for inference by each model, as indicated in Fig. 4. From these experimental results, we discovered that the YOLOv8s model had the highest accuracy, and its inference time was measured in 15 ms. This analysis emphasizes the balance between model complexity, accuracy, and computational efficiency, which is crucial for practical applications. As shown in Fig. 5, the model successfully extracts the violent areas from the images.

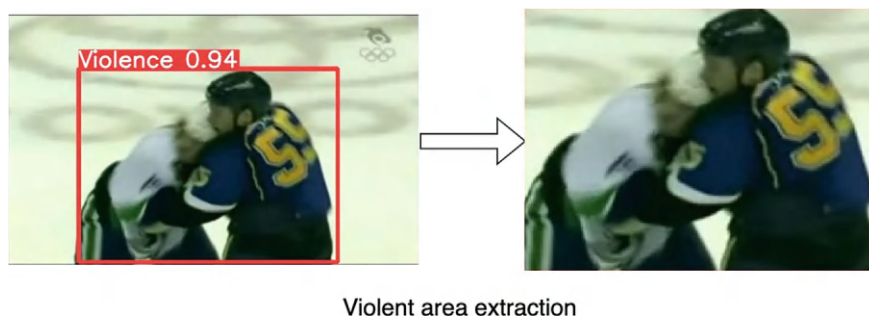


Fig. 5 Using YOLOv8 model extracts violent areas

5 Conclusion

The paper introduces a sophisticated fine-grained access control system designed for videos. This system allows users to access distinct perspectives of the same videos based on their assigned permissions. A VSS method is also incorporated into the encryption process, enabling users to decrypt sensitive areas with minimal computational burden. The experiment demonstrates the proposed approach's viability in delivering authorized views to users under their permissions.

The model has been deployed to classify health education materials catering to adult and underage users. The model will incorporate an access control framework in its extended version to accommodate multiple permission assignments. This enhancement will permit users with encrypted video content to decrypt specific video segments under their supplementary permissions. Furthermore, the scheme mandates user identity verification and the decryption of authorized video segments during playback, potentially resulting in delays. Consequently, optimizing the model's efficiency is a paramount concern that must be addressed in the next phase of the proposed scheme.

Acknowledgements The work was supported in part by the Ministry of Science and Technology of the Republic of China, Taiwan, under Grant MOST 110-2221-E-182-026-MY3, and in part by the Kaohsiung Chang Gung Memorial Hospital, with grant number CMRPD3N0011.

References

1. OpenCV: Face recognition with OpenCV. https://docs.opencv.org/3.4/da/d60/tutorial_face_main.html
2. What is yolov8? the ultimate guide (2023). <https://blog.roboflow.com/whats-new-in-yolov8/>
3. Accattoli S, Sernani P, Falcionelli N, Mekuria DN, Dragoni AF (2020) Violence detection in videos by combining 3d convolutional neural networks and support vector machines. *Appl Artif Intell* 34(4):329–344

4. Bermejo Nievas E, Deniz Suarez O, Bueno García G, Sukthankar R (2011) Violence detection in video using computer vision techniques. In: Computer analysis of images and patterns: 14th international conference, CAIP 2011, Seville, Spain, August 29–31, Proceedings, Part II 14. Springer, pp 332–339
5. Bianculli M, Falcionelli N, Sernani P, Tomassini S, Contardo P, Lombardi M, Dragoni AF (2020) A dataset for automatic violence detection in videos. Data Brief 33:106587
6. Cheng M, Cai K, Li M (2021) RWF-2000: an open large scale video database for violence detection. In: 2020 25th international conference on pattern recognition (ICPR). IEEE, pp 4183–4190
7. Ding C, Fan S, Zhu M, Feng W, Jia B (2014) Violence detection in video by using 3d convolutional neural networks. In: Advances in visual computing: 10th international symposium, ISVC 2014, Las Vegas, NV, USA, December 8–10, Proceedings, Part II 10. Springer, pp 551–558
8. Hanson A, Pnvr K, Krishnagopal S, Davis L (2018) Bidirectional convolutional LSTM for the detection of violence in videos. In: Proceedings of the European conference on computer vision (ECCV) workshops, pp 0
9. Hassner T, Itcher Y, Kliper-Gross O (2012) Violent flows: real-time detection of violent crowd behavior. In: 2012 IEEE computer society conference on computer vision and pattern recognition workshops. IEEE, pp 1–6
10. Hou YC (2003) Visual cryptography for color images. Pattern Recogn 36(7):1619–1629
11. Li J, Jiang X, Sun T, Xu K (2019) Efficient violence detection using 3d convolutional neural networks. In: 2019 16th IEEE international conference on advanced video and signal based surveillance (AVSS). IEEE, pp 1–8
12. Naor M, Shamir A (1995) Visual cryptography. In: Advances in cryptology—EUROCRYPT’94: workshop on the theory and application of cryptographic techniques Perugia, Italy, May 9–12, 1994 Proceedings, vol 13. Springer, pp 1–12
13. Sernani P, Falcionelli N, Tomassini S, Contardo P, Dragoni AF (2021) Deep learning for automatic violence detection: Tests on the airtlab dataset. IEEE Access 9:160580–160595
14. Serrano Gracia I, Deniz Suarez O, Bueno Garcia G, Kim TK (2015) Fast fight detection. PloS One 10(4):e0120448
15. Song W, Zhang D, Zhao X, Yu J, Zheng R, Wang A (2019) A novel violent video detection scheme based on modified 3d convolutional neural networks. IEEE Access 7:39172–39179
16. Sudhakaran S, Lanz O (2017) Learning to detect violent videos using convolutional long short-term memory. In: 2017 14th IEEE international conference on advanced video and signal based surveillance (AVSS). IEEE, pp 1–6
17. Ullah FUM, Ullah A, Muhammad K, Haq IU, Baik SW (2019) Violence detection using spatiotemporal features with 3d convolutional neural network. Sensors 19(11):2472
18. Wang L, Xu Y, Cheng J, Xia H, Yin J, Wu J (2018) Human action recognition by learning spatio-temporal features with deep neural networks. IEEE access 6:17913–17922
19. Zhang L, Zhu G, Shen P, Song J, Afaq Shah S, Bennamoun M (2017) Learning spatiotemporal features using 3dCNN and convolutional LSTM for gesture recognition. In: Proceedings of the IEEE international conference on computer vision workshops, pp 3120–3128

Information and Public Opinion Analysis Platform Based on Social Media



Han-Ting Gao, Shang-Yi Yang, Chun-Han Chang, Jung Kuo,
Guan-Yu Chen, Jheng-Jia Huang, and Nai-Wei Lo

Abstract In the modern age of rapid global information circulation, public opinion on social media reflects differences in people's opinions on a certain issue. These opinions can include emotions, likes and dislikes, etc. Once an incident happens, many social media gather a lot of discussions. Such situations let us to analyze the public's perception and preferences through public opinion analysis. Moreover, people cannot live without food, thus our team takes gourmet restaurants as an entry point to develop a public opinion analysis platform for gourmet restaurants. This study uses Python web crawler technology to capture data from public websites, such as PTT, Dcard, and Google Maps user comments. Then we extract public opinion data in the last year, clean and sort the data, and combine the sentiment analysis tool SnowNLP, analyze the positive and negative emotions of users' comments in various restaurants. And we screen out the top popular restaurants in the form of scores, build a website entry function so that users can query and view the names,

H.-T. Gao · S.-Y. Yang · C.-H. Chang · J. Kuo

Management Undergraduate Program, National Taiwan University of Science and Technology,
Taipei, Taiwan

e-mail: B10933019@mail.ntust.edu.tw

S.-Y. Yang

e-mail: B10933037@mail.ntust.edu.tw

C.-H. Chang

e-mail: B10933032@mail.ntust.edu.tw

J. Kuo

e-mail: B10933005@mail.ntust.edu.tw

G.-Y. Chen (✉) · J.-J. Huang · N.-W. Lo

Department of Information Management, National Taiwan University of Science and Technology,
Taipei, Taiwan

e-mail: d11209103@mail.ntust.edu.tw

J.-J. Huang

e-mail: jhengjia.huang@mail.ntust.edu.tw

N.-W. Lo

e-mail: nwlo@cs.ntust.edu.tw

ratings of various restaurants, and emotional scores sorted out by our team. We hope this paper can be used as a reference to assist individual decision-making.

Keywords Data analysis · Public opinion · Sentiment analysis · Web crawler · Social media

1 Introduction

1.1 Research Background

Big data is the trend of the times, and the ubiquity of vast amounts of information has driven modern individuals to constantly seek more convenient and reliable sources of information and ways to access it. Even the average person can obtain data through free databases or web scraping, making data science accessible to all. The key challenge lies in how to effectively use data to discover and solve problems, and building automated and analytical templates using various programming languages and machinery is undoubtedly the future trend.

With transportation becoming more convenient, people's living spheres are gradually expanding. When faced with restaurants they have never been to or tried before, people find it increasingly difficult to resist the temptation to refer to online restaurant reviews. Online ratings significantly influence people's preferences for restaurants. Even if a restaurant doesn't have an attractive appearance, people may gather the courage to try it out due to high ratings. Conversely, a restaurant that seems busy and appealing may cause hesitation when seeing lower ratings or negative reviews.

But can those five-star ratings truly represent people's real thoughts about the restaurant? Restaurant owners are well aware of how much customers value reviews, so they often offer various incentives to earn five-star ratings from diners. Some even hire individuals to write positive reviews to control public opinion.

So how can we determine whether these reviews are genuine or fake? Even if people give high ratings against their true feelings, they may sometimes leave negative comments or comments without explicit praise in the review section. Therefore, we can use "sentiment analysis" [1] tools to discern whether a review is a "valid positive review." We quantify and standardize the sentiment values and then compare them with the ratings displayed on the website to confirm whether the positive reviews are genuine and whether the restaurant has received widespread praise.

1.2 Research Motivation

Taipei City is the capital of Taiwan, known for its numerous and diverse dining establishments. Imagine having a comprehensive website that provides a reference

point from various angles. For example, it could list the most popular restaurants from different review websites, displaying their star ratings and sentiment analysis scores. This would enable users of the website to evaluate restaurants based on an alternative scoring system.

In this context, we have chosen three large websites that people often search for restaurant or food-related information and use reviews and comments as references. These websites are selected based on their “interactivity” (the ability to rate a restaurant in a specific way) and “representativeness” (popularity in food reviews). The selected websites are:

1. Google Maps, where users can rate restaurants on a map, including providing star ratings and comments.
2. PTT Food Forum, an anonymous discussion board mainly focused on restaurant food reviews, featuring a liking/disliking system.
3. Dcard, a platform where users display their school names or nicknames and use the number of hearts as a measure of an article’s popularity.

This website caters to individuals who want to explore different rating systems and use them as a reference to discover new popular restaurants (based on discussion volume). It targets young people who frequently use social media and have an interest in current food trends. While regularly collecting new data using web scraping techniques, the website also provides real-time popularity rankings for reference.

In essence, this website serves as a valuable tool for those seeking diverse perspectives on restaurant ratings and a way to discover the latest culinary trends in Taipei.

2 Theory and Practice

2.1 Public Opinion Analysis

“Public opinion,” also referred to as “public sentiment” or “public sentiment analysis,” refers to the collective thoughts of the public, reflecting common opinions and ideas. With the advancement of technology and the progress of time, public opinion is increasingly conveyed through the internet. The aggregation of public opinion can be analyzed through various techniques to appropriately classify and process different opinions, ultimately producing analysis results that help us interpret, utilize, and make decisions. This process is known as sentiment analysis.

The process of sentiment analysis can be divided into four main steps:

1. Collection: Gather relevant data from various sources, such as social media, news reports, blogs, etc.
2. Filtering: Use keyword filters to sift through data, categorize it into different types, and remove duplicate information.

3. Analysis: Determine the direction of analysis based on research objectives and data characteristics, such as volume, sentiment, and trends.
4. Presentation: Organize the analysis results and generate corresponding reports, including textual explanations and visualizations, to assist users in decision-making.

Sentiment analysis is a valuable tool for understanding and interpreting public sentiment, which can be crucial for businesses, governments, and organizations to make informed decisions and respond effectively to public sentiment.

2.2 Sentiment Analysis

In general, when we need to make decisions in our daily lives, we often seek advice from friends around us or browse relevant discussions on social media. For instance, during mealtime, we might ask friends for their personal restaurant recommendations, which can serve as references for our own choices. However, the most common approach is to open social media and check out the restaurants that are currently generating the most buzz. If the comments below are consistently positive and there are fewer negative comments, most users tend to prioritize that restaurant.

Emotion analysis, or sentiment analysis, utilizes techniques such as natural language processing and text mining [2] to determine whether data or text carries a positive, neutral, or negative sentiment. The aim is to use computers to objectively assess the prevailing emotions of humans. Emotion analysis is often applied on the business end to observe whether social media users hold a positive or negative attitude toward a company's products. When the sentiment is negative, businesses can quickly address product flaws based on consumers' negative feedback. Our team aims to use emotion analysis to evaluate consumers' reviews of restaurants, providing sentiment scores through the underlying NLP technology behind computers.

3 Research Methods and Procedures

3.1 Information System Development

Information system construction refers to the design, development, and implementation of an information system. It is an integrated and interconnected combination of hardware, software, databases, and more, designed to collect, process, store, and transmit information with specific objectives in mind. The process involves determining system requirements, designing system architecture, selecting appropriate technologies and platforms, developing system functionality, and conducting testing and validation. Information system construction considers multiple factors, including development processes, data requirements, scalability, and user experience.

The steps involved in constructing the Taipei Food Sentiment Information Platform for this research are as follows:

1. **Data Collection:** The research focuses on Taipei's food scene. Relevant online data is collected from social media platforms using keywords related to Taipei and food. Web pages are constructed, web data is crawled, and data is cleaned and processed, including volume analysis and sentiment analysis.
2. **Integration:** The results of frontend webpages and back-end data analysis are integrated to create a convenient information system for the public to access information about the hottest food restaurants and dishes in Taipei.
3. **Data Sources:** Dcard, Google Maps, and PTT are chosen as data sources based on factors like browsing frequency, user engagement (likes, dislikes, comments), and representativeness of the platforms. These elements reflect the participation rates of the general public on the platforms.
4. **Platform Segmentation:** The information system separates and presents popular food-related information from each community separately. This separation is based on the fact that Dcard, Google Maps, and PTT each have different target audiences.

3.2 Research Process

Our team initiated the process from the back-end by scraping user comments from PTT, Dcard, and Google Maps using web crawling techniques. Subsequently, we employed SnowNLP for sentiment analysis and calculated sentiment scores, giving weight to the analysis results. On the other hand, we developed a website to present the analysis results in a user-friendly and easily comprehensible manner. This integration of the front-end and back-end aims to provide users with a platform that assists them in decision-making by displaying the sentiment scores and other relevant information in a clear and accessible way (Fig. 1).

4 Research Results

Based on the various theories previously elaborated, our team combined the front-end and backend to build a public opinion analysis platform in the form of a website, and presented it on a website named "DElicacy". The English word DELicacy itself means delicacies. Here we integrate the Chinese character "北" into the web page logo, conveying the close connection between "Taipei" and "food". The following will explain the front-end and backend parts of the platform in detail.

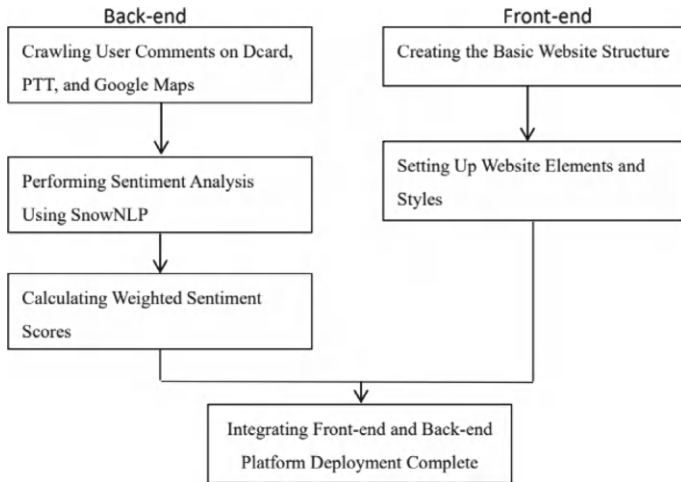


Fig. 1 Research process diagram for this platform

4.1 Front-End

1. Social Media Platforms Overview

This page displays the restaurant’s popularity rankings from three social media platforms: PTT, Dcard, and Google Maps. Our team aims to provide users with a clear overview, so we are only listing the top five from each platform here, along with concise information. If users wish to delve deeper into specific social media platforms or restaurant details, they can click the “More” button to access the detailed rankings (Fig. 2).

2. Popularity and Sentiment Scores

In comparison to point (2), this page presents more comprehensive restaurant information, including the top ten restaurants in terms of popularity ranking. Within the webpage, users can browse through rankings, restaurant names, addresses, popularity metrics (PTTs-total upvotes/downvotes, Dcard-total likes, Google Maps-total reviews), sentiment scores (max score of 1), and Google Maps star ratings (max score of 5).

Additionally, to help users quickly and clearly assess restaurant ratings, our team has included small icons next to the scores, using different colors (red–orange–yellow–green) to represent the quality of the restaurant’s ratings (Fig. 3).

3. Introduction to Dcard Web Scraping

Dcard is one of the most popular social media platforms among young people today. Users often share their thoughts on various topics or aspects of their lives on this

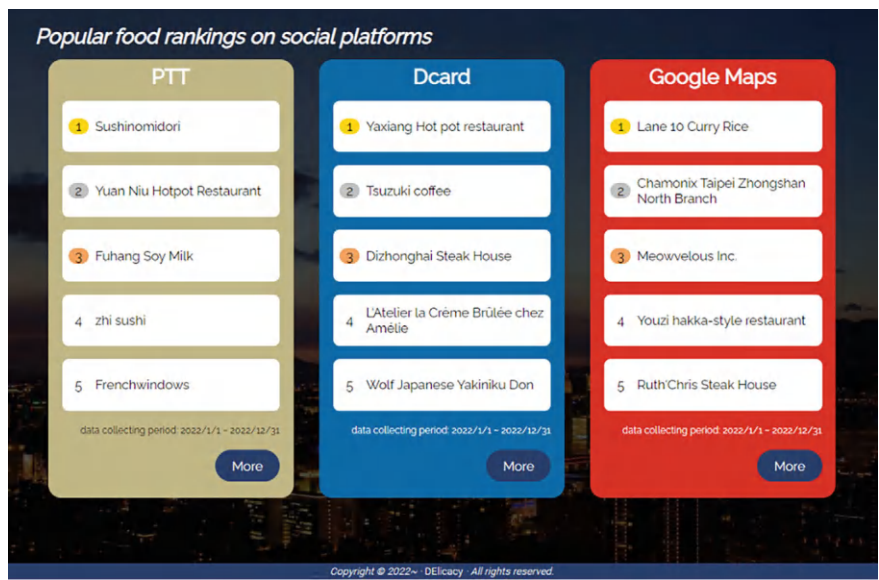


Fig. 2 Community categories overview screen of this platform

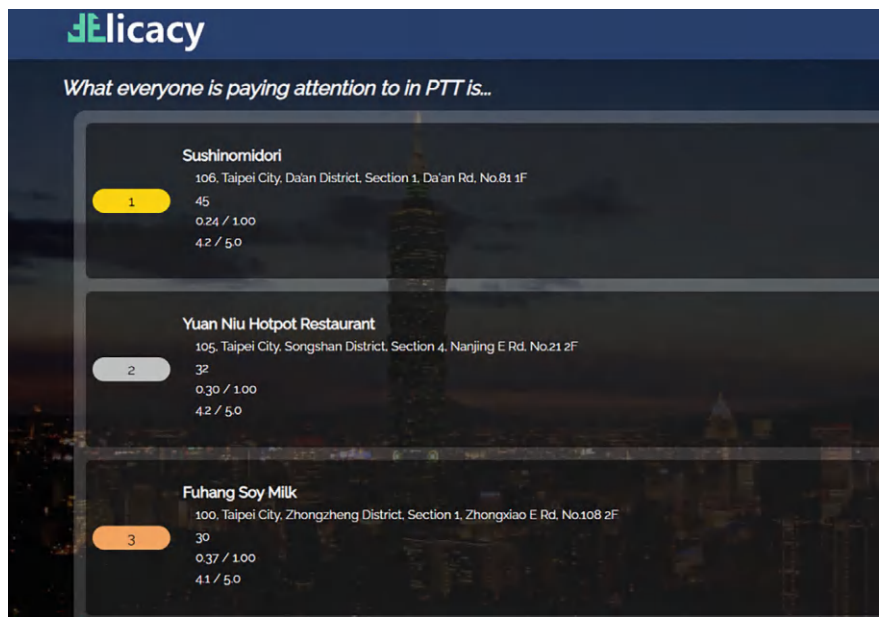


Fig. 3 Community rankings screen on this platform-PTT

platform, leading to frequent interactions and discussions through comments. Therefore, our team decided to include Dcard as a reference indicator when creating the Taipei Food Public Opinion Analysis Platform.

This research utilizes Python for web crawling. Firstly, it searches for pages related to keywords such as “Taipei,” “Northern,” “food,” and “restaurant.” Then, it accesses the URLs using the `get()` method from the BeautifulSoup and requests libraries. Afterward, the `find_all()` method is used to retrieve webpage data, extracting the title, number of likes, and URL from the web page.

4. Data Organization Results

Sorting the data by the number of likes in descending order, we have organized the top ten popular articles. Subsequently, we have extracted the names of restaurants from the content of these ten popular articles, and after integrating them with the front-end, they have been added to the rankings.

5. Introduction to PTT Web Scraping [3, 4]

PTT is an anonymous electronic bulletin board system (BBS) that was originally intended for academic discussions but has since evolved into various topic-specific boards. It allows people to freely express their opinions online. The user demographic on PTT tends to be concentrated in the age group of 30–39, and the platform features mechanisms such as “hotness,” upvotes/downvotes (positive/negative ratings), and comments to facilitate discussion and indicate the popularity of posts.

We also use a Python program to initially crawl the HTML of each page, then utilize web scraping to obtain the links to each individual article, and subsequently scrape structured data. This data includes the main post, author, title, timestamp, as well as categorizing comments into upvotes, downvotes, and author replies.

6. Data Compilation Results

As for the calculation of the article’s popularity, it’s derived from the total number of likes and dislikes in the comments. After sorting this data, it’s placed into an Excel file. However, manual filtering is required to remove instances where the same person posted multiple times or transferred posts in order to finalize the rankings.

Some articles receive high popularity mainly due to negative reviews, and a selection process is needed. We choose to retain those articles because we are selecting the “most discussed restaurants.” However, for articles related to restaurants that are difficult to make reservations for, there might be instances of transferred posts. To avoid this, we added the “dining brief” tag in the keywords section.

7. Introduction to Google Maps Web Scraping

Google Maps is a web-based mapping service provided by Google. It offers detailed electronic maps for locations all around the world. One of its notable features is the landmark rating and review system, making it one of the most popular review websites globally. The rating system typically ranges from 1 to 5 stars, and customer reviews provide valuable insights and references for various businesses and places.

We use the Chrome driver to simulate the scrolling action that a human would perform. We start by searching for restaurants in Taipei City using keywords and then scroll down to the bottom of the page to load all the restaurants.

Next, in the first round of web scraping, we first extract information such as the restaurant name, the number of reviews, and the URL of the reviews. Then, we filter and select the top ten restaurants based on the number of reviews.

8. Introduction to Sentiment Analysis

The process involves organizing the top ten restaurants selected from the three major social media platforms mentioned earlier. It also includes scraping articles and comment content associated with these restaurants. This scraping process is typically performed using web scraping libraries like BeautifulSoup and requests.

To achieve this, the program utilizes a looped code structure that navigates to the articles related to the target restaurants and collects the comments within those articles. This allows for the systematic gathering of comment data from the specified sources.

The comments collected by the program are written into a database, and they are organized and categorized based on different platforms and different restaurants. This organization and categorization facilitate the subsequent sentiment analysis processes.

Next, we read the previously organized comment database and performed sentiment analysis on each restaurant's comments. The computer objectively determined the emotional category of each comment, with a maximum score of 1 as the standard. If the sentiment score of a comment was greater than 0.6, we classified it as a positive comment. Conversely, if the score was less than 0.4, it was categorized as a negative comment. If the score fell between 0.4 and 0.6, the comment was classified as neutral.

The program then repeated this scoring process in a loop for each comment, tallying the total number of positive and negative comments for each restaurant. Finally, this sentiment score data was written into the database for further calculations, including the calculation of the overall average sentiment score.

5 Conclusion

Our team hopes to use sentiment analysis and crawler technology to build a food and restaurant public opinion analysis platform. We hope to contribute to the future related practices and discussions through this project, and achieve the effect of inspiring others. Theoretically, we collect the comments of users on social media websites, filter, analyze, and score the collected data through sentiment analysis, and classify them according to different platforms. We rank the three platforms and hope that users can choose according to their own preferences. Practically, our team presents the public opinion analysis platform to general users in order to assist their decision-making. The team analyzes and scores the sorted comments with emotional tools, and

ranks them according to the popularity of the restaurants. And combine it with the public opinion analysis platform. We hope that with a simple and intuitive website, users can clearly understand the popularity ranking and popularity of the restaurant.

However, after researching this project, we still found some details that need to be taken into consideration. For example, we considered it from the perspective of objective computer analysis, and did not start from the perspective of users [5]. We think that some restaurant names cannot be directly judged by the computer as what kind of dishes they are. They must be identified through tedious manual judgment. This kind of manual operation is somewhat contrary to our idea of objective analysis through the computer. On the other hand, we did not add article links to the pages of each restaurant. The current design is that if users are interested in the restaurant, they can search on each platform separately. If this project is commercially operated in the future, these problems need to be discussed and improved by our team. We look forward to this platform being able to truly help the public.

Acknowledgements This work was partially supported by the National Science and Technology Council of Taiwan, under grants NSTC 112-2221-E-011-092-MY2, NSTC 112-2634-F-011-002-MBK, NSTC 112-2221-E-011-094-MY2, and NSTC 112-2218-E-011-008-.

References

1. Gauri DK, Bhatnagar A, Rao R (2008) Role of word of mouth in online store loyalty. *Communication of the ACM* 51(3):89–91.
2. Medhat W, Hassan A, Korashy H (2014) Sentiment analysis algorithms and applications: a survey. *Ain Shams Engineering Journal* 5(4):1093–1113
3. Blake C (2011) Text mining. *Ann Rev Inf Sci Technol* 45(3):121–155
4. Thomas DM, Mathur S (2019) Data analysis by web scraping using python. In: 2019 3rd international conference on electronics, communication and aerospace technology (ICECA). IEEE, pp 450–454
5. Khder MA (2021) Web scraping or web crawling: state of art, techniques, approaches and application. *Int J Adv Soft Comput & Its Appl* 13(3)
6. Espenshade TJ, Calhoun CA (1993) An analysis of public opinion toward undocumented immigration. *Popul Res Policy Rev* 12:189–224

Analysis of the Domain Name of Fraud Investigation Sites with the Lexical Entailment Recognition Methods



Yeu-Pong Lai

Abstract Fraudulent crimes have been a major concern in criminal patterns for a long time. Investment fraud, in particular, has become increasingly rampant in recent years, since it causes substantial financial losses in a single case. Fraud gang can gain high amount of proceeds of fraud crime with low physical harm and time expenditure. This type of fraud is often perpetrated through online channels, in the virtual world, so that the law enforcement agencies are difficult to investigate, trace, identify, and catch the fraud gang. One of the challenges in combating online investment scams is the ability to identify malicious websites. Domain names of fraudulent websites are often changed frequently and even sometimes shut-downed for preventing from law enforcement detection. In this paper, we propose a text analysis algorithm for identifying domain names of fraudulent investment websites. The proposed algorithm extracts a set of linguistic features from domain names, such as length, number of hyphens, number of digits, presence of keywords associated with investment fraud, and presence of typos or grammatical errors. The algorithm then uses these features to train a machine learning model to classify domain names as malicious or legitimate. The proposed text analysis algorithm provides a promising approach for law enforcement agencies to proactively combat online investment scams. The algorithm can be used to inspect domain names and flag suspected fraudulent domains, enabling the authorities to take timely action to block these websites and protect potential victims.

Keywords Fraud · Domain name · Open data · Neural network · Lexical entailment

Y.-P. Lai (✉)

Department of Information Management, Central Police University, Taoyuan City, R.O.C, Taiwan
e-mail: yopolai@mail.cpu.edu.tw

1 Introduction

In recent years, violent crimes gradually decrease in Taiwan, while the incidence of fraudulent activities has been on the rise. Among fraudulent crimes, investment fraud is the most serious, because of its propensity to cause significant financial losses in individual cases. Moreover, investment fraud cases often involve communication with fraudulent tools being primarily computer websites or mobile apps. These computer websites and mobile apps are associated with domain names to facilitate access by victims. These domain names differentiate from those associated with long-term, commercially operated computer websites or mobile apps. Commercial domain names are typically composed of meaningful combinations of words to enhance memorability for the public or customers, thereby increasing the likelihood of repeat visits. Easily memorable domain names, especially those representing businesses and having a history of prolonged use, can hold exceptional value. Conversely, fraudulent domain names are frequently changed to evade law enforcement and make victim unable to remember. They are often renamed using meaningless words or simply adding numbers, since fraud gang want to rapidly generate and apply for new domain names. Experienced investigators can even identify fraudulent domain names solely based on the naming characteristics. Therefore, the translation of the domain name sensitivity exhibited by these experienced investigators into artificial intelligence capable of quickly and automatically screening large volumes of domain names and services for suspicion holds significant research value.

In previous fraud research, many fraud models have been proposed to detect and classify fraudulent activities. Fraud incidents often begin with the leakage of personal information, such as through phishing with fake investment or auction websites. Nidhi Desai et al. proposed a new privacy model against background knowledge attacks in 2022 [1]. Background knowledge is an important data source to attackers and can pose a significant threat to privacy-preserving data. They also proposed a new privacy model to counter these knowledge attacks. Zahoor Ahmed Soomro et al. proposed a framework for ID fraud prevention policies [2]. Their researches focus on policy systems and the impact of fake identity fraud. They concluded that data leakage plays a significant role in facilitating identity fraud. They proposed strengthening identity fraud prevention policies, data access management strategies, and policy awareness and updates [2]. These policies, along with compliance and audits, can reduce incidents of identity fraud. Yuying Li et al., for cyber insurance applications, model health-data breach with the Markov-modulated non-homogeneous Poisson process [3]. Their proposed model captures the dynamic and time-varying nature of a data breach, where the intensity of breach events depends on the state of a Markov process. Rafat Salameh and Khalid Munther Lutfi surveyed Jordanian commercial banks using a questionnaire to find the limits of artificial intelligence in combating cybercrimes [4].

There has recently been a rapid development in the application of artificial intelligence, particularly in nature language processing. Depending on the granularity of processing, text analysis can be divided into article, sentence, and character-level

segmentation. The domain names can be segmented into domain-name and top-level-domain. Character-level text analysis can be extended into analyzing certain sequences of characters, called n-grams. However, these text analyses are hard to apply for processing the domain name. In fraud domain name examinations, the sub-words of domain names should be examined for their lexical meanings and linguistic logic. This makes it challenging to find closely related studies. One notable area of related research is lexical entailment recognition, which focuses on finding hypernyms based on word meanings to expand information and infer semantic relationships.

In this paper, we analyze domain names of fraudulent investment websites, which are announced and provided on Taiwan's Anti-Fraud 165 platform to identify the features. We extract distinctive features from these domain names and employ them to train predictive models. When evaluating a network domain name whether binding to a fraud website, we input it into the trained predictive model. Additionally, we employ an n-gram segmentation method to compare the constituent parts of the network domain name under scrutiny with those of known fraudulent network domain names. Besides, the top-level domains are also considered as an index to determine whether the domain name belongs to a fraud website or not. As previously mentioned, many fraudulent domain names are composed of seemingly random or meaningless text. Hence, we utilize the components of known fraudulent network domain names as the basis for the textual composition of fraudulent network domain names. The results of this comparison are utilized as attention weighting to enhance predictive accuracy.

In the next section, we will provide an explanation of the data background available for the research and discuss the current relevant research literature. In the third section, we will elucidate the methodological procedures proposed. Finally, we present our conclusions.

2 Backgrounds and Related Works

The Taiwan police have established the Anti-Fraud 165 platform, which focuses on blocking domain names associated with fraud reported by the victims. These fraudulent domain names are also made public on government information websites. All of the domain names that are publicly disclosed have already been blocked, rendering them inaccessible for obtaining information related to the respective websites, such as webpage content, connection statistics, website certificate details, and so on.

On September 5, 2023, the Anti-Fraud 165 platform announced the open data of fraudulent domain names, totaling 9,737 blocked network domains. The dataset includes fields for WEBSITE_NM, WEBURL, CNT, STA_SDATE, and STA_EDATE, which are for website names, URLs, counts, date of URL appearance, and date of URL disappearance, respectively, as shown in Table 1. The "count" field represents the number of victims who are defrauded by accessing the website with

the domain name. To evade law enforcement, the fraud gang frequently shuts down websites, resulting in appearance and disappearance dates.

Observing the network domains in Table 1, we can see that these malicious domain names associated with fraud are markedly distinct from typical, legitimate domain names. They are not composed with meaningful words. This paper aims to employ Natural Language Processing (NLP) techniques to predict and evaluate whether certain URLs are likely being utilized by suspected fraudsters. To achieve this, we must extract features from the domain names.

AI research has mainly focused on analyzing entire articles or sentences to understand their meaning. In this context, however, the focus is on URLs, which involves analyzing individual characters to segment them and compare them to known fraud domain names. The distributional hypothesis states that words with similar distributions have related meanings. This hypothesis is used to deter similarity in general-purpose word embedding models [5, 6]. These models provide context-independent vector representations of words that contain semantic information. Word2Vec a well-known model proposed by Mikolov et al. [7] learns word representations by training a classifier to distinguish between nearby and distant words. Word2Vec tries to capture co-occurrence one window at a time. Pennington et al. proposed GloVe, which tries to capture the counts of overall statistics. The GloVe model learns by constructing a co-occurrence matrix that counts how often a word appears in different contexts [8]. Bojanowski et al. introduced fastText to generate better word embedding for rare words. The fastText is a variant of the Word2Vec model that builds word vectors as the sum of their constituent character n-gram vectors [9]. In addition to context-independent models, several contextual word embedding models have proven effective in contextual NLP tasks, such as ELMo, BERT, and GPT [10–12]. BERT, a Bidirectional Encoder Representations from Transformers model proposed by Devlin et al., is designed to pre-train deep bidirectional representations from unlabeled texts by jointly conditioning on both left and right context in all layers [11]. With BERT, a word can have different vector representations depending on its appearance context.

A major drawback of the universal word embedding model is that it merges various relations into a single vector space, which can lead to ambiguity and inaccuracies. Several works have shown that the representation of words can be improved for lexical entailment recognition (LER) tasks by using external knowledge. This article investigates the recognition of fraudulent website domain names, which differ from regular websites in that they are designed to be forgettable by victims. Conventional websites aim to attract regular user visits, so their domain names are meaningful and easy to remember. Fraudulent website domain names, on the other hand, are often designed to be similar to or deceptive of the domain names of legitimate websites, such as by replacing letters with numbers, like “o” with “0” and “” with “1”. Some of the fraudulent domain names are listed in Table 2. The fields of the table represent the website name, URL, number of fraud cases, statistical start date, and statistical end date, with the titles WEBSITE_NM, WEBURL, CNT, STA_SDATE, STA_EDATE. In fact, the domain names are provided by the victims after they report the fraud crime to the police. The policemen immediately start a case to compile statistics on the number of victims and the amount of scammed money. On average,

Table 1 The domain names of fraud investment websites provided by Taiwan’s 165 anti-fraud platform (as of September 5, 2023)

1	WEBSITE_NM	WEBURL	CNT	STA_SDATE	STA_EDATE
2	網站名稱	網址	件數	統計起始日期	統計結束日期
3	傑富瑞	zzn1ojin.com	2	2023/1/18	2023/1/31
4	「冒名」禾健金屬	mer.zwnflom.com	1	2023/7/4	2023/7/10
5	「冒名」禾健金屬	mer.zwnflom.com	1	2023/6/19	2023/6/25
6	0x	www.0xplusdapp.com	3	2023/5/1	2023/5/7
7	0x	www.0xdappplus.com	2	2023/4/25	2023/4/30
8	100Ex	www.100exbank.com	2	2023/5/1	2023/5/7
9	11st	global-11st.com	6	2023/8/15	2023/8/21
10	11st	global-11st.com	2	2023/8/8	2023/8/14
9730	鑫鴻財富	www.yxbdghsn.com	2	2023/7/11	2023/7/17
9731	鑫鴻財富	www.yxbdghsn.com	1	2023/7/4	2023/7/10
9732	鑫鴻財富	www.yxbdghsn.com	18	2023/6/12	2023/6/18
9733	鑫鴻財富	www.yxbdghsn.com	18	2023/5/29	2023/6/4
9734	鑫鴻財富	www.yxbdghsn.com	25	2023/5/22	2023/5/28
9735	鑫鴻財富	www.yxbdghsn.com	6	2023/5/8	2023/5/14
9736	鑫鴻財富	www.yxbdghsn.com	2	2023/5/1	2023/5/7
9737	鑫鴻財富	www.dhfudhf.com	22	2023/5/15	2023/5/21
9738	鑫鴻財富	api.ancjknae.com	4	2023/6/19	2023/6/25
9739	鑫鴻財富	api.ancjknae.com	30	2023/6/5	2023/6/11

the websites are shut down within 7 days. In the next section, the concept of the proposed method will be introduced.

3 Proposed Method

Fraudulent investment scam perpetrators typically provide a uniform resource locator (URL) for victims to access. A URL is composed of a protocol, domain name, and path. The path includes the subfolder structure where the webpage is located. As shown in Table 1, many of the domain names of fraud investment websites are composed of meaningless character combinations. The following is the basic format. (Format) protocol://domain-name.top-level-domain/path.

(Example) <https://www.cpu.edu.tw/p/422-1000-1004.php?Lang=zh-tw>

Top-level domains (TLDs), which are directly managed by the Internet Corporation for Assigned Names and Numbers (ICANN) through the Internet Assigned Numbers Authority (IANA), are primarily categorized based on the purpose of the websites. There are six main types:

1. <.com>: Originally designed for commercial organizations, it now has no restrictions and is the most widely used top-level domain.
2. <.edu>: Reserved for educational and academic organizations in the United States. Other countries typically use it as a second-level domain.

Table 2 The characteristics of fraud websites

WEBSITE_NM	WEBURL	CNT	STA_SDATE	STA_EDATE
網站名稱	網址	件數	統計起始日期	統計結束日期
im.B借貸媒合平台	www.imb.com.tw	135	2023/5/1	2023/5/7
Bit-C	www.bit-cone.com	88	2022/2/21	2022/2/27
BCH	app.jinniuio52.com/apps/index/97ec10a349318152	83	2022/1/24	2022/2/6
幣牛 (CoinBull)	phone.bittrillion.com	66	2022/5/9	2022/5/15
德勝	www.obnerw.com	61	2022/8/15	2022/8/21
幣牛 (CoinBull)	phone.bittrillion.com	58	2022/5/2	2022/5/8
JOOM	web.joom91.com	56	2022/4/18	2022/4/24
陽信	b005.yuangxin.me	55	2022/2/21	2022/2/27
Sftimo	www.sftimoeth.com	53	2023/5/22	2023/5/28
Hopoo	mq7fh.hnwstsc.com/zfkec	50	2022/7/25	2022/7/31
im.B借貸媒合平台	www.imb.com.tw	49	2023/5/8	2023/5/14
Platforms	aner.platforms.agency	47	2022/2/28	2022/3/6
Platforms	aner.platforms.asia	41	2022/2/21	2022/2/27
臺寶	taibao.tw	41	2022/4/18	2022/4/24
Quant-Fin	quantfin.com.tw	40	2022/1/24	2022/2/6
Currency	rays.currency.golf	37	2022/1/24	2022/2/6
TF Global	www.tfglobaltrade.com	36	2022/1/3	2022/1/9
穆迪 (Moody's)	www.modsinvestorscby.com	36	2022/3/7	2022/3/13
im.B借貸媒合平台	www.imb.com.tw	35	2023/4/25	2023/4/30
Quant-Fin	www.quant-fin.com	34	2022/1/17	2022/1/23
華鼎	www.cjmqvyq.com	34	2022/6/20	2022/6/26
傑富瑞	zznuojin.com	33	2023/1/11	2023/1/17

- 3. <.gov>: Represents government-related entities in the United States. Other countries often use it as a second-level domain.
- 4. <.mil>: Reserved for military-related organizations in the United States Department of Defense.
- 5. <.net>: Originally intended for network service providers, but now unrestricted and serves as an open top-level domain.
- 6. <.org>: Initially designated for non-profit entities and organizations, but now unrestricted, and still primarily used by non-profit organizations.

The top-level-domains of these fraud investment websites are shown in Table 3. Some of the top-level domains of fraud investment websites are not in the regular organization classification. Observing the domain names and top-level domains together, we found that the domain names used by fraudulent websites are different from the naming conventions of normal websites. Within such fraudulent website URLs, the analysis is primarily focused on the preceding portion of the URL to extract features, without considering the “path” part. Regarding URL features, it is important to note that fraudulent website URLs reported by victims through law enforcement notifications typically contain only the basic format mentioned above. They lack certain data that can be used to obtain additional information to facilitate further analysis, such as certificate application dates, certificate applicants, website establishment times, website content, web ranking, and web connections per day.

Additionally, within the domestic context, once victims report fraudulent activities, websites are typically blocked, making it impossible to gain more information about the fraudulent websites. This information is typically evaluated and provided by statistics websites such as <https://www.similarweb.com/zh/> and <https://www.semrush.com/website/>.

In this context, we conduct an analysis of the domain name section in the aforementioned URL format using the following criteria:

1. Typo Weighting: Common fraudulent websites replace the letters “o” and “l” with the numbers “0” and “1” to deceive users.

Table 3 Top-level domain in domain names of fraud investment websites provided by Taiwan’s 165 anti-fraud platform (as of September 5, 2023)

Top-level-domain in domain names of fraud investment websites				
agency	center	gallery	mobileprovi- sion	today
ageny	cf	gold	mom	tokyo
ai	chcprotw	golf	net	top
api	city	group	network	town
app	click	gtxd	oexib5	training
appshops	cloud	host	one	tv
asia	club	icu	online	tw3368
aspx	co	in	org	tw692
ayju	com	info	pro	tw718
beauty	cyou	ink	qalj	university
best	design	institute	red	vip
bi	didjx	io	rocks	website
bio	digital	kbai	sale	wiki
bixc	directory	kudj	sbs	win
biz	dn	life	shop	work
bond	dns	link	site	works
bq9r	drpj	live	social	world
buzz	expert	ltd	solutions	xraf
c588	finance	lvcqot	space	xyz
cam	fit	me	store	
casa	fun	mhrc	tennis	
cc	fwtjfu	mobi	tips	

2. **Numeric Weighting:** Fraudulent webpages are often re-registered with numbers added before or after the domain name.
3. **Similarity to Other Known Fraudulent URLs:** Similar to the previous point, fraudulent URLs often add symbols such as “—” to reapply for domain names.
4. **Meaningless Weighting:** Fraudulent websites, often provided by scam operators, do not use memorable text combinations related to webpage content in their domain names. In some cases, fraudulent domain names are simply combinations of letters in close proximity on the keyboard.
5. **Top-Level Domain Units:** Common top-level domains include “.com,” while fraudulent websites use the top-level domains listed in Table 3.

In our approach, we consider sub-words as part of the features and dissect them. We retain the extracted sub-words for similarity detection with the domain names of known fraudulent websites. Additionally, if a sub-word has hypernyms related to “investment” or “gambling,” we assign it additional weighting. This process becomes a reference indicator in predicting whether a domain name belongs to a fraudulent website. We employ a dot product formula as the Attention Weighting Scheme in the prediction function, as follows.

$$AWS(v_{candidate}, v_{fraud}) = \max_{v_i \in v_{fraud}} \frac{v_{candidate} \cdot v_i}{\|v_{candidate}\| \|v_i\|}$$

The proposed schema is shown in Fig. 1. It has two phases: training and evaluation. In the training phase, we process the fraud domain names, obtained from the Taiwan’s Anti-Fraud 165 platform, to extract features. We store the sub-words of the fraud domain names (by-products) in a database for the attention weighting scheme in the evaluation phase. We extract the features of domain names of fraudulent websites and popular investment websites according to the domain-name section analysis criteria and feed them into the training neural network. The features are in vector format. We obtain the list of popular investment websites from the website <https://www.similarweb.com/zh/>.

In the evaluation phase, we process the examined domain name in the feature extraction stage to obtain its vector value. Then, the AWS similarity step calculates the similarity of the sub-words from the domain name to those from the known fraudulent websites. We feed the features of the domain name into the trained neural network. By considering the outputs of these two stages, we provide the quick screening prediction result for the domain name.

The domain-name part in the URL format is analyzed under the following conditions:

1. **Typo weighting:** Common scam websites use “0” to replace “o,” “1” to replace “l,” etc., to deceive.
2. **Numerical weighting:** Since fraudulent webpages may be blocked by the police, they are often prefixed or followed by numbers during re-application.

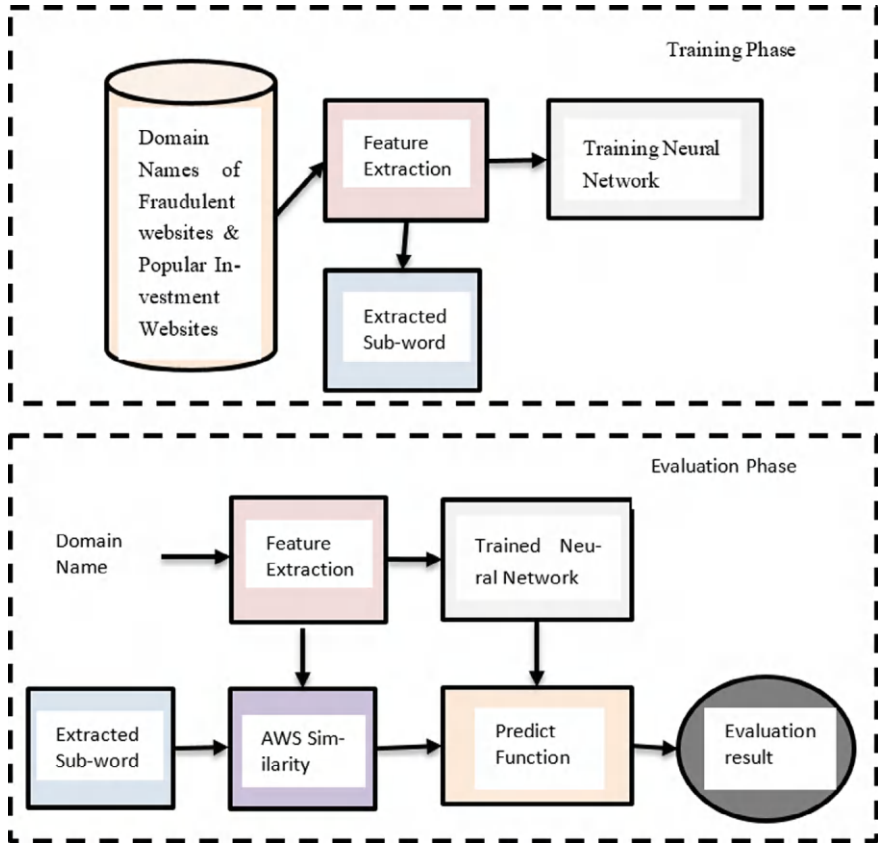


Fig. 1 Proposed model

3. Similarity to other known fraudulent URLs: This is similar to the previous condition; it is common to add symbols such as “—” during re-application for a domain name.
4. Meaningless weighting: Fraudulent webpages are often created by operators who then send the URL to the victim. Unlike ordinary webpages, they often combine domain names with words related to the content of the webpage to create memorable URLs. In some cases, fraudulent website domain names are simply a combination of letters in adjacent positions on the keyboard.
5. The top-level domain: the common one is .com. In addition to .com, the top-level domain used by fraudulent web pages also uses uncommon top-level-domain.

4 Conclusion

Fraudulent crimes have become increasingly rampant in recent years. Most researchers provide ways to smartly classify the types of fraud crimes to find prevention methods. This paper proposes an AI-based method for quick screening of domain names to identify potentially fraudulent websites. In other words, the proposed text analysis algorithm provides a promising approach for law enforcement agencies to proactively combat online investment scams. First, we perform sub-word segmentation on fraudulent domain names obtained from Taiwan's Anti-Fraud 165 platform. We then extract vector features of domain names of these known fraudulent websites and popular investment websites to train predictive models. The trained models are then used to quickly screen internet domain names. Further scrutiny is then applied to the screened domain names that are suspected to be fraudulent websites.

Due to the variety of scam tactics and fake investment targets, the open data from Taiwan's Anti-Fraud 165 platform provides the feature of "website name". Typically, several fraudulent websites with the same website name are manipulated by the same fraud gang, which means that these fraudulent websites can be classed into one group. In this paper, we employ a semantic approach, utilizing Natural Language Processing (NLP) techniques, to parse and assess internet domain names. We assess the similarity of sub-words in examined domain names and fraudulent domain names by applying Attention Weighting Schemes to determine whether they are in the same class. In the future, we plan to gather additional malicious internet domain names and establish hierarchies for various fraud case categories to enable rapid classification of fraud types. We will also develop analytical models tailored to different fraud patterns.

References

1. Desai N, Das ML, Chaudhari P, Kumar N (2022) Background knowledge attacks in privacy-preserving data publishing. *Comput Secur* 122:1–12
2. Soomro ZA, Shah MH, Thatcher J (2021) A framework for ID fraud prevention policies in E-tailing sector. *Comput Secur* 109:1–16
3. Li Y, Mamom R (2023) Modelling health-data breaches with application to cyber insurance. *Comput Secur* 124:1–23
4. Salameh R, Lutfi KM (2021) The role of artificial intelligence on limiting Jordanian commercial banks cybercrime. *Accounting* 7:1147–1156
5. Harris Z (1954) Distributional structure. *Word* 10(2–3):146–162
6. Firth JR (1957) A synopsis of linguistic theory 1930–1955. *Stud Linguist Anal* 1–32
7. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. In: *Proceedings of the 1st international conference on learning representations*
8. Pennington J, Socher R, Manning CD (2014) Glove: global vectors for word representation. In: *Proceedings of the 2014 conference on empirical methods in natural language processing*, vol 14. pp 1532–1543.
9. Bojanowski P, Grave E, Joulin A, Mikolov T (2017) "Enriching word vectors with subword information", *Transactions of the Association for. Comput Linguist* 5:135–146

10. Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L (2018) Deep contextualized word representations. In: Proceedings of the 2018 conference of the north American chapter of the association for computational linguistics: human language technologies, association for computational linguistics. pp 2227–2237
11. Devlin J, Chang M-W, Lee K, Toutanova K (2019) BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, association for computational linguistics, vol 1, (long and short papers). Association for Computational Linguistics, pp 4171–4186
12. Radford A, Jeffrey W, Child R, Luan D, Amodei D, Sutskever I (2019) Language models are unsupervised multitask learners. OpenAI Blog 1(8):9

The Evaluation Approach for Indoor Design Usability Based on Gaussian Mixed Model



Hsing-Chung Chen, Yong-Jian Siao, Pei-Yu Hsu, Jhih-Sheng Su,
and Yuan-Jung Lee

Abstract In the realm of interior design, professionals categorize space layouts based on the attributes and characteristics of furniture. Various furnishings exist, and tools like Rhino6 can autonomously produce numerous layout designs. However, while such automated layout software offers convenience, the generated designs often fail to adhere to established interior design principles, such as “doors should not directly face beds” or “bed headboards should not be positioned against windows.” This study delineates two distinct features for algorithmic analysis. The first pertains to the angle and distance between the bed’s edge midpoint and the room’s window, while the second relates to the angle and distance between the bed’s edge midpoint and the door. Subsequent analysis of these features via diverse clustering algorithms revealed that the Gaussian mixture model outperforms its counterparts in clustering efficiency. Consequently, this model emerges as a viable approach for scrutinizing interior design rules.

H.-C. Chen (✉) · Y.-J. Siao · P.-Y. Hsu · J.-S. Su
Department of Computer Science and Information Engineering, Asia University,
Taichung 413305, Taiwan
e-mail: cdma2000@asia.edu.tw; shin8409@ms6.hinet.net

Y.-J. Siao
e-mail: 107121005@live.asia.edu.tw

P.-Y. Hsu
e-mail: 112221001@live.asia.edu.tw

J.-S. Su
e-mail: 108121019@live.asia.edu.tw

H.-C. Chen
Department of Medical Research, China Medical University Hospital, China Medical University,
Taichung 404327, Taiwan

Y.-J. Lee (✉)
Feng Chia University, Taichung 40724, Taiwan
e-mail: yjlee@o365.fcu.edu.tw

1 Introduction

In the process of interior design, designers conceptualize furniture arrangements to optimize interior spaces. Notably, a single room can accommodate diverse furniture layouts. Numerous layout schemes are generated, from which interior designers meticulously select the most viable ones. These chosen layouts subsequently serve as blueprints for interior decorators, construction firms, and academic institutions specializing in interior design. It's observed that successful layouts often share certain commonalities, such as the principles that "doors should not directly face beds" and "bed headboards should not be positioned beneath windows." Various clustering algorithms can be employed to discern the optimal algorithms suitable for interior design applications.

2 Background Knowledges and Related Researches

1. Rhinoceros 3D

Rhinoceros 3D, commonly referred to as Rhino, stands as a premier 3D modeling software tailored for professionals. The most recent iteration of this software is Rhino 6 [1]. Its versatility has made it an indispensable tool across various sectors, including industrial design, yacht fabrication, jewelry crafting, transportation, toy manufacturing, and construction-related domains [1]. This study leverages the capabilities of Rhino 6, in conjunction with its integrated plug-in, Grasshopper, to conceptualize and formulate an interior design blueprint for a room.

2. Google Colaboratory

Google's innovative platform, Colaboratory (often abbreviated as Colab) [2], generously offers complimentary access to NVIDIA K80 GPU resources and virtual machines equipped with dual 2.2 GHz Xeon CPUs. This platform seamlessly integrates a Linux (Ubuntu) environment, Python, and the Jupyter notebook, encompassing pivotal packages like TensorFlow. Moreover, Colab facilitates the installation of essential execution packages, including but not limited to Keras, OpenCV, PyTorch, MxNet, and XGBoost. For individuals with an active Google Drive account, the platform is readily accessible at no cost.

3. Scikit-learning

Scikit-learn [3] stands as an open-source machine learning library tailored for the Python programming language [4]. It encompasses a comprehensive suite of algorithms for classification, regression, and clustering. Noteworthy algorithms include support vector machines, random forests, gradient boosting, K-means, and DBSCAN, among others. This study employs the functionalities offered by Scikit-learn to execute clustering algorithm computations.

4. Gaussian Mixture Model

The Gaussian Mixture Model (GMM) [5] serves as an advanced extension of the singular Gaussian probability density function. It possesses the capability to adeptly approximate the density distribution of entities regardless of their shape [5]. Distinctively, there are two primary variants of Gaussian Mixture Models: The Single Gaussian Model (SGM) and the Gaussian Mixture Model (GMM). Their differentiation lies in the parameters of the Gaussian probability density function (PDF). Each individual Gaussian model can be perceived as a distinct category. When a sample, denoted as x , is introduced, its value is ascertained through the PDF. Subsequently, a threshold assessment determines the affiliation of the sample to a specific Gaussian model [5]. While the SGM is adept for bifurcated classifications, the GMM, with its multiple models, offers a more nuanced categorization, making it apt for multi-class divisions. The versatility of GMM renders it suitable for intricate object modeling [5, 6].

5. Hierarchical Clustering

Hierarchical grouping encompasses two primary methodologies: agglomerative (clustering) and divisive. The agglomerative approach, often referred to as the bottom-up method, commences with individual data points and progressively amalgamates them into larger clusters. Conversely, the divisive approach, characterized as the top-down method, begins by treating all data points as a singular expansive cluster. The subsequent process entails systematically partitioning this extensive cluster into smaller, more defined clusters [7, 6].

6. K-Means

K-Means is a clustering algorithm introduced by J. B. MacQueen in 1967. A prerequisite for its application is the pre-specification of the number of clusters, denoted as k . The algorithm seeks to optimize clustering by maximizing the value derived from a specific formula shown in Eq. (1), thereby ensuring optimal groupings [8, 6].

$$\underset{i=1}{\operatorname{argmin}} \sum_{i=1}^k \sum_{\mathbf{x}j \in Si} \|\mathbf{x}j - \boldsymbol{\mu}i\|^2 \quad (1)$$

7. DBSCAN

Density-based spatial clustering of applications with noise (DBSCAN) is a prominent clustering algorithm introduced by Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu in 1996 [9]. The foundational principle of this algorithm is density. Given a set of points within a specific space, DBSCAN has the capability to aggregate proximate points into a cluster, particularly those with a high density of neighboring points. Concurrently, it identifies and labels outlier points situated in low-density regions, where even the nearest point is considerably distant.

8. BIRCH

Balanced iterative reducing and clustering using hierarchies (BIRCH) is a clustering methodology introduced by Tian Zhang in 1996 [10]. A distinctive feature of BIRCH is its ability to execute high-caliber clustering on extensive datasets while operating within constrained memory resources [6].

9. Cluster evaluation indicators

(a) Calinski-Harabasz Index

The index of group evaluation, commonly referred to as the standard of difference ratio, serves as a metric to gauge the efficacy of clustering. A hallmark of this index is its straightforward computation. A higher score, denoted as ‘s’, is indicative of superior clustering outcomes. The mathematical expression shown in Eq. (2) to determine this score is delineated below [6].

$$s(k) = \frac{tr(B_k)}{tr(W_k)} \frac{m - k}{k - 1} \quad (2)$$

In the given context m represents the total number of samples present in the training set, while k denotes the number of distinct categories or clusters. The matrix B_k signifies the covariance matrix between clusters, capturing the degree of spread between different clusters. Conversely, W_k represents the covariance matrix of the data within a specific cluster, indicating the spread of data points inside that cluster. The term ‘tr’ stands for the trace of a matrix, which is the sum of its diagonal elements. For optimal clustering, it is desirable to have a minimal covariance within clusters and a maximal covariance between clusters. Under such conditions, the Calinski-Harabasz scores, which gauge the quality of clustering, tend to be elevated [11, 6].

(b) Davies-Bouldin Index

The Davies-Bouldin index (DBI), introduced by David L. Davies and Donald W. Bouldin in 1979 [12], serves as a metric for cluster validation. The essence of this index lies in its ability to quantify the quality of clustering. Specifically, a lower DBI value is indicative of superior cluster separation, meaning clusters are well-distinct from each other, and enhanced “tightness” or cohesion within the clusters, implying data points within a cluster are closely packed.

(c) Silhouette Coefficient

The Silhouette Coefficient is a measure employed in cluster analysis to gauge the quality of clusters formed [13]. It is computed using the average intra-cluster distance (denoted as a) and the average distance to the nearest cluster that the sample is not a part of (denoted as b). The Silhouette Coefficient for a particular sample is derived by Eq. (3) from the difference between b and a divided by the greater of the two. The resulting value provides insights into the sample’s appropriateness to its assigned cluster in comparison to other clusters [6].

$$\frac{(b - a)}{\max(a, b)} \quad (3)$$

In the context of the Silhouette Coefficient, a value of 1 is considered the best, signifying that the sample is appropriately clustered. Conversely, a value of -1 is deemed the worst, indicating that the sample might have been better suited to a neighboring cluster. Values proximate to 0 suggest that clusters are overlapping, making it challenging to distinctly separate the data points [6].

10. Related Researches

The exploration on the applicability of Gaussian Mixture Model in interior design via poster presentation by Chen et al. [14], which is also according to the Siao's master thesis in Ref. [6]. This poster delves into the innovative intersection of the Gaussian Mixture Model's applicability within the realm of interior design. Presented at the NCS 2019 National Computer Conference, it likely offers preliminary insights and observations, signifying a pioneering approach to amalgamating traditional design principles with contemporary statistical modeling.

Next, there are some artificial algorithms [15–18] was proposed to deal with the similar problems. For example, the week-wise student performance early prediction in virtual learning environment using a deep Explainable Artificial Intelligence (XAI) proposed by Chen et al. [15]. This manuscript addresses the pivotal challenge of predicting student performance on a week-by-week basis in virtual learning environments, employing deep explainable AI. Given the current surge in remote and online education, such research holds paramount importance, offering educators a tool to identify and assist students who may be lagging, in a timely manner. In addition, the pulse-line intersection method with Unboxed Artificial Intelligence (UAI) for hesitant pulse wave classification is also proposed by Chen et al. [16]. Their study introduces a novel pulse-line intersection methodology, integrated with unencapsulated AI techniques, tailored for the classification of hesitant pulse waves. This could represent a significant stride in the medical and health domain, potentially revolutionizing cardiovascular health assessment and prognostication. Moreover, the novel predictor for exploring PM2.5 spatial-temporal propagation by using Convolutional Recursive Neural Networks (CRNN) is also presented in Ref. [17]. This research elucidates a state-of-the-art predictor for examining the spatial-temporal propagation of PM2.5, utilizing convolutional recursive neural networks. In light of global concerns surrounding air pollution and environmental health, such investigations are crucial for the predictive modeling and mitigation of air quality challenges. Furthermore, the estimation of various walking intensities based on wearable plantar pressure sensors using Artificial Neural Networks (ANN) is presented in Ref. [18]. This manuscript investigates the estimation methodologies of diverse walking intensities, leveraging wearable plantar pressure sensors in conjunction with artificial neural networks. With the advent of wearable technology, such research endeavors can significantly enhance our understanding and monitoring capabilities of individual locomotion and overall health metrics.

In summary, the aforementioned articles span a broad spectrum of domains, from interior design to health and environmental sciences, underscoring the transformative potential of artificial intelligence and machine learning in a myriad of practical applications. Each paper presents novel methodologies and perspectives, serving as invaluable references for researchers and experts within the respective fields.

3 Experiments and Results

This study employs room floor plans that are randomly generated using Rhino6 software [1]. As depicted in Fig. 1, we measure the distance and angle between the central point at the head of the bed and the midpoint of the left window. Subsequent computational tasks referred in [2] are executed by using Google Collaboratory, as illustrated in Fig. 2.

From the data representation in Fig. 3, it is evident that the data distribution is segmented into five distinct clusters. Algorithms such as the Gaussian mixture model, K-Means, and BIRCH necessitate the pre-specification of the number of cluster centers. Consequently, we utilize five cluster centers for our calculations. The final visualizations are presented in Figs. 4, 5, 6, 7 and 8, respectively. Comprehensive group evaluation metrics are calculated by Eqs. (1), (2) and (3), which are systematically tabulated in Table 1.

Adhering to the aforementioned methodology, we executed analogous procedures considering the relationships between the headboard and the right window, as well as between the headboard and the door. The resultant data and metrics derived from these analyses are systematically presented in Tables 2 and 3, respectively.

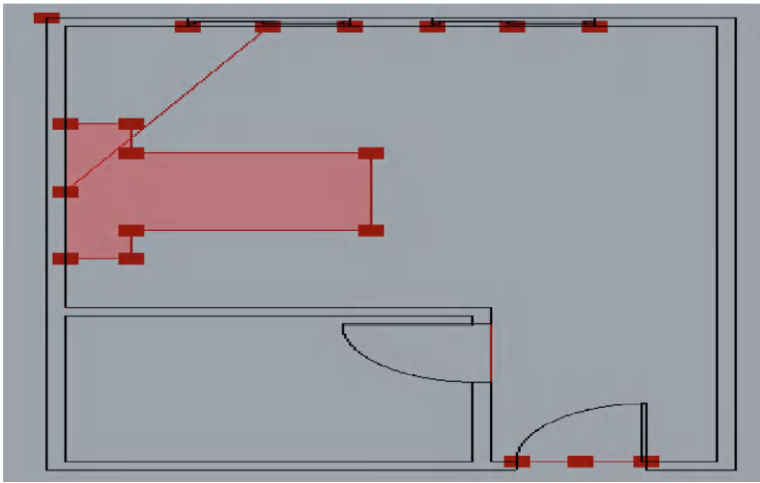


Fig. 1 Example of the distance between the bed and the left window

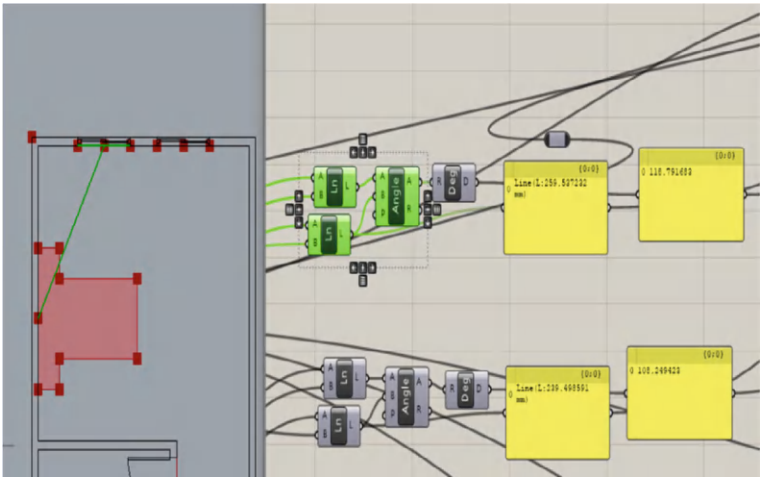


Fig. 2 Data capture of the distance and angle between the bed and the left window

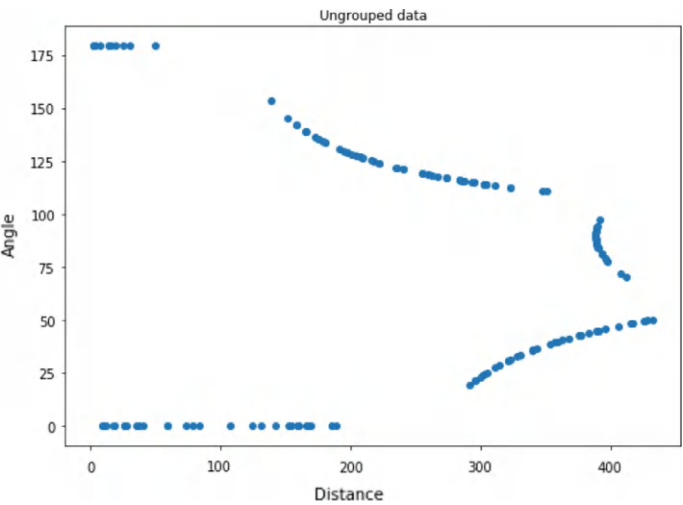


Fig. 3 Distribution of data without clustering

Upon examining the data presented in Tables 1 and 2, it is evident that the K-Means algorithm yields the most optimal score among the clustering methods evaluated. DBSCAN emerges as the second most effective, while the Gaussian mixture model’s score lags behind the other algorithms. Analyzing the data from Table 3, the hierarchical grouping method outperforms K-Means in both the Calinski Harabasz index and the Davies-Bouldin index. However, when considering the Silhouette Coefficient across all samples, K-Means exhibits superior performance.

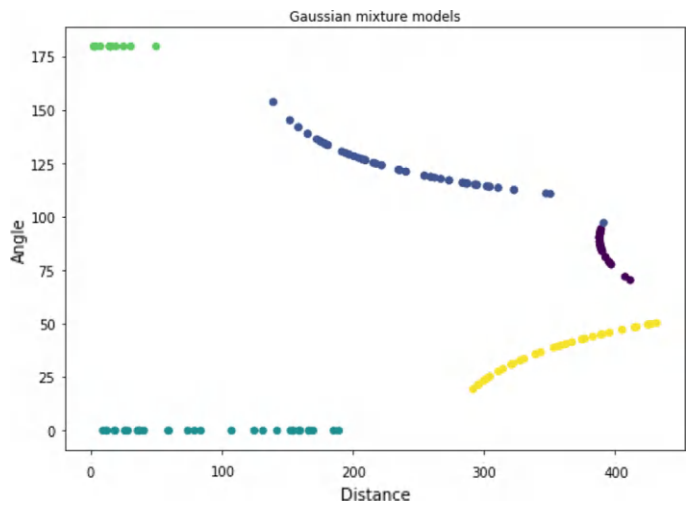


Fig. 4 Clustering effect of Gaussian mixture model

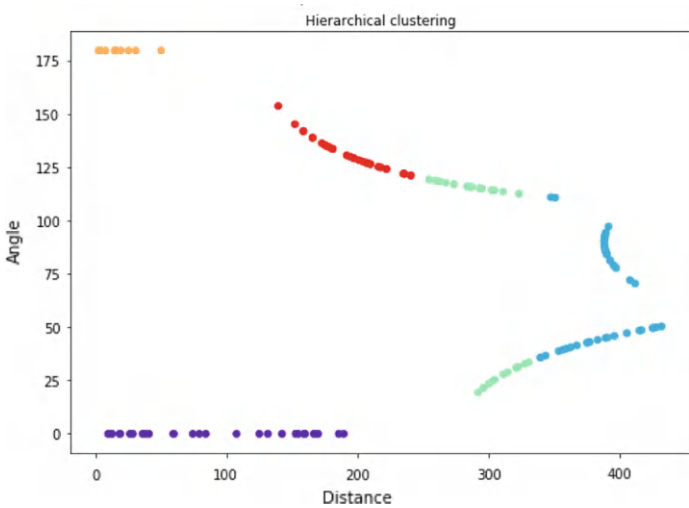


Fig. 5 The grouping effect of the hierarchical grouping method

Based on the aforementioned observations, K-Means demonstrates the most proficient clustering capabilities across the three evaluation indicators. A closer inspection of Fig. 3 reveals that the data distribution exhibits extremities. Comparing the clustering outcomes from Figs. 4, 5, 6, 7 and 8, the Gaussian mixture model distinctly segregates the data into five clusters, each demarcated based on data density. DBSCAN’s clustering results in four distinct clusters. The clustering outcomes of the hierarchical grouping method, K-Means, and BIRCH are relatively analogous.

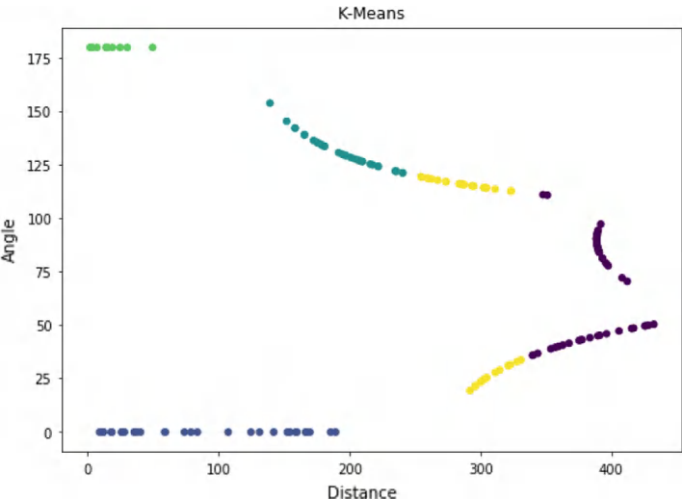


Fig. 6 K-Mean’s grouping effect

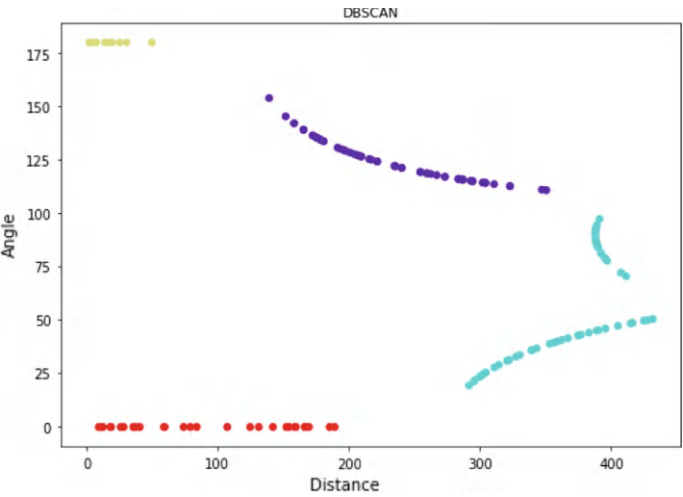


Fig. 7 The clustering effect of DBSCAN

However, juxtaposed against the Gaussian mixture model’s clustering, there remains potential for enhancement.

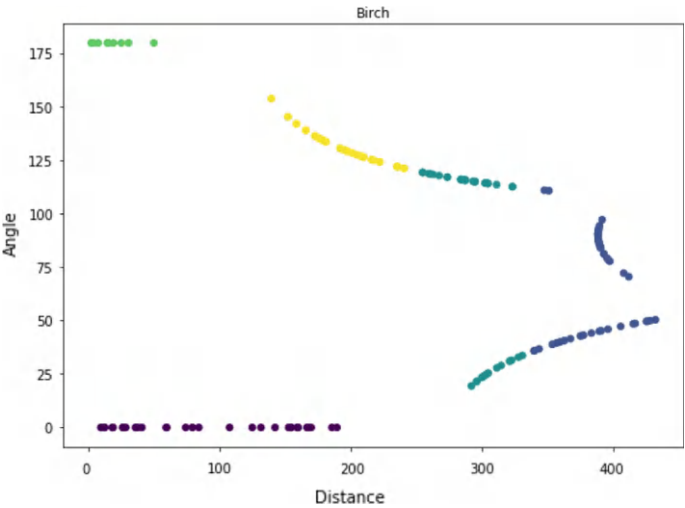


Fig. 8 The grouping effect of BIRCH

Table 1 The scores of the classification effect of the bedside and left window

Indexes	Calinski-Harabasz index	Davies-Bouldin index	Silhouette coefficient
Algorithms			
Gaussian mixture model	324.52146010052155	0.599210082239195	0.5316455493568039
Hierarchical clustering	459.3926485900972	0.6079168622377505	0.5836543659397396
K-means	*572.9345788845251	*0.4688595695175325	*0.6139561056693847
DBSCAN	390.24677342590365	0.5245342236599866	0.6035363885430575
BIRCH	472.2288658259639	0.5620348720983936	0.5621086687709492

The asterisk (*) highlights the highest score in each column, indicating the best-performing algorithm based on the given metric

Table 2 The scores of the classification effect of the bedside and the right window

Indexes	Calinski-Harabasz index	Davies-Bouldin index	Silhouette coefficient
Algorithms			
Gaussian mixture model	520.5270417923067	0.5147006568503714	0.5603277968680792
Hierarchical clustering	485.48591696538904	0.49860265608836407	0.6079913709361388
K-means	*572.9345788845251	*0.4688595695175325	*0.6139561056693847
DBSCAN	374.10262948197214	0.5463451631272976	0.6000450986501619
BIRCH	472.2288658259639	0.5620348720983936	0.5621086687709492

The asterisk (*) highlights the highest score in each column, indicating the best-performing algorithm based on the given metric

Table 3 The scores of the effect of classification of bedside and door

Indexes	Calinski-Harabasz index	Davies-Bouldin index	Silhouette coefficient
Algorithms			
Gaussian mixture model	819.7710775942018	0.7034562911578274	0.5466873825743074
Hierarchical clustering	*910.1899184937782	*0.5689491222265286	0.5646900105986986
K-means	864.9599760852363	0.5804845503658185	*0.5966425871093759
DBSCAN	304.71240646600916	0.6875858193552119	0.4756885523003814
BIRCH	670.0380246076829	0.5948667083757726	0.5666787058228743

The asterisk (*) highlights the highest score in each column, indicating the best-performing algorithm based on the given metric

4 Conclusions

This study systematically evaluated various clustering algorithms, including K-Means, DBSCAN, the Gaussian mixture model, hierarchical grouping methods, and BIRCH, using specific evaluation indicators such as the Calinski Harabasz index, Davies-Bouldin index, and the Silhouette Coefficient. Data derived from room floor plans generated by Rhino6 software served as the foundation for this analysis. Preliminary results, as presented in Tables 1 and 2, indicated that K-Means consistently outperformed other algorithms, with DBSCAN emerging as the second most effective. However, when considering the data in Table 3, the hierarchical grouping method demonstrated superior performance in certain metrics compared to K-Means, though K-Means retained an edge in the Silhouette Coefficient. A detailed examination of data distributions, as visualized in Figs. 3, 4, 5, 6, 7 and 8, further underscored the efficacy of K-Means, especially in scenarios with extreme data distributions. While the Gaussian mixture model exhibited clear demarcation in clustering, other methods like DBSCAN and BIRCH showed potential areas for improvement. Overall, K-Means emerged as the most proficient clustering algorithm across the evaluation indicators used in this study.

Acknowledgements This work was supported by the Chelpis Quantum Tech Co., Ltd., Taiwan, under the Grant number of Asia University: I112IB120. This work was supported by the National Science and Technology Council (NSTC), Taiwan, under NSTC Grant numbers: 111-2218-E-468-001-MBK, 110-2218-E-468-001-MBK, 110-2221-E-468-007, 111-2218-E-002-037 and 110-2218-E-002-044. This work was also supported in part by Asia University, Taiwan, and China Medical University Hospital, China Medical University, Taiwan, under grant numbers below. ASIA-111-CMUH-16, ASIA-110-CMUH-22.

References

1. Rhinoceros 3D. <https://www.rhino3d.com/tw/>
2. Google Colaboratory. <https://colab.research.google.com>
3. Pedregosa et al (2011) Scikit-learn: machine learning in python. *JMLR* 12 2825–2830
4. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Perrot M, Duchesnay É (2011) Scikit-learn: machine learning in Python. *J Mach Learn Res* 12:2825–2830
5. Introduction to the Gaussian Mixture Model (GMM). <https://blog.csdn.net/jojzhangju/article/details/19182013>
6. Siao YJ (2020) The study of the applicability for decision tree in interior design. Master's Thesis, Asia University
7. Rokach L, Oded M (2005) Clustering methods. In: *Data mining and knowledge discovery handbook*. Springer US, pp 321–352
8. MacQueen JB (2009) Some methods for classification and analysis of multivariate observations. In: *Proceedings of 5th Berkeley symposium on mathematical statistics and probability*. University of California Press, pp 281–297. 1967 [2009–04–07]. MR 0214227. Zbl 0214.46201
9. Ester M, Kriegel HP, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Proceedings of the second international conference on knowledge discovery and data mining (KDD-96)*. AAAI Press, pp 226–231. CiteSeerX 10.1.1.121.9220. ISBN 1–57735–004–9
10. Zhang T, Ramakrishnan R, Livny M (1996) BIRCH: an efficient data clustering method for very large databases, vol 25, no 2. In: *ACM SIGMOD record*. pp 103–114. <https://doi.org/10.1145/235968.233324>
11. Liu J Pinard. <https://www.cnblogs.com/pinard/>
12. Davies DL, Bouldin DW (1979) A cluster separation measure. *IEEE Trans Pattern Anal Mach Intelligence*. PAMI-1(2):224–227
13. Rousseeuw PJ (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Comput Appl Math* 20:53–65. [https://doi.org/10.1016/0377-0427\(87\)90125-7](https://doi.org/10.1016/0377-0427(87)90125-7)
14. Chen HC, Siao YJ, Lee YJ, Wang TW (2019) Exploration on the applicability of Gaussian mixture model in interior design. In: *NCS 2019 national computer conference*. Poster Session, National Quemoy University, pp 736–740
15. Chen HC, Prasetyo E, Tseng SS, Putra KT, Prayitno, Kusumawardani SS, Weng CE (2022) Week-wise student performance early prediction in virtual learning environment using a deep explainable artificial intelligence. *Appl Sci* 12(4):1–16. Article No. 1885. <https://doi.org/10.3390/app12041885>
16. Chen HC, Damarjati C, Putra KT, Chen HM, Hsieh CL, Lin HJ, Wu MY, Chen CS (2022) Pulse-line intersection method with unboxed artificial intelligence for hesitant pulse wave classification. *Inf Process & Manag* 59(2):1–12. Article Number 102855. <https://doi.org/10.1016/j.ipm.2021.102855>
17. Chen HC, Putra KT, Weng CE (2022) A novel predictor for exploring PM2.5 spatiotemporal propagation by using convolutional recursive neural networks. *J Internet Technol* 23(1):165–176. <https://doi.org/10.53106/160792642022012301017>
18. Chen HC, Sunardi, Liao BY, Lin CY, Akbari VBH, Lung CW, Jan YK (2021) Estimation of various walking intensities based on wearable plantar pressure sensors using artificial neural networks. *Sensors* 21(19):6513, 1–14. <https://doi.org/10.3390/s21196513>

A Lightweight Deep Leakage Prevention Model with Mixing Multi-gradients



Wei-Chun Hsiao and Chih-Hung Wang

Abstract In deep learning model, the training procedure can obtain appropriate weights by leveraging the information of gradient descent. However, the gradients between the weights may leak the original neural network training data if they are not well-designed. Zhu et al. (Advances in neural information processing systems, Curran Associates, Inc., [5]) as proposed a kind of attacking scenario named deep leakage from gradients. There were many mitigation strategies proposed in the literature, such as mixing Gaussian noises and model quantization. However, they usually somewhat lost classification accuracy during prevention process. This paper proposes a lightweight attachment model which can be used as an extension on the original classification model to prevent deep leakage from gradients with low information loss. This approach can complete the goal of mitigation on trying to minimize the changes of the original structure of the model and can be easily implemented on some existent deep learning classifiers.

Keywords Deep learning · Deep leakage from gradients · Artificial intelligence · Collaborative learning

1 Introduction

In recent years, artificial intelligence (AI) has become a critical technique for many research area, e.g., image classification, intrusion detection, recommender systems, etc. However, with the age of knowledge explosion, training process, such as deep learning, may be difficult to be conducted at only one single organization, since the data source are needed to be collected from different places. That means the distribution training is more popular in recent years. In addition, Konečný et al. [1] proposed a Federated Learning model (or called collaborative learning) that can be used to train

W.-C. Hsiao · C.-H. Wang (✉)

Department of Computer Science and Information Engineering, National Chiayi University, Chiayi, Taiwan

e-mail: wangch@mail.ncyu.edu.tw

a centralized model with high performance with the property of remaining training data on local machines. For the privacy protection, the individual organizations can hide their own training data and only exchange the updated information such as gradients of the model to the centralized server or other parties.

Some previous papers have pointed out the sharing of gradients is dangerous since it reveals a part of properties of the training data [2–4]. According to the paper proposed by Zhu et al. [5], the leakage of gradients may cause the recovery of original training data to be successful. That means the sharing of gradients without protection cannot be safe during collaborative learning, and even the individual training datasets can be completely stolen. Therefore, some possible solutions can be found such as adding noise (e.g., Gaussian or Laplacian noises) on gradients before releasing [5], or compressing/pruning gradients to reduce both communication cost and leakage risk [6]. Leveraging noise or quantization method will decrease the float precision and may not propagate precise information to the neural network. According to the study of Tishby and Zaslavsky [7], using bottleneck method to prevent the risk of gradients leakage can appropriately keep accurate classification and feature extraction, but it is somewhat difficult to predict the width of bottleneck.

In this paper, we put an attachment at the end of the original vulnerable/leakable deep learning model to migrate the possible flaws. The expended attachment has two fully connected layers with enough widths. To avoid information loss or compression, it is required that the widths of these two layers are multiple times of the last fully connected layer of the original model, which depends on the average size of pooling range. Afterwards, add the outputs of these two layers, carry out average pooling and perform normalization procedure. The extra expended layers in fact do not impact on the accuracy of the classification model and have the capability of making difficult for an adversary to reverse the training data from exchanging gradients and thus the threats of deep leakage can be avoided.

Meanwhile, our design was also inspired from Low-Rank Adaptation (LoRA) [8], widely applied on model fine-tuning recently. Under the minimized changes of the original model, we only add some specific layers at the end to connect the original model and therefore, our model can be regarded as a lightweight add-on.

2 Related Work

Consider the possibility of leakage for the deep learning model by the gradients, specifically in the image training sets, we review some important developments of Convolutional Neural Networks (CNN) technology for image classification to explain the background of attack strategies.

2.1 Convolutional Neural Networks

Lecun et al. in 1998 created LeNet [9], one of earliest CNN models. It got a high performance on the Modified National Institute of Standards and Technology database (MNIST) for the classification of handwritten digits. Afterwards, Krizhevsky et al. developed an outstanding classification model for ImageNet in 2012 [10]. They leveraged Rectified Linear Unit (ReLU) with non-saturating nonlinearity as eight layers activation function of CNN to speed up the training process. They also provided an efficient GPU implementation and tried to avoid overfitting by using dropout approach.

With the rapid development of GPU product, some deeper CNN models have been proposed in recent years. Simonyan and Zisserman designed a VGG (Visual Geometry Group) model [11]. There are several variants with different depths, such as VGG11, VGG13, VGG16 and VGG19. The proposed method can achieve outstanding performance for image classification.

However, the vanishing gradient problem may occur caused by increasing the layers of neural network in deep learning model. He et al. proposed a new structure ResNet, a deeper CNN, to apply residual structure to mitigate the problem of vanishing gradient [12].

2.2 Deep Leakage Problem

Zhu et al. in 2019 proposed a method to steal the complete training data from the leaked gradients [5]. That means if the adversary gets other participant k 's gradients $\nabla W_{t,k}$ at step t , and the differentiable model $F(x, W)$ and the weight W_t are shared among all participants, the adversary can try to recover the training data by the following formula. Assume that at first, the adversary initially selects a random dummy input x' and a label input y' . Then the dummy gradients can be obtained by feeding these dummy inputs into models.

$$x'^*, y'^* = \arg \min_{x', y'} \|\nabla W' - \nabla W\|^2, \text{ where}$$

$$\nabla W' = \frac{\partial \ell(F(x', W), y')}{\partial W}.$$

Note that the subscripts $\{t, k\}$'s are omitted in the above formula. That means if we optimize the dummy gradients close to the original ones in training set, it may also lead the dummy data to be gradually changed to close to the actual training data.

Figure 1 shows the abstract concept of Zhu et al.'s attacking scenario. The malicious adversary begins with a dummy image and applies participant's gradients ∇W to find the minimum of gradient distance for updating.

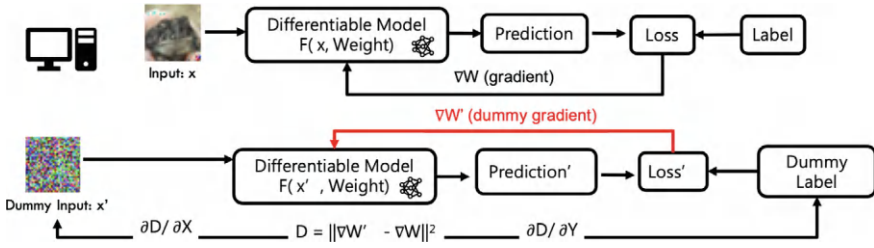


Fig. 1 Concept of Zhu et al.'s attacking scenario

Scheliga et al. [13] proposed a PRECODE to prevent the gradients leakage, which also can be regarded as adding a generic extension on arbitrary model architectures. They appropriately leveraged a “bottleneck” to decrease the leakage level of training data and used “feature extraction” technique to effectively avoid destroying the model performance. Moreover, Geiping et al. [3] demonstrated the attack on federated (or called collaborative) learning even if the system has aggregated the gradients of several iterations or images for the resistance of possible privacy leakage.

3 The Proposed Lightweight Solution

The proposed solution is to attach a lightweight extension connecting to the end of original model. Assume that the width of original model is m . The output of original model is separately delivered to two linear layers with width of $m \times n$. For security, the value of n is expected to be large. However, increasing the value of n would make the model become more uncontrollable, and adding extra parameters will reduce the training efficiency.

Afterwards, the system executes a tensor addition calculation on two outputs of the liner layers and proceeds to conduct an average pooling operation for the result. These two operations can effectively make the data recovery process more difficult. Since the tensor addition calculation extends the range of value, a normalization process is needed to narrow down the output range and keep the stability of training result. Finally, the system outputs model results by going through a linear layer with width m .

The concept of our lightweight add-on is shown in Fig. 2. The detailed design will be shown in Fig. 3.

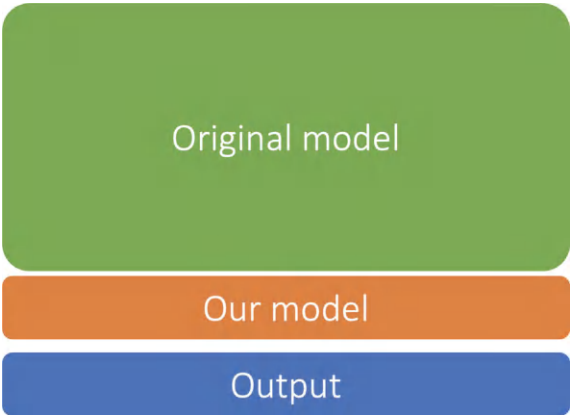


Fig. 2 Concept of lightweight add-on model

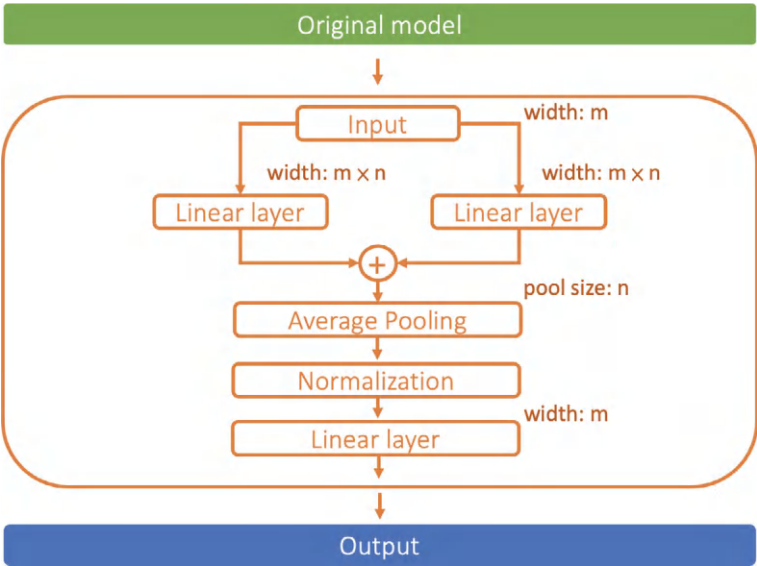


Fig. 3 Detailed design of proposed model

4 Experiments

In our experiments, we connect our extension attachment to different popular deep learning model, such as LeNet-5, VGG11, ResNet18 and two CNN models setup by ourselves. The experiments were conducted on the CIFAR-10 dataset [14]. The major observation is to verify that our extension would only cause slight impact on training accuracy.

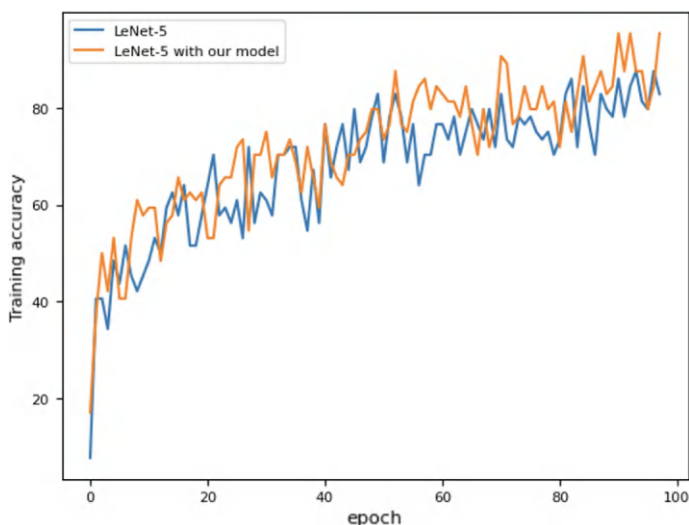


Fig. 4 Training accuracy of LeNet-5 model connecting to our extension

4.1 Model Training

The experiments show that as the different models are applied to connect our extension add-on, the training accuracies are still rapidly rising with the number of epochs increasing. That means the training accuracy would not be affected by adding extra extension model. Figures 4, 5, 6, 7 and 8 show the training accuracies results of LeNet-5, VGG11, ResNet18, four-layer CNN, and one-layer CNN plus three-layer linear model, respectively, connecting to our extension.

4.2 Test Accuracy

We also evaluate the test accuracy of different models connecting to our extension. Figure 9 shows the comparison results. It can be seen that the accuracies of our designs can be maintained close to the original models or even slightly higher than them, which means the extra parameters we add in the model can be effectively operated.

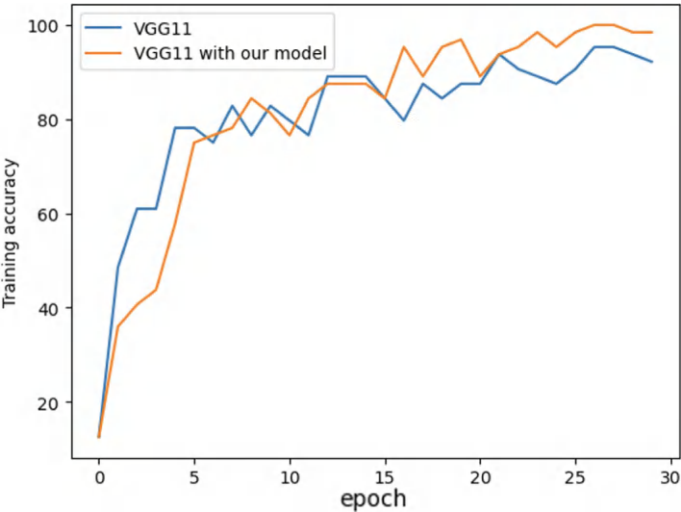


Fig. 5 Training accuracy of VGG11 model connecting to our extension

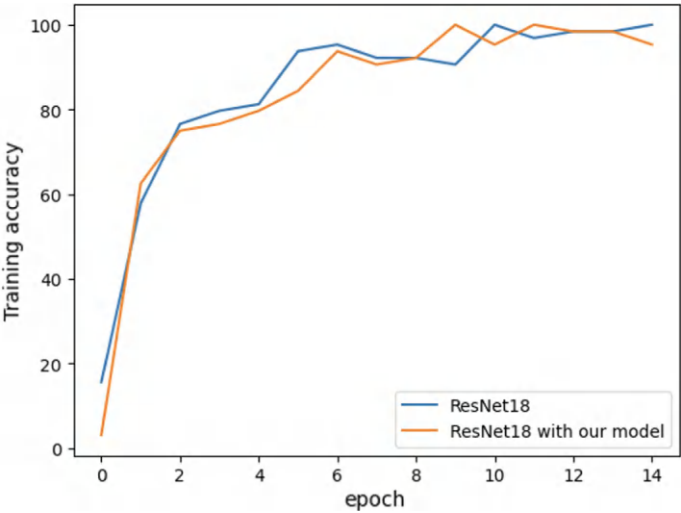


Fig. 6 Training accuracy of ResNet18 model connecting to our extension

4.3 Deep Leakage Testing

We refer to [5] to complete the programs for launching deep leakage from gradients. The results can be referred to Figs. 10, 11, 12, 13, 14 and 15 for four-layer CNN, VGG11 and LeNet-5. From these figures, we can see that as the models without connecting to our attachment (e.g., Figs. 10, 12 and 14), the rough shapes of the images

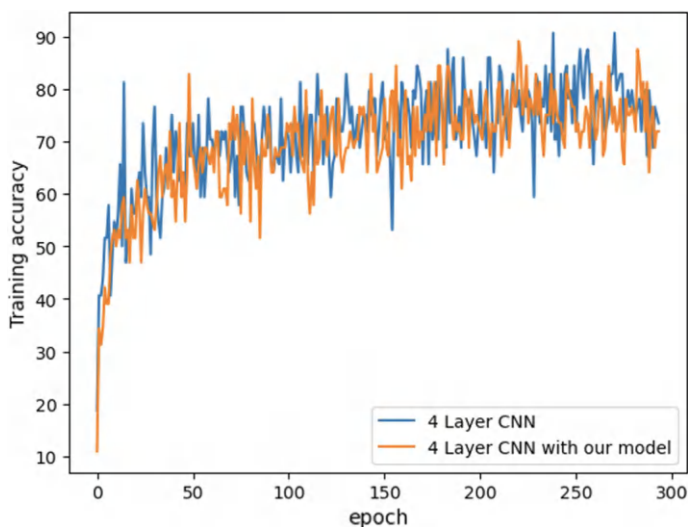


Fig. 7 Training accuracy of four-layer CNN model connecting to our extension

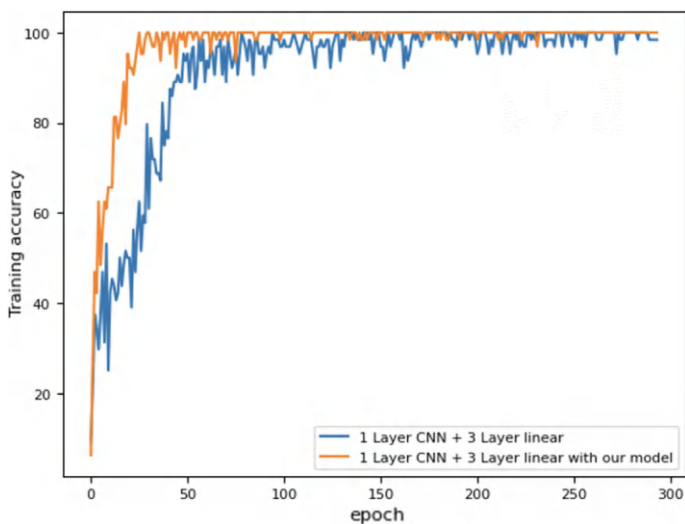


Fig. 8 Training accuracy of one-layer CNN plus three-layer linear model connecting to our extension

can be gradually recovered and become more and more evident during several iterations. On the other hand, as the model connecting to our attachment (e.g., Figs. 11, 13 and 15), the output results are still secure and just produce random noises under the attack with over 290 iterations.

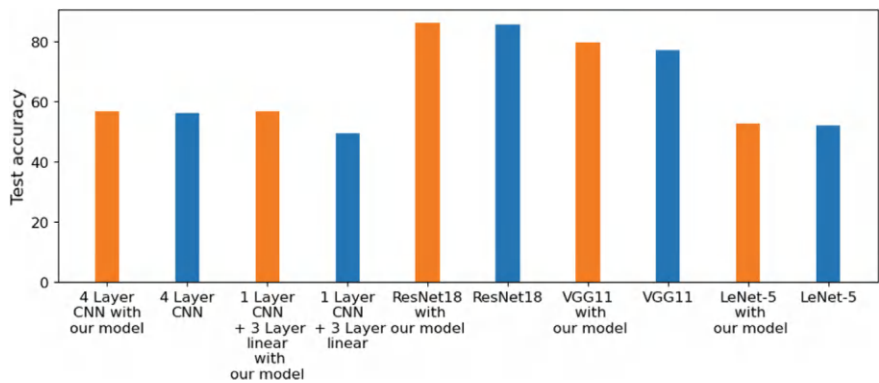


Fig. 9 The comparison of test accuracy

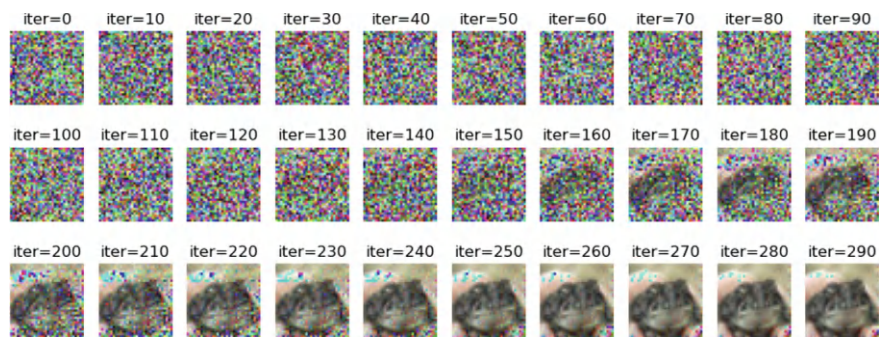


Fig. 10 Leakage test of four-layer CNN

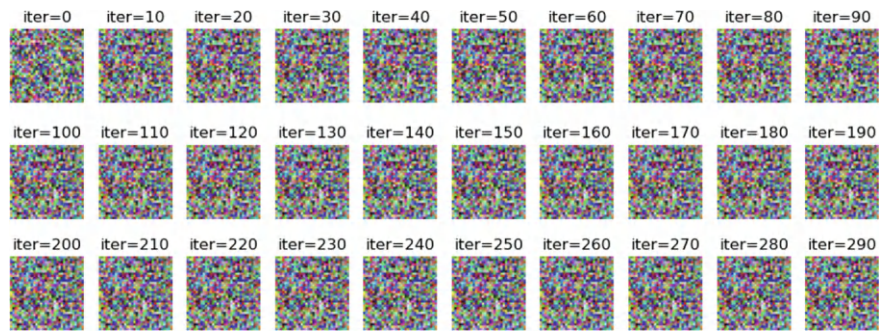


Fig. 11 Leakage test of four-layer CNN connecting to our extension

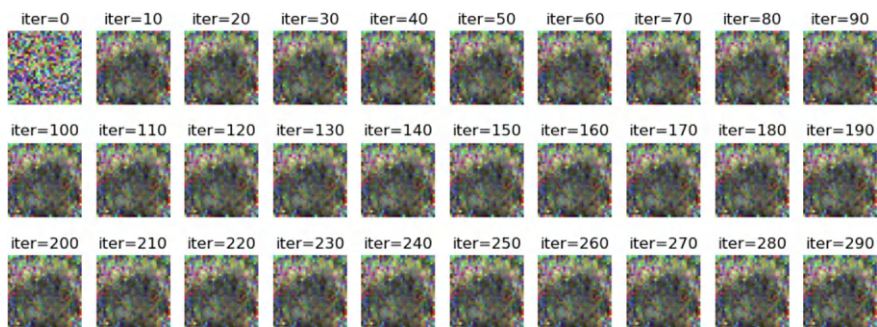


Fig. 12 Leakage test of VGG11

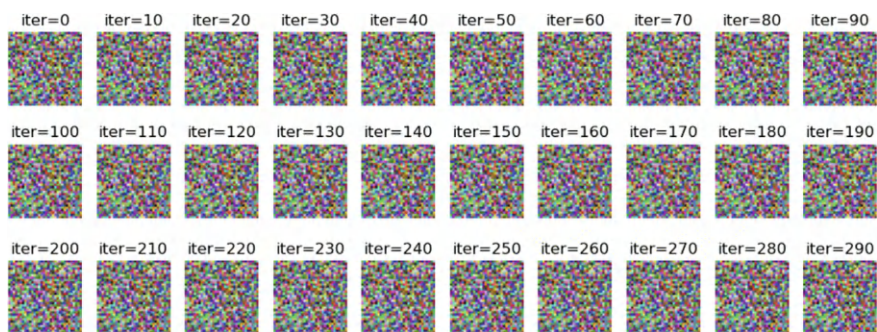


Fig. 13 Leakage test of VGG11 connecting to our extension

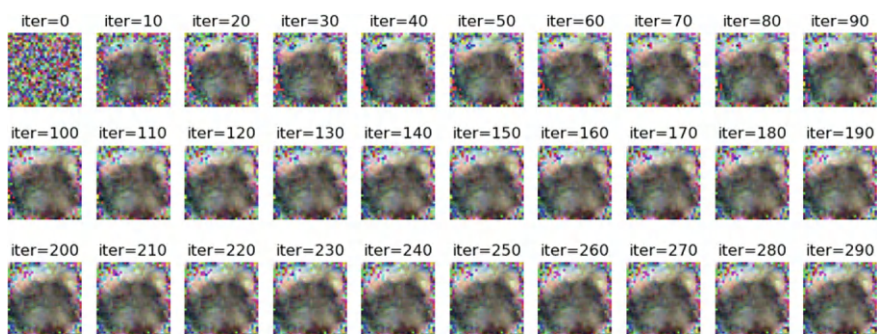


Fig. 14 Leakage test of LeNet-5

5 Conclusions

The proposed extension can be easily attached to most extant CNN models to resist again deep leakage attack by applying computations on gradients. Further, our model can be regarded as an adaptive design since it can be leveraged by many kinds of

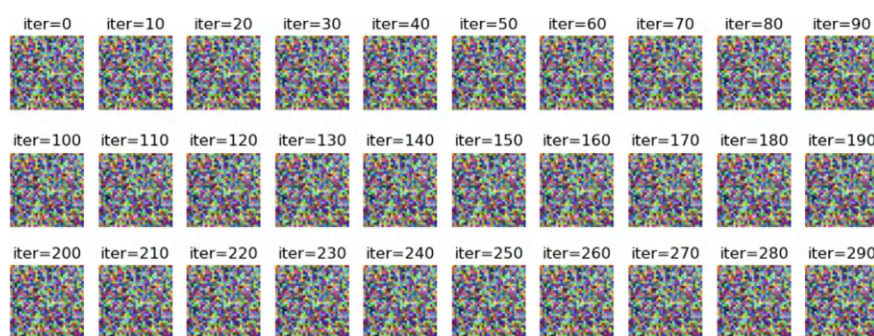


Fig. 15 Leakage test of LeNet-5 connecting to our extension

neural networks and its parameters also can be used to assist the generalization of neural networks in some certain situations. Although some previous method such as gradients compression/pruning, encryption, etc. can effectively protect the leakage of privacy, our model is more ingenious and handy to be implemented on resource constraint devices without expensive burden.

We are planning to implement our lightweight model on IoT devices such as Raspberry Pi in the future to evaluate the execution performance. Some comparisons are also needed to make for different strategies of resisting against leakage attack conducted in the resource constraint environment.

References

1. Konečný J, McMahan HB, Yu FX, Richtárik P, Suresh AT, Bacon D (2016) Federated learning: strategies for improving communication efficiency. In: NIPS workshop on private multi-party machine learning (PMPML'16), Barcelona, Spain. [arXiv:1610.05492](https://arxiv.org/abs/1610.05492)
2. Hitaj B, Ateniese G, Perez-Cruz F (2017) Deep models under the GAN: information leakage from collaborative deep learning. In: Proceedings of the 2017 ACM SIGSAC conference on computer and communications security (CCS'17), October 2017, pp 603–618
3. Geiping J, Bauermeister H, Dröge H (2020) Inverting Gradients—How easy is it to break privacy in federated learning? In: 34th conference on neural information processing systems (NeurIPS 2020), Vancouver, Canada
4. Gong H, Jiang L, Liu X, Wang Y, Gastro O, Wang L, Zhang K, Guo Z (2023) Gradient leakage attacks in federated learning. *Artif Intell Rev* 56:1337–1374
5. Zhu L, Liu Z, Han S (2019) Deep leakage from gradients. In: Advances in neural information processing systems, vol 32 (NeurIPS). Curran Associates, Inc. [arXiv:1906.08935](https://arxiv.org/abs/1906.08935)
6. Lin Y, Han S, Mao H, Wang Y, Dally WJ (2018) Deep gradient compression: reducing the communication bandwidth for distributed training. In: 6th international conference on learning representations (ICLR). [arXiv:1712.01887](https://arxiv.org/abs/1712.01887)
7. Tishby N, Zaslavsky N (2015) Deep learning and the information bottleneck principle. In: 2015 IEEE information theory workshop (ITW), Jerusalem, Israel, pp 1–5
8. Hu EJ, Shen Y, Wallis P, Allen-Zhu Z, Li Y, Wang S, Wang L, Chen W (2022) LoRA: low-rank adaptation of large language models. In: The tenth international conference on learning representations (ICLR)

9. Lecun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. In: *Proceedings of the IEEE*, vol 86, no 11, pp 2278–2324, November 1998
10. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*, vol 25. Curran Associates, Inc.
11. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: *3rd international conference on learning representations (ICLR)*. Computational and Biological Learning Society, pp 1–14
12. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: *2016 IEEE conference on computer vision and pattern recognition (CVPR)*, Las Vegas, NV, USA, pp 770–778
13. Scheliga D, Mäder P, Seeland M (2022) PRECODE—a generic model extension to prevent deep gradient leakage. In: *2022 IEEE/CVF winter conference on applications of computer vision (WACV)*, Waikoloa, HI, USA, pp 3605–3614
14. The CIFAR-10 dataset. <https://www.cs.toronto.edu/~kriz/cifar.html>

PP-3FAT Scheme for 5G-IoHT Telemedicine Systems



Tzu-Wei Lin

Abstract 5G networks have many attractive features which can improve capacity and speed to provide reliable and speedy connectivity to thousands of devices of Internet of Things. Telemedicine relies on mobile healthcare system which includes alarm system, clinical data storage, and wireless communication where transmitted data might be damaged or leaked without proper security measures. This research proposed a privacy-preserved three-factor authentication with time-bound (PP-3FAT) scheme for 5G-IoHT telemedicine systems to achieve security requirements.

Keywords 5G-IoHT · Privacy-preserved · Telemedicine systems · Three-factor authentication · Time-bound

1 Introduction

5G (the fifth generation) networks have been deployed because of ability to provide more speedy, bigger capacity, and more scalability than previous telecommunication technology [1, 2]. 5G networks have been proven that efficiency of energy consumption while providing advanced quality of services in number of devices environments is one of the advantages of 5G networks. In other words, 5G networks bring impact on connected services and devices through higher reliability, connectivity, and cloud storage and improve quality of devices-to-devices (D2D) communication [1, 3].

The Internet of things (IoT) is a distributed network with thousands of arranged devices.

IoT successfully connects objects and human beings by interacting and cooperating with other devices to communicate and exchange information, which is one of the main missions of IoT, and lots of fields have taken advantages above to adopt IoT

T.-W. Lin (✉)

i.School, Feng Chia University, Taichung City, Taiwan

e-mail: tweilin@fcu.edu.tw

Information Security Office, Office of Information Technology, Feng Chia University, Taichung City, Taiwan

for industrial transforming. We focus on one of the IoT applications which combines healthcare systems with portable healthcare devices called Internet of Health Things (IoHT) [2, 4–7].

In this paper, we design a privacy-preserved three-factor authentication with time-bound (PP-3FAT) scheme for 5G-IoHT telemedicine systems to solve security issues above.

2 Related Works

2.1 Telemedicine Systems

Telemedicine system is a technology combining electronic message and telecommunication [8]. Patient sends important, sensitive, and private healthcare information to healthcare services through public and wireless networks [8]. Patients' health condition can be obtained by medical professionals if medical professionals are able to view the information immediately [8]. Because telemedicine system relies on wireless communication which is more vulnerable compared with wire communication, many researchers and adversary on Internet pay attention on data transmission security and related attacks, such as eavesdropping, man-in-the-middle attack, data tempering, etc. [9]. Technical support is not enough though Health Insurance Portability and Accountability Act, General Data Protection Regulation, and Safe Harbor Laws have been made which provide personal information privacy [9]. Telemedicine relies on mobile healthcare system which includes alarm system, clinical data storage, and wireless communication where transmitted data might be damaged or leaked without proper security measures.

2.2 Three-Factor Authentication (3FA)

Passwords have been chosen to use for identification authentication in network and computer applications [10]. Passwords are suggested being as strong as possible, but, generally, users choose weak password, which is short or easy to remember and be guessed easily. Biometric authentication demands a wide range of solutions against increasing cyber-crimes and digital identity thefts and improves weakness of password-based authentication, such as password-missing, weak password, etc. [11]. Three-factor authentication (3FA) has been proposed and utilized for raising security instead of using single factor authentication.

2.3 Chebyshev Chaotic Maps

The relationship between cryptography and chaotic system has been discussed for decades because chaotic system has sensitive dependence on initial conditions, pseudo-randomness, and ergodicity, which can correspond to cryptosystem's properties [12]. If even an initial value has changed slightly, result will be unpredictable [13, 14]. Chaotic system is a complex oscillation [13, 14]. Chaotic system has qualitative change of character of solutions [13, 14]. Above features can be corresponded to confusion and diffusion of cryptosystem [12–16]. Mathematical definitions of Chebyshev chaotic maps can be referred to references [12, 15, 16].

3 Proposed Scheme

Patients and medical professionals may transmit biodata and access healthcare services through desktop computers or mobile devices. Measured biodata is transmitted from wearable healthcare devices and synchronized with applications in smart phone, and then patient can send measured biodata to server patient wants at specified time. Because communication and data transmission between patients and servers in telemedicine systems happen in wire or wireless public networks, privacy of communication and measured biodata transmission should be protected. Moreover, patients and medical professionals should be able to access services in multi-server environments.

We describe PP-3FAT Scheme for 5G-IoHT telemedicine systems. We provide time-bound feature during login and first authentication phase. Server can identify whether user login within authorized time. PP-3FA with time-bound achieves a dynamic identification code that can be traced internally but not externally traceable, anonymous authentication and some other important properties. Proposed scheme includes two actors, user U_i and server S_j . In registration phase, after registering to server S_j and obtains the time bound (t_1, t_2) of authorization, user U_i confirms the secret and public parameters. In login and first authentication phase, user U_i and server S_j mutually authenticate that their identities are legitimate, in which user U_i must be logged within authorized time. After that, user U_i and server S_j establish a shared session key for authentication. The used notations are shown as Table 1.

User U_i needs to register to server S_j as a legitimate user via a secure channel. To complete mutual authentication and session key confirmation and obtain services, user U_i and server S_j perform login and the first-time authentication phase. Phases above are illustrated as Figs. 1 and 2.

Table 1 Notations

Notations	Definitions
ID_i	Identity of user U_i
SID_j	Identity of server S_j
PW_i	Password of user U_i
$\mathfrak{B}_i$	Biometric template of user U_i
k_{ji}^y	Session key
$H(.)$	Collision-resistant one-way hash functions
$E_k(.)/D_k(.)$	A symmetric encryption/decryption algorithm with secret key k
p	Large random primes
$h_k(.)$	Collision-resistance secure one-way keyed chaotic hash function
$\oplus$	Exclusive OR (XOR) operation
t_i, t_s	Time stamp
t_1, t_2	Authorized time bound
MAC_A	Message authentication code algorithm of A

4 Security Analysis

Theoretical analyses are presented to prove that proposed scheme can achieve security properties and resist potential common attacks.

4.1 Preventing Man-In-The-Middle (MITM) Attack

User U_i and server S_j can confirm the message is resent, modified, and replaced or not by checking information in message authentication codes MAC_{S_j} and MAC_{U_i} . User U_i verifies MAC_{S_j} , and server S_j verifies MAC_{U_i} in authenticated key exchange phase of proposed scheme. By this way, adversary cannot modify message authentication codes MAC_{S_j} and MAC_{U_i} without session key.

4.2 Key Confirmation

Proposed scheme provides key confirmation which can guarantee the correctness of the encryption key in the session through message authentication code MAC_{S_j} and MAC_{U_i} . If the adversary wants to obtain session key, the adversary has to solve CMDHP. Moreover, session key is not the same every time because of random number.

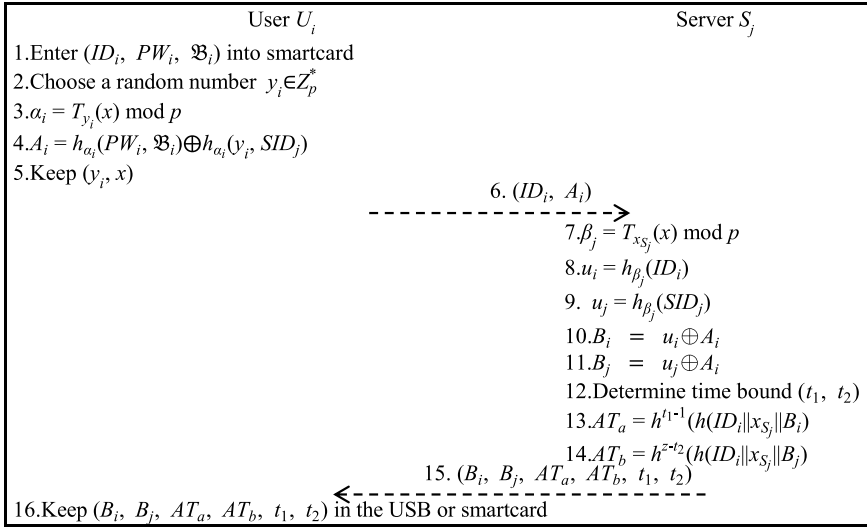


Fig. 2 Registration Phase

4.3 Mutual Authentication

Server S_j encrypts (SID_j, ID_i, μ_i) as message authentication code MAC_{S_j} with session key and sends (MAC_{S_j}, ω_j) to user U_i . User U_i uses ω_j from server S_j to obtain session key and uses it to verify MAC_{S_j} . Server S_j verifies message authentication code MAC_{U_i} . MAC_{S_j} and MAC_{U_i} are included in the session key that only two parties of the communication have, so only user U_i and server S_j can verify each other.

5 Conclusion

We proposed PP-3FAT scheme for 5G-IoHT telemedicine systems, which could achieve preventing MITM attack, key confirmation, and mutual authentication. PP-3FAT scheme for 5G-IoHT telemedicine systems establishes a secure communication channel using authenticated session keys between patients and services of telemedicine systems and allow patients and medical professionals access to multiple telemedicine services with password, smartcard, and biometric feature. Feature of time-bound enable servers to restrict access time of end-point users, which can manage schedule of medical visits and avoid unauthorized or unnecessary access. Moreover, patients have data ownership because patient can control and decide data's destination and time of transmission.

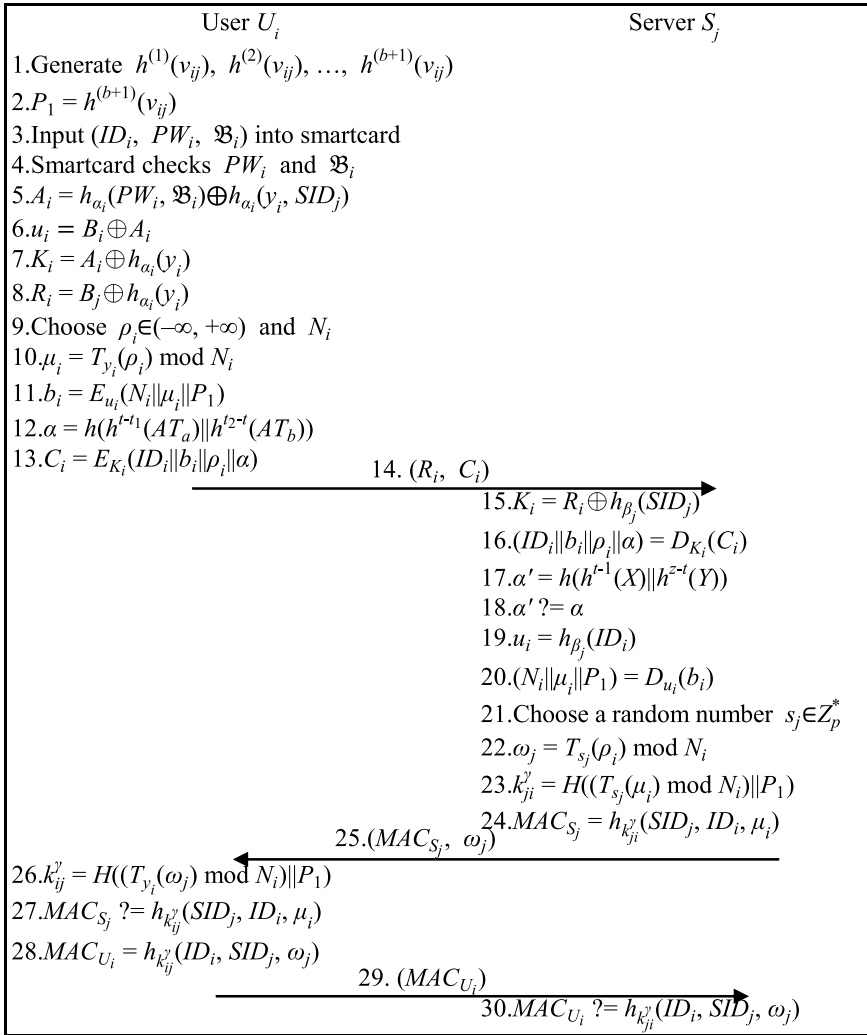


Fig. 3 Mutual authentication and key agreement phase

References

1. Ahad A, Tahir M (2019) Kok-Lim Alvin Yau, 5G-based smart healthcare network: architecture, taxonomy, challenges and future research directions. *IEEE Access* 7:100747–100762
2. Chettri L, Bera R (2020) A comprehensive survey on Internet of Things (IoT) toward 5G wireless systems. *IEEE Internet Things J* 7:16–32
3. Anwar S, Prasad R (2018) Framework for future telemedicine planning and infrastructure using 5g technology. *Wireless Pers Commun* 100:193–208

4. Shaik Jhani Bhasha, P. Sunita, An IoT-based body area network in medical care system: related challenges and issues, *International Journal of Innovative Technology and Exploring Engineering*, 2019.
5. D. Jude Hemanth, Valentina Emilia Balas, *Telemedicine technologies: big data, deep learning, robotics, mobile and remote applications for global healthcare*, Elsevier Science, 2019.
6. Shehzad Ashraf Chaudhry, Azeem Irshad, Muhammad Asghar Khan, Sajjad Ahmad Khan, Summera Nosheen, Ahmad Ali AlZubi, Yousaf Bin Zikria, A lightweight authentication scheme for 6g-iot enabled maritime transport system, *IEEE Transactions on Intelligent Transportation Systems* (2021) 1–10.
7. Shehzad Ashraf Chaudhry (2021) Azeem Irshad, Jamel Nebhen, Ali Kashif Bashir, Nour Moustafa, Yasser D. Al-Otaibi, Yousaf Bin Zikria, An anonymous device to device access control based on secure certificate for internet of medical things systems, *Sustainable Cities and Society* 75:103322
8. Garai Á, Péntek I, Adamkó A (2019) Revolutionizing healthcare with IoT and cognitive, cloud-based telemedicine. *Acta Polytechnica Hungarica* 16:163–181
9. Isra'a Ahmed Zriqat (2016) Ahmad Mousa Altamimi, Security and privacy issues in eHealthcare systems: Towards trusted services. *Int J Comput Sci Appl* 7:229–236
10. Liu Z, Hong Y, Pi D (2014) A large-scale study of web password habits of Chinese network users. *Journal of Software* 9:293–297
11. Kaur H, Khanna P (2020) Privacy preserving remote multi-server biometric authentication using cancelable biometrics and secret sharing. *Futur Gener Comput Syst* 102:30–41
12. Eun-Jun Yoon, Kee-Young Yoo, Cryptanalysis of group key agreement protocol based on chaotic hash function, *IEICE Transactions on Information and Systems*, E94-D (2011) 2167–2170.
13. Kocarev L (2001) Chaos-based cryptography: a brief overview. *IEEE Circuits Syst Mag* 1:6–21
14. Soley D, Janjic P, Kocarev L (2011) Introduction to chaos, *Studies in Computational Intelligence* 354:1–25
15. Lin H-Y (2015) Improved chaotic maps-based password-authenticated key agreement using smart cards. *Commun Nonlinear Sci Numer Simul* 20:482–488
16. Tian-Fu Lee, Chia-Hung Hsiao, Shi-Han Hwang, Tsung-Hung Lin, Enhanced smartcard-based password-authenticated key agreement using extended chaotic maps, *PLoS ONE* 12 (2017).

Cryptography, Blockchain & IoT

Enhanced Lightweight Authenticated Key Agreement Protocol with Privacy Protection for Vehicle Cloud Computing



Tian-Fu Lee and Wen-Hsien Wu

Abstract Vehicle Ad Hoc Networks (VANET) and Internet of Vehicles (IoV) provide the effectiveness and convenience of people in traffic and transportation. They often make a large amount of information transmitted between vehicles and between vehicles and traffic management systems. They also increase the demand for vehicle computing and storage. In order to solve this problem, Vehicle Cloud Computing (VCC) uses a VANET combined with cloud computing technology to assist vehicle drivers for controlling vehicle-related data. Recently, Lee et al. revealed limitations and security problems in previous related studies and developed a more secure and efficient authenticated key agreement protocol for VCC to solve these limitations and problems. Although their study shows a formal security proof of their protocol and resistance to possible attacks, adopting a constant encryption key makes their scheme vulnerable to some possible attacks. This study states the security weaknesses of Lee et al.'s protocol, and develops an enhanced authenticated key agreement protocol for VCC. Compared with related works, the enhanced protocol overcomes the previous weaknesses, provides more security requirements, and retains a lower computational cost.

Keywords VANET · Vehicular cloud computing · Network security · Authentication protocol · Key agreement

T.-F. Lee (✉) · W.-H. Wu

Department of Medical Informatics, Tzu Chi University, No. 701, Zhongyang Road, Sec. 3, Hualien 97004, ROC, Taiwan

e-mail: jackytflee@mail.tcu.edu.tw

W.-H. Wu

e-mail: 111325103@gms.tcu.edu.tw

1 Introduction

For VANET and IoV, drivers and traffic management systems can share and use various services by sharing this information with other connected vehicles. As the scale of the vehicle system increases, the traffic information becomes more and more complex. It often makes a large amount of information transmitted between vehicles and between vehicles and traffic management systems, and increases the demand for vehicle computing and storage. The Vehicular Cloud Computing is proposed for addressing the problems of computing power and storage of mobile devices or vehicles. Authenticated key agreements for VCC allow vehicles to efficiently and quickly construct a communication channel with road-side units, and then to transmit data to the vehicle cloud through the road-side units [1–6]. Therefore, it is very important to develop a secure and efficient authenticated key agreement protocol for VCC.

1.1 The System Model of VCC

The system model of the VCC consists of a trusted authority, vehicles, road-side units and vehicular cloud, as shown in Fig. 1 [5, 6].

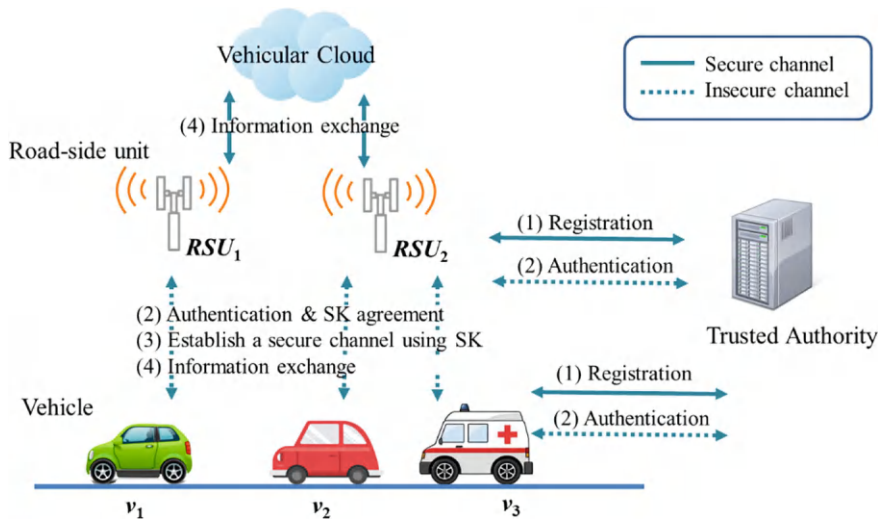


Fig. 1 The system model of the vehicle cloud computing

- (1) **Trusted Authority (TA):** The Trusted Authority is a trusted party and is responsible for setup of system parameters, registering communicating entities including roadside units and mobile devices (or vehicles), and generation of secret keys of roadside units and mobile devices (or vehicles).
- (2) **Vehicle:** A vehicle registers a secret key with TA and stores it in its device on-board units (OBUs) for authentication to securely access roadside units.
- (3) **Road-Side Unit (RSU):** A roadside unit also registers its secret key with TA and stores it in its storage for authentication to generate a common session key with vehicles through the help of the TA. Then RSU and vehicles establish a secure communication channel by using the session key.
- (4) **Vehicular Cloud:** The vehicle cloud is a storage server that saves the data collected and shared by a large number of vehicles in VCC, and performs heavy calculations required by these vehicles to reduce the load on the vehicles because of the low computing performance and small storage space of the OBU in vehicle. Thus, the vehicle needs to securely transmit and exchange data to the vehicle cloud through the RSU.

1.2 Related Works

In 2017, Zhou et al. [7] discussed the assumption of tamper-proof devices (TPDs) and proposed an authentication protocol for vehicle sensor networks, which is based on ECC and preserves privacy protection, to overcome the problems in the previous protocol [8]. In 2019, Wu et al. [9] pointed out that the protocol of Zhou et al. cannot prevent identity guessing and impersonation attacks and still cannot solve privacy-preserving problem. In the same year, Limbasiya et al. [5] presented a message confirmation protocol for VCC environment to address OBU computation limitations in [10]. However, in 2020, Lee et al. [6] showed that the protocol of Limbasyia et al. cannot resist side channel attacks, session key disclosure attacks and impersonation attacks, and fails to provide mutual authentication and privacy-preserving. Lee et al. also proposed an efficient and secure authentication protocol using lightweight hash and exclusive-or operations for VCC as alternatives. Although they have proved the formal security of their proposed protocol and showed that their protocol can resist possible attacks, their protocol encrypts the communication messages with a constant secret key, and directly uses this secret key to encrypt the random number by using exclusive-or operations. Then vehicles and roadside units can obtain each other's secret key for authentication and encryption. It will lead to a malicious user can easily perform impersonation attacks, replay attacks, man-in-the-middle attacks, and derive session keys for other legitimate users and the roadside unit.

1.3 Our Contributions

This study addresses the security weaknesses of the protocol of Lee et al., and develops an enhanced authenticated key agreement protocol for VCC. The enhanced protocol utilizes challenge-response techniques to ensure the freshness of messages, and uses random secrets generated by the trusted authority to protect the secret keys of legitimate participants. Compared with related works, the enhanced authenticated key agreement protocol overcomes the previous weaknesses, provides more security requirements, and retains a lower computational cost. The contributions of this study are summarized as follows.

- (1) This study addresses the security issues of the protocol of Leet et al., which includes failure to withstand impersonation attacks, replay attacks and the man-in-the-middle attacks, and provide session key security.
- (2) An enhanced protocol based on the protocol of Lee et al. is developed to overcome previous security problems.
- (3) The enhanced protocol provides more security properties than related schemes, and retains a lower computational cost.

1.4 Organization of Paper

The rest of the paper is organized as follows. Section 2 briefly reviews the protocol of Leet et al. for VCC. Section 3 discusses and analyzes the weaknesses of the protocol of Leet et al. Section 4 presents an enhanced authenticated key agreement protocol for VCC. The security and performance analyses are also provided in Sect. 5. Finally, Sect. 6 draws conclusions.

2 Preliminaries

This section defines the notations used in this paper, and briefly reviews the protocol of Lee et al. for VCC.

2.1 Notations

Table 1 lists the used notations in this paper. v_i denotes the i -th vehicle in the network; RSU_j denotes the j -th road-side unit; TA denotes the trusted authority.

Table 1 Notations

Notations	Meanings
ID_i	The identity of vehicle v_i
PW_i	The password of vehicle v_i
ID_{RSU_j}	The identity of road-side unit RSU_j
$h(.)$	One-way hash function
$\parallel$	Concatenate operation
$\oplus$	XOR operation

2.2 Review of Lee et al.'s Protocol for VCC

Registration

Step 1: A vehicle v_i selects identity ID_i , password PW_i and random number b_i , computes $PE_i = h(PW_i||b_i)$ and $BE_i = b_i \oplus h(ID_i||PW_i)$, and sends $\{ID_i, PW_i, PE_i, BE_i\}$ to TA .

Step 2: TA randomly selects a_i and s_i for v_i , and computes $AE_i = h(ID_i||PE_i) \oplus a_i$, $HID_i = h(ID_i||PW_i||a_i)$, $HPW_i = h(HID_i||PW_i)$, $MV_i = h(HID_i||h(x||y))$, $A_i = HPW_i \oplus s_i$, $V_i = MV_i \oplus s_i$ and $VS_i = h(HID_i||MV_i||s_i)$, where x is TA 's master key and y is TA 's secret key, respectively. Then TA keeps $(A_i, VI_i, AE_i, BE_i, VS_i)$ in OBU_i , and sends it to v_i .

Step 3: The roadside unit RSU_j selects identity ID_{RSU_j} and a random number a_j and sends $\{ID_{RSU_j}, a_j\}$ to TA .

Step 4: TA computes $RA_j = h(ID_{RSU_j}||a_j)$, $RB_j = h(RA_j||h(x||y))$, and sends $\{RA_j, RB_j\}$ to RSU_j .

Key Agreement and Authentication Phase

Step 1: v_i inputs ID_i , PW_i , and calculates $b_i = BE_i \oplus h(ID_i||PW_i)$, $PE_i = h(PW_i||b_i)$, $a_i = AE_i \oplus h(ID_i||PE_i)$, $HID_i = h(ID_i||PW_i||a_i)$, $HPW_i = h(HID_i||PW_i)$, $MV_i = VI_i \oplus s_i$, $VS_i' = h(HID_i||MV_i||s_i)$. Then v_i checks $VS_i' = ?VS_i$. If successful, v_i randomly selects r_i and computes $Auth_1 = h(r_i||MV_i)$, $M_1 = MV_i \oplus r_i$, and sends $\{Auth_1, M_1, HID_i\}$ to RSU_j .

Step 2: RSU_j randomly selects r_j , computes $B_i = RB_j \oplus r_j$, $Auth_2 = h(ID_{RSU_j}||RB_j||r_j)$, and sends $\{Auth_1, M_1, HID_i, B_i, RA_j, Auth_2\}$ to TA .

Step 3: TA computes $MV_i' = h(HID_i||h(x||y))$, $r_i = M_i \oplus MV_i$, $Auth_1' = h(r_i||MV_i)$, checks $Auth_1' = ?Auth_1$. If successful, TA computes $RB_j = h(RA_j||h(x||y))$ and $r_j = RB_j \oplus B_i$ and $Auth_2' = h(ID_{RSU_j}||RB_j||r_j)$, and checks $Auth_2' = ?Auth_2$. If successful, TA generates a new secret key y_{new} , computes $RB_{j_{new}} = h(RA_j||h(x||y_{new}))$, $C_i = RB_j \oplus r_i$, $D_i = MV_i \oplus r_j$, $E_i = RB_{j_{new}} \oplus r_j$, $Auth_3 = h(RB_j||r_i)$ and $Auth_4 = h(MV_i||r_j)$, and sends $\{C_i, D_i, E_i, Auth_3, Auth_4\}$ to RSU_j .

Step 4: RSU computes $r_i = RB_j \oplus C_i$ and $RB_{jnew} = r_j \oplus E_i$, $Auth_{3'} = h(RB_j || RB_{jnew} || r_i)$ and checks $Auth_{3'} = ?Auth_3$. If successful, RSU updates RB_j as RB_{jnew} , computes $SK = h(r_i || r_j)$ as the session key, and sends the message $\{D_i, Auth_4\}$ to v_i .

Step 5: v_i computes $r_j = MV_i \oplus D_i$, $Auth_{4'} = h(MV_i || r_j)$, and checks $Auth_{4'} = ?Auth_4$. If successful, v_i computes the session key $SK = h(r_i || r_j)$ shared with RSU_j .

Message signature and confirmation phase

Step 1: In order to sign M_i , v_i calculates $\sigma_i = SK \oplus h(M_i || T_1)$ and $M_{i'} = SK \oplus M_i \oplus T_1 \oplus ID_{RSU_j}$, then v_i sends $\{\sigma_i, M_{i'}, T_1\}$ to RSU_j .

Step 2: RSU_j computes $M_i = SK \oplus M_{i'} \oplus T_1 \oplus ID_{RSU_j}$, $\sigma_i' = SK \oplus h(M_i || T_1)$ and checks $\sigma_i' = ?\sigma_i$. If it holds, RSU_j uses M_i for the future computation.

3 Limitations of the Protocol of Lee et al.

In the key agreement and authentication phase of the protocol of Lee et al. [6], a vehicle v_i and the trusted authority TA encrypt the communication messages with a constant secret key, and directly use this secret key to encrypt the random number by using exclusive-or operations, so that both v_i and RSU_j can obtain each other's secret key for authentication and encryption. This will cause the fact that a legitimate and malicious v_i^* can easily deduce the secret keys of roadside units RSU_j and other legitimate v_i , and successfully perform impersonation attacks, replay attacks and the man-in-middle attacks. Even the malicious v_i^* can easily derive the session key of the other legitimate v_i and the roadside unit RSU_j .

3.1 Failure to Impersonation Attacks

Impersonation of a roadside unit

In the Lee et al.'s protocol, after communicating with RSU_j and TA , a malicious v_i^* has r_i^* and r_j , and has capability to derive RB_j from C_i and RB_{jnew} from E_i , where C_i and E_i are public communication messages of RSU_j and TA . However, RB_j and RB_{jnew} are the share key of RSU_j and TA . Thus, a malicious v_i^* can impersonate the roadside unit RSU_j to communicate with other legitimate users v_i and the trusted authority TA .

Impersonation of other legitimate vehicles

After that, the malicious v_i^* can obtain r_i and derive the share key MV_i of v_i and TA . Hence, the malicious v_i^* can impersonate other legitimate users v_i to communicate the roadside unit RSU_j with a trusted authority.

Impersonation of the trusted authority

After that, the malicious v_i^* has v_i 's secret key MV_i and RSU_j 's secret keys RB_j and $RB_{j_{new}}$, and can impersonate the trusted authority TA to communicate with the roadside unit RSU_j and other legitimate users v_i .

3.2 Failure to Replay Attacks

Additionally, the malicious v_i^* can directly send v_i 's previous request $\{Auth_1, M_1, HID_i\}$ to RSU_j , and successfully perform replay attacks. Then, v_i^* can obtain C_i from TA and B_i from RSU_j , and calculate $RB_j = C_i \oplus r_i$ and $r_j = B_i \oplus RB_j$ because that v_i^* has v_i 's random secret r_i . Thus, v_i^* can compute the correct $SK = h(r_i || r_j)$ shared with RSU_j .

3.3 Violation of Session Key Security

Moreover, since the malicious v_i^* has v_i 's random secret r_i and RSU_j 's random secret r_j , v_i^* can easily obtain the session key $SK = h(r_i || r_j)$. Therefore, the Lee et al.'s protocol violates session key security.

3.4 Failure to the Man-in-Middle Attacks

Furthermore, since the malicious v_i^* has v_i 's secret key MV_i and RSU_j 's secret key RB_j for authentication and en/decryption, v_i^* selects random numbers r_i^* and r_j^* , and can perform the man-in-middle attack, so that v_i^* and v_i share $SK^* = h(r_i || r_j^*)$ and v_i^* and RSU_j share $SK^{**} = h(r_i^* || r_j)$. Therefore, the Lee et al.'s protocol cannot withstand the man-in-middle attacks.

3.5 Maintaince of a Verifier Table

In the protocol of Lee et al. [6], TA generates a new secret key y_{new} to compute $RB_{j_{new}}$, where $RB_{j_{new}} = h(RA_j || h(x || y_{new}))$. It will cause TA to maintain a verifier table to store the key y_{new} for each RSU_j .

4 Enhanced Authenticated Key Agreement Protocol for VCC

This section presents an enhanced protocol for VCC, which uses a random secret generated by a trusted authority to prevent legitimate participants' secret keys from being deduced by other participants. The enhanced protocol for VCC is also comprised of the registration phase, and authentication and key agreement phase. The registration phase of the enhanced protocol is similar to that of the protocol of Lee et al., and is not presented here.

The authentication and key agreement phase of the enhanced protocol

Figure 2 demonstrates the authentication and key agreement phase of the enhanced protocol for VCC, and is described in detail as follows.

Step 1: v_i inputs ID_i , PW_i , and calculates $b_i = BE_i \oplus h(ID_i || PW_i)$, $PE_i = h(PW_i || b_i)$, $a_i = AE_i \oplus h(ID_i || PE_i)$, $HID_i = h(ID_i || PW_i || a_i)$, $HPW_i = h(HID_i || PW_i)$, $MV_i = VI_i \oplus s_i$, $VS'_i = h(HID_i || MV_i || s_i)$. Then v_i checks $VS'_i = VS_i$. If successful, v_i randomly selects r_i and computes $Auth_1 = h(r_i || MV_i)$, $M_1 = MV_i \oplus r_i$, and sends $\{Auth_1, M_1, HID_i\}$ to RSU_j .

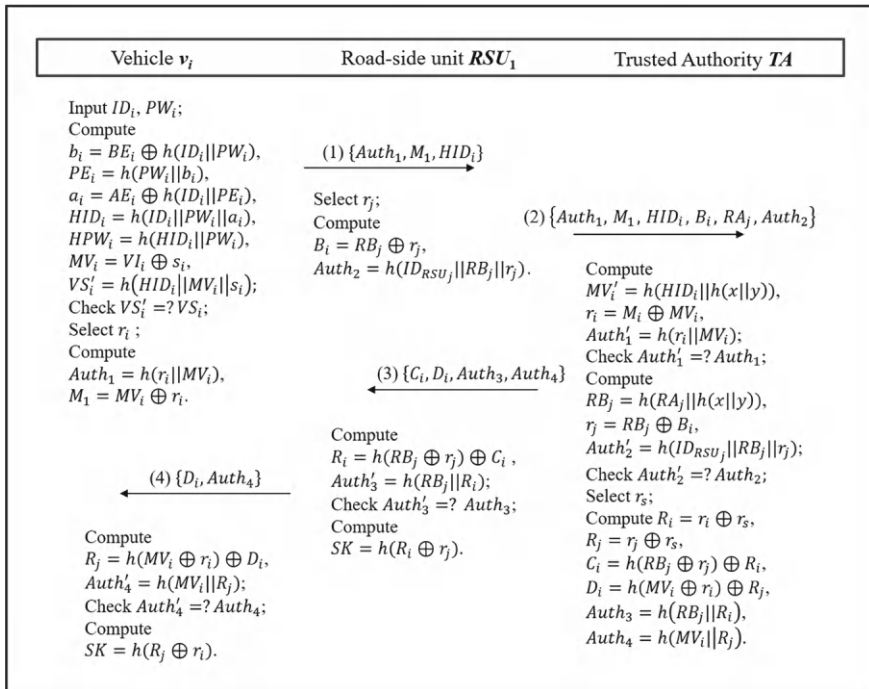


Fig. 2 The authentication and key agreement phase of the enhanced protocol for VCC

Step 2: RSU_j randomly selects r_j , computes $B_i = RB_j \oplus r_j$, $Auth_2 = h(ID_{RSU_j} || RB_j || r_j)$, and sends $\{Auth_1, M_1, HID_i, B_i, RA_j, Auth_2\}$ to TA . RSU_j also acts as a relay station to help transmit messages between v_i and TA .

Step 3: TA computes $MV_{i'} = h(HID_i || h(x || y))$, $r_i = M_i \oplus MV_i$, $Auth_{1'} = h(r_i || MV_i)$ and checks $Auth_{1'} = ?Auth_1$. If holds, TA successfully authenticates v_i . Next, TA computes $RB_j = h(RA_j || h(x || y))$, $r_j = RB_j \oplus B_i$ and $Auth_{2'} = h(ID_{RSU_j} || RB_j || r_j)$, and checks $Auth_{2'} = ?Auth_2$. If holds, TA successfully authenticates RSU_j . Then TA randomly selects r_s , computes $R_i = r_i \oplus r_s$, $R_j = r_j \oplus r_s$, $C_i = h(RB_j \oplus r_j) \oplus R_i$, $D_i = h(MV_i \oplus r_i) \oplus R_j$, $Auth_3 = h(RB_j || R_i)$ and $Auth_4 = h(MV_i || R_j)$, and sends $\{C_i, D_i, Auth_3, Auth_4\}$ to RSU_j .

Step 4: RSU computes $R_i = h(RB_j \oplus r_j) \oplus C_i$ and $Auth_{3'} = h(RB_j || R_i)$ and checks $Auth_{3'} = ?Auth_3$. If holds, RSU_j successfully authenticates TA and v_i . Next, RSU computes $SK = h(R_i \oplus r_j)$ as the session key, and sends $\{D_i, Auth_4\}$ to v_i .

Step 5: v_i computes $R_j = h(MV_i \oplus r_i) \oplus D_i$, $Auth_{4'} = h(MV_i || R_j)$, and checks $Auth_{4'} = ?Auth_4$. If holds, v_i successfully authenticates TA and RSU_j . Next, v_i computes the session key $SK = h(R_j \oplus r_i)$ shared with RSU_j .

5 Security and Performance Analyses

This section analyzes the security of the enhanced protocol, compares functionality of the enhanced protocol with that of related works, and provides performance analysis and comparisons.

5.1 Security Analysis of the Enhanced Protocol

Correctness

RSU computes the session key $SK = h(R_i \oplus r_j) = h(r_i \oplus r_s \oplus r_j)$ in Step 4, and v_i computes the session key $SK = h(R_j \oplus r_i) = h(r_j \oplus r_s \oplus r_i)$ in Step 5. Thus, RSU and v_i obtain the same session key SK in the protocol.

Providing mutual authentication

TA authenticates v_i by validating $Auth_1 = h(r_i || MV_i)$ and authenticates RSU_j by validating $Auth_2 = h(ID_{RSU_j} || RB_j || r_j)$. RSU_j authenticates TA and v_i by validating $Auth_3 = h(RB_j || R_i)$. v_i authenticates TA and RSU_j by validating $Auth_4 = h(MV_i || R_j)$. Therefore, the enhanced protocol provides mutual authentication.

Providing session key security

The random numbers r_i of v_i are protected with MV_i in M_1 and with $h(RB_j \oplus r_j)$ and r_s in C_i ; the random numbers r_j of RSU_j is protected with RB_j in B_i and with

$h(MV_i \oplus r_i)$ and r_s in D_i ; and the random numbers r_s of TA is protected with r_i in R_i and with r_j in R_j . Except for TA , only v_i can get $R_j (= r_j \oplus r_s)$, and RSU_j can get $R_i (= r_i \oplus r_s)$, respectively, because of the one-way property of hash functions, and thus only v_i and RSU_j can compute a secure $SK = h(r_i \oplus r_s \oplus r_j)$ as the session key.

Providing privacy protection

The enhanced protocol protects v_i 's real identity ID_i with the shadow identity HID_i , and protects RSU_j 's real identity ID_{RSU_j} with the shadow identity RB_j , where $HID_i = h(ID_i || PW_i || a_i)$, $RA_j = h(ID_{RSU_j} || a_j)$ and $RB_j = h(RA_j || h(x || y))$. The real identities ID_i of v_i and ID_{RSU_j} of RSU_j are not revealed, and the enhanced protocol thus provides privacy protection.

Withstanding impersonation attacks

An attacker including the malicious v_i^* cannot correctly compute $M_1 = MV_i \oplus r_i$ and $Auth_1 = h(r_i || MV_i)$ without v_i 's secret keys MV_i , and send out $\{Auth_1, M_1, HID_i\}$ to impersonate a legitimate vehicle. The attacker cannot correctly compute $B_i = RB_j \oplus r_j$, $Auth_2 = h(ID_{RSU_j} || RB_j || r_j)$ without RSU_j 's secret keys RB_j , and send out $\{Auth_1, M_1, HID_i, B_i, RA_j, Auth_2\}$ to impersonate a roadside unit. Similarly, the attacker cannot correctly compute $C_i = h(RB_j \oplus r_j) \oplus R_i$, $D_i = h(MV_i \oplus r_i) \oplus R_j$, $Auth_3 = h(RB_j || R_i)$ and $Auth_4 = h(MV_i || R_j)$ without MV_i and RB_j , and send out $\{C_i, D_i, Auth_3, Auth_4\}$ to impersonate the trusted authority. Thus, the enhanced protocol withstands impersonation attacks.

Withstanding replay attacks

The enhanced protocol utilizes random numbers r_i of v_i and r_j of RSU_j and challenge-response techniques to ensure the freshness of messages. Additionally, the enhanced protocol uses r_s randomly generated by TA to protect r_i and r_j , so as to prevent v_i (or RSU_j) from being able to obtain r_j (or r_i), and then deduce the other party's secret key. Therefore, the enhanced protocol withstands replay attacks.

Withstanding the Man-In-Middle Attacks

The enhanced protocol provides mutual authentication and ensures the freshness of messages. Then, an adversary is hard to perform the man-in-middle attacks.

5.2 Functionality Comparisons

Table 2 compares the security properties of the enhanced protocol with those of the protocols of Limbasiya et al. [5] and Lee et al. [6]. The protocol of Limbasiya et al. has a correctness problem, fails to provide mutual authentication and privacy protection, and cannot resist impersonation attacks. The protocol of Lee et al. [6] encrypt the communication messages with a constant secret key, and directly use this secret key to encrypt the random number by using exclusive-or operations, so that their protocol

Table 2 Security comparison

Protocols	P ₁	P ₂	P ₃	P ₄	P ₅	P ₆	P ₇
Limbasiya et al. [5]	×	×	✓	×	✓	×	✓
Lee et al. [6]	✓	✓	×	✓	×	×	×
Enhancement	✓	✓	✓	✓	✓	✓	✓

✓: the security property is provided. × : the security property is not provided

P₁: Providing correctness; P₂: Providing mutual authentication

P₃: Providing session key security; P₄: Providing privacy protection

P₅: Withstanding replay attacks; P₆: Withstanding impersonation attacks

P₇: Withstanding the man-in-middle attacks

violates session key security and cannot resist replay attacks, impersonation attacks and the man-in-middle attacks. The enhanced protocol provides most of the security requirements, is resistant to possible attacks, and is superior to related protocols in terms of security.

5.3 Performance Analysis and Comparisons

Table 3 compares the computational cost of the enhanced protocol with those of the protocols of Limbasiya et al. [5] and Lee et al. [6], where T_h denotes the response time of executing a hash function; and T_{ecc} denotes the response time of executing a scale multiplication on elliptic curve cryptography (ECC). The simulation environment includes Intel® Core™ i5-3317U @ 1.70GHz CPU; Windows 10; 4G Memory; python3; Hash function- MD5 and scalar multiplication on ECC for analyzing the response time.

The enhanced protocol is based on the protocol of Lee et al. [6] and uses more hash operations in order to eliminate the problem of encryption with a constant secret key. However, the enhanced protocol eliminates the problem of the protocol of Lee et al. [6] to update the temporary identity $RB_{j_{new}}$, so the required calculations are reduced. Therefore, the enhanced protocol still maintains lightweight operation close to the protocol of Lee et al. [6]. Although the protocol of Limbasiya et al. [5] does not require an on-line TA, their protocol needs to maintain complex public key cryptography and requires time-consuming scale multiplications on ECC so that their protocol is inefficient in computation. Hence, the enhanced protocol solves

Table 3 Performance comparison

Schemes	v_i	RSU_j	TA	Total computation/response time
Limbasiya et al. [5]	$2 T_h + 2 T_{ecc}$	$2 T_h$	—	$4 T_h + 2 T_{ecc} / 1.4388 \text{ ms}$
Lee et al. [6]	$9 T_h$	$3 T_h$	$9 T_h$	$21 T_h / 0.0441 \text{ ms}$
Enhancement	$10 T_h$	$4 T_h$	$9 T_h$	$23 T_h / 0.0483 \text{ ms}$

previous security problems, retains a low computational cost and is lightweight in computation.

6 Conclusion

This study reviews the system model for VCC, and states that the Lee et al.'s protocol for VCC may be subject to some potential attacks and violates the session key security. To solve the previous weaknesses, an enhanced protocol is developed for VCC, which utilizes challenge-response techniques to ensure the freshness of messages, and uses *TA*'s random secrets to protect legal participants' secret keys. Compared with related works, the enhanced authenticated key agreement protocol overcomes the previous weaknesses, provides more security requirements, and retains a lower computational cost. Therefore, it is more suitable for the practical application of VCC.

Acknowledgements This work was supported by the Ministry of Science and Technology of the Republic of China, Taiwan, under Contract No. NSTC 110-2221-E-320-005-MY2 and Buddhist Tzu Chi Medical Foundation, Taiwan, under the grants TCMMP 112-02-03.

References

1. Zhang L, Wu Q, Domingo-Ferrer J, Qin B, Hu C (2017) Distributed aggregate privacy-preserving authentication in VANETs. *IEEE Trans Intell Transp Syst* 18:516–526
2. Wazid M, Das AK, Kumar N, Odelu V, Reddy AG, Park K, Park Y (2017) Design of lightweight authentication and key agreement protocol for vehicular ad hoc networks. *IEEE Access* 5:14966–14980
3. Liu Z, Xiong L, Peng T, Peng D, Liang H (2018) A realistic distributed conditional privacy-preserving authentication scheme for vehicular ad hoc networks. *IEEE Access* 6:26307–26317
4. Zhang J, Cui J, Zhong H, Chen Z, Liu L (2019) PA-CRT: Chinese remainder theorem based conditional privacy-preserving authentication scheme in vehicular ad-hoc networks. *IEEE Trans Dependable Secur Comput* 2019
5. Limbasiya T, Das D (2019) Secure message confirmation scheme based on batch verification in vehicular cloud computing. *Physical Commun.* 34:310–320
6. Lee J, Yu S, Kim M, Park Y, Lee S, Chung B (2020) Secure key agreement and authentication protocol for message confirmation in vehicular cloud computing. *Appl Sci* 10:6268. <https://doi.org/10.3390/app10186268>
7. Zhou Y, Zhao X, Jiang Y, Shang F, Deng S, Wang X (2017) An enhanced privacy-preserving authentication scheme for vehicle sensor network. *Sensors* 17:2854
8. Chuang CM, Lee FJ (2014) TEAM: trust-extended authentication mechanism for vehicular ad hoc networks. *IEEE Syst J* 8:749–758
9. Wu L, Sun Q, Wang X, Wang J, Yu S, Zou Y, Liu B, Zhu Z (2019) An efficient privacy-preserving mutual authentication scheme for secure V2V communication in vehicular ad hoc network. *IEEE Access* 7:55050–55063
10. Zhong H, Wen J, Cui J, Zhang S (2016) Efficient conditional privacy-preserving and authentication scheme for secure service provision in VANET. *Tsinghua Sci Technol* 21:620–629

An Extended Euclidean Algorithm Implementation over Ciphertext Space



Po-Wen Chi, Yuan-Pang Lin, Ming-Hung Wang, and Yun-Hsin Chuang

Abstract Fully homomorphic encryption (FHE) is a useful cryptographic primitive. It makes a third-party manipulate data in the ciphertext space and therefore is a key technology for privacy-preserving outsourcing services. Theoretically, FHE can evaluate any circuits. However, the implementation is not trivial. The reason it lacks lots of basic operations like conditional check and branch control. In this paper, we focus on extended Euclidean algorithm operation over FHE. We introduce the problems we encounter and show how we solve these issues. We implement our design and evaluate its performance on a real computer. Finally, we propose some research directions for our future work.

Keywords Fully homomorphic encryption · Extended euclidean algorithm · Secure computation

P.-W. Chi (✉) · Y.-P. Lin

Department of Computer Science and Information Engineering, National Taiwan Normal University, Taipei, Taiwan, R.O.C.

e-mail: neokent@gapps.ntnu.edu.tw

Y.-P. Lin

e-mail: 61147001s@gapps.ntnu.edu.tw

M.-H. Wang

Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan, R.O.C.

e-mail: tonymhwang@cs.ccu.edu.tw

Y.-H. Chuang

Engineering Management Graduate Program, College of Engineering, National Cheng Kung University, Tainan, Taiwan, R.O.C.

e-mail: yhchuang@gs.ncku.edu.tw

1 Introduction

Outsourcing services are more and more popular. People are get used to outsource their own data and enjoy the customized services. With only a lightweight device and network access, a user can access ubiquitous services anytime and anywhere. However, this brings a new critical challenge about user privacy: the user data is leased to a third party. Undoubtedly, users can protect their data by traditional encryption techniques and make the data unknown to the service provider. However, without the knowledge of user data, most outsourcing services cannot work. Nowadays, most service providers ask users to sign agreements before using services, like Google Privacy Policy [1]. In this approach, we assume that the service provider is trusted. However, the service provider technically can make use of the data illegally.

Fully homomorphic encryption (FHE) may be a good technique for the third party to provide services while keeping user privacy at the same time. FHE is a cryptographic primitive that allows computations on the ciphertext space instead of the message space [18]. That is, the service provider can operate encrypted user data, derive the result in the ciphertext form and give the encrypted computational result to the user. After decrypting the response from the service provider, the user can get the wanted data, which is treated as the service output, directly. In this process, the service provider learns nothing about user data and user privacy is kept. The concept of FHE was first proposed in 1978 [19] but the FHE implementation had been an open question until 2009. In 2009, Gentry proposed the first FHE scheme [15] and lots of FHE works have been proposed in recent years [6, 9–11, 14, 16].

Theoretically, FHE supports any functions over ciphertexts, which may be called circuits in other paper. However, the function implementation over FHE may be far from triviality. There are some reasons as follows:

- **Division:** Division is a basic arithmetic operation. Unfortunately, most FHE schemes for integers support only addition and multiplication. This makes lots of functions hard to be implemented. Moreover, modulus operation, which is very useful in computer science, is also absent.
- **Conditional Control:** Conditional control is a basic grammar in programming, like if-statement in C language. When a third party does computations over the ciphertext space, it should learn nothing about the branch result or data will be somewhat leaked [13]. This makes the function implementation difficult.
- **Loop:** Loop is also a kind of conditional control mechanism and it also have the same problem as described above. This raises a new problem: when should the loop be terminated? The loop stop point may also leak the user information.

According to the above reasons, lots of functions may not be easily implemented with FHE. In this work, we focus on the extended Euclidean algorithm implementation. Extended Euclidean algorithm is very useful, especially in the cryptographic field. With the algorithm, we can easily calculate the greatest common divisor (GCD) and multiplicative modular inverse. When implementing extended

Euclidean algorithm, division and iteration¹ are unavoidable operations. We will show our implementation in this paper. Our contributions are as follows:

- We implement extended Euclidean algorithm implementation with OpenFHE [4]. We use TFHE scheme [10] and build division and modulus operation. We integrate these operational blocks and construct a GCD calculator and a modular multiplicative inverse calculator.
- We evaluate our work and see the relevance between the data size and the computational time.
- We propose some applications for FHE-based extended Euclidean algorithm implementation.

The rest of our paper is organized as follows. We briefly introduce homomorphic encryption in Sect. 2. Our proposed implementation is described in Sect. 3. In Sect. 4, we evaluate the computational performance of our implementation. We discuss some application scenarios for this implementation in Sect. 5. Finally, we give our conclusions and future works in Sect. 6.

2 Introduction to Fully Homomorphic Encryption

In this section, we will briefly introduce fully homomorphic encryption techniques. We first introduce some important FHE works and then give a formal definition for a public key FHE.

2.1 Related Works

Homomorphic encryption, which was originally proposed by Rivest et al. [19], allows operations can be evaluated over a series of ciphertexts without the necessity of decrypting ciphertexts first. The term **fully homomorphic** means that the encryption scheme allows arbitrarily many circuit gates² of unbounded depth on ciphertexts. Though the idea was proposed very early, the construction approach had been kept unknown until 2009.

Gentry constructed the first fully homomorphic encryption scheme based on ideal lattices in 2009 [15]. The scheme supports ciphertext addition and multiplication at the same time. Gentry also introduced a concept called bootstrapping, which can reduce the noise generated in the operation and can make the unbounded operation depth possible. In 2010, Gentry et al. applied the same concept over integers and built a new FHE scheme [11].

¹ Most people implement extended Euclidean algorithm with recursion. However, since iteration and recursion are equally powerful, here we use iteration.

² Generally speaking, at least two gates are required.

Though Gentry's scheme sounds reasonable, the scheme is computationally inefficient. The main computational cost comes from the multiplication operation and the bootstrapping process. In 2011, Brakerski and Vaikuntanathan applied the similar idea on LWE (Learning with Errors) and RLWE (Ring-LWE) to build FHE schemes for better multiplication performance [7, 8]. Later, Brakerski, Gentry and Vaikuntanathan defined an idea of the leveled FHE, which avoids the computationally expensive bootstrapping process by limiting the circuit depth [6]. In 2013, Gentry, Sahai and Waters further enhanced the scheme by replacing the Kronecker product with matrix multiplication, making the ciphertext size smaller [16].

Ducas and Micciancio introduced a new homomorphic approach which can compute the NAND gate result of two ciphertexts [12]. Since NAND is functionally complete, this approach can compose arbitrary functional circuits. In a subsequent work, Chillotti, Gama, Georgieva and Izabachène improved Ducas and Micciancio scheme by using a different bootstrapping technique. The scheme is built on Torus and is generally called TFHE [10]. Compared to previous schemes, which focus on arithmetic operations, TFHE provides binary circuit operations over ciphertexts.

In 2017, Cheon, Kim, Kim and Song proposed a new FHE scheme which is generally called CKKS for abbreviation [9]. The scheme does not require the accurate homomorphic operation result. Instead, it allows approximate result and this feature is suitable for the floating point operations, since rational numbers are also approximate values in computer science.

In this paper, we will use TFHE scheme for our implementation. The detailed reason is described in Sect. 3.

2.2 Definition

A public key homomorphic encryption scheme $\mathcal{E}$ is generally composed of a set of probabilistic polynomial-time (PPT) algorithms as follows [18]:

- **KeyGen**(1^λ) $\rightarrow \{pk, sk, evk\}$: the algorithm takes a security parameter λ as input and returns the public key pk , the private key sk and the evaluation key evk . evk is required to perform homomorphic operations over ciphertexts.
- **Enc**(pk, m) $\rightarrow c$: the algorithm takes the public key pk and the message m as input and outputs the ciphertext c .
- **Dec**(sk, c) $\rightarrow m$ or $\perp$: the algorithm takes the private key sk and the ciphertext c as input. If the ciphertext can be correctly decrypted, the algorithm return the message m ; otherwise, the algorithm returns $\perp$.
- **Eval**($evk, f, (c_0, c_1, \dots, c_{t-1})$) $\rightarrow c_f$: the algorithm takes the evaluation key evk , the function f and a set of ciphertexts as input. The output is also in the ciphertext space and the probability

$$Pr[\mathbf{Dec}(sk, c_f) \neq f(m_0, m_1, \dots, m_{t-1}) | c_i = \mathbf{Enc}(pk, m_i), i = 0, \dots, t-1]$$

is negligible.

We call a FHE multi-hop homomorphic if functions can be homomorphically evaluated one by one on a series of ciphertexts.

3 Our Extended Euclidean Algorithm Implementation Through FHE

In this section, we will show our extended Euclidean algorithm construction over the ciphertext space. We first review extended Euclidean algorithm. Then we explain our preferred FHE algorithm in our construction. We will show our modulus and iteration implementation in Sects. 3.3 and 3.4 respectively. Finally, we will integrate our proposed method to implement extended Euclidean algorithm over the ciphertext space.

3.1 Extended Euclidean Algorithm

Extended Euclidean algorithm is an extension to Euclidean algorithm. Extended Euclidean algorithm is a way to find the solution of the modular multiplicative inverse problem.

Algorithm 1: Extended Euclidean Algorithm

Input: a, b
Output: s, t, r
 $(r, r_{new}) \leftarrow (a, b);$
 $(s, s_{new}) \leftarrow (1, 0);$
 $(t, t_{new}) \leftarrow (0, 1);$
while $r_{new} \neq 0$ **do**
 $q \leftarrow r / r_{new};$
 $(r, r_{new}) \leftarrow (r_{new}, r - q \times r_{new});$
 $(s, s_{new}) \leftarrow (s_{new}, s - q \times s_{new});$
 $(t, t_{new}) \leftarrow (t_{new}, t - q \times t_{new});$
end

From extended Euclidean algorithm, we will get an equation $a \cdot s + b \cdot t = r$, where r is the greatest common divisor of (a, b) . If $r \neq 1$, the modular multiplicative inverse doesn't exist, otherwise s is the solution of $x \equiv a^{-1} \pmod{b}$.

3.2 Our Choice in FHE Schemes

There are several FHE schemes, BGV [6], CKKS [9], TFHE [10], and so on. Both the BGV scheme and the CKKS scheme supports addition and multiplication, and TFHE supports binary operations like AND, OR, and XOR. Neither of them supports division nor modulo operation. To overcome this problem, we treat all numbers as a bit array and construct a binary divider. We do not rely on FHE-based arithmetic operations because those schemes do not have zero equality check. This makes division hard to be implemented. As for the binary operations, since the operand's bit size is well known, it is easier to control the iteration number. For this reason, in this work we prefer TFHE scheme as our FHE base. We transpose all numbers into binary and encrypt each bit separately. We use the constructed divider and logical gates to implement the modulus operation, which is described in the following subsection.

3.3 Modulus Operation Implementation

To implement extended Euclidean algorithm in the TFHE scheme, we first construct a divider with logic gates. Figure 1 shows a 4-bits divider. The dividend $A = a_7a_6a_5a_4a_3a_2a_1a_0$ is an 8-bits number, and the divisor $B = b_3b_2b_1b_0$, quotient $Q = q_3q_2q_1q_0$, and remainder $R = r_3r_2r_1r_0$ are 4-bits number. In this case, we have 4 cells and 4 iterations. The number of cells depends on the length of divisor $|B|$, and the number of iterations depends on $|A| - |B|$. Each cell consists of a FS(full subtractor) and a MUX gate. In each iteration, the FSs inside the cell compute from the lowest bit (right hand side on the figure) to the highest bit(left hand side on the figure) then the OR gate computes a bit of Q . After that, the MUX gates inside the cell compute and send the results to the next iteration. In the last iteration (downside of the figure), the results become R . The divider can compute the quotient and the remainder now, but there's still a problem. If the highest bit of the divisor is 0, it will give us the wrong quotient and the wrong remainder. It is a problem that we may align the dividend and the divisor in the wrong way. Suppose we're dividing 27 by "06", we always ignore the zero and write it as $27 \div 6$. Since the bits are ciphertexts, we don't know whether the bit is 0 or not without decryption. Due to this problem, we will temporarily make the dividend longer during the division. By filling ciphertexts of 0(encrypted by the public key), we get a temporary dividend A' where $A' = A$ and $|A'| = |A| + |B| - 2$. The reason we make its length be $|A| + |B| - 2$ is to make the lowest two bits of B align with the highest two bits of A' since our divider always aligns the dividend and the divisor with their highest bit. It increases the number of iteration from $|A| - |B|$ to $|A'| - |B| = |A| - 2$ in exchange of solving this problem.

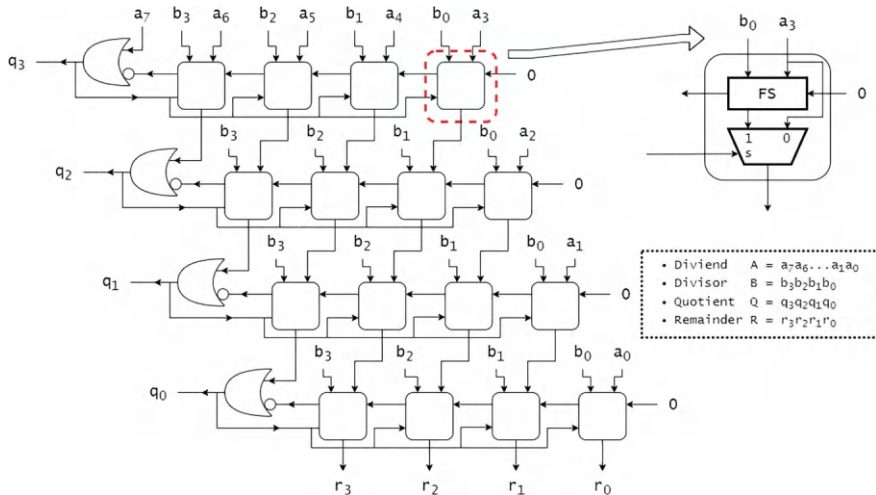
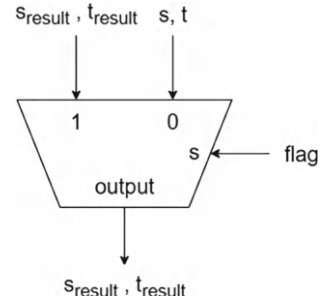


Fig. 1 Our divider diagram

3.4 Iteration Issue

Besides the difficulty dividing, condition control is still a big problem. Recall the end condition of the extended Euclidean algorithm, we end it when $r_{new} = 0$. There's no way to check whether r is 0 or not, which makes the algorithm unstopable. To prevent the algorithm from running forever, we set a boundary to limit the number of iteration. Like the Euclidean algorithm, the extended Euclidean algorithm can be ended in limited iterations [17]. To compute the boundary, we first find out that the smallest (a, b) that needs N iterations is (F_{N+2}, F_{N+1}) , where F_m is the m th Fibonacci number. By induction, we can prove that it is true for $N \in \mathbb{N}$. In other words, for the number $b \leq F_{N+1}$, the algorithm can be ended in N iterations. Take $b = F_{N+1}$, $b = F_{N+1} > \varphi^{N-1}$, where φ is the golden ratio. $b > \varphi^{N-1}$, so $\log_{\varphi} b > N - 1$, $\frac{\log_2 b}{\log_2 \varphi} > N - 1$, then $\log_{\varphi} 2 \times \log_2 b > N - 1$. Now, we know $N < \log_{\varphi} 2 \times \log_2 b + 1 \approx 1.44 \log_2 b + 1$. $\log_2 b$ is $|b|$, so we let the loop run $1.5|b| + 1$ times to make sure the algorithm has done its work. Despite the ending condition, the next iteration after $r_{new} = 0$, there will be a problem that r is going to divide by 0. We use a MUX gate to solve this problem. In the beginning, we declare a variable named flag = 0 then encrypt it into ciphertext. During every iteration, it updates itself by $\text{flag} = \bigwedge_{i=0}^n \bar{r}_i \vee \text{flag}$, where r_i represents the i th bit of r . flag will change from 0 to 1 when $r = 0$ and never changes its value again. Shown in Fig. 2, the return value of (s, t) will update every iteration until $r = 0$, and never change themselves in the remaining iterations. Since the return values stop changing, dividing by zero is not a problem.

Fig. 2 Our MUX gate in the extended Euclidean algorithm



3.5 Integration

We choose OpenFHE to implement our extended Euclidean algorithm. We use the Boolean FHE which supports several logic gates like OR, AND, XOR, and so on. Our extended Euclidean algorithm is shown in Algorithm 2. For computing the modulus inverse, we suppose that $r = 1$, what means that r is no need to return. The major difference between the extended Euclidean algorithm and ours are almost described in the last section. One thing to mention is that our divider can compute the quotient and remainder at the same time, so there will be a little difference when assigning value to r and r_{new} .

Algorithm 2: Our Extended Euclidean Algorithm

Input: a, b

Output: s_{result}, t_{result}

$(r, r_{new}) \leftarrow (a, b);$

$(s, s_{new}) \leftarrow (1, 0);$

$(t, t_{new}) \leftarrow (0, 1);$

$flag \leftarrow 0;$

for $i \leftarrow 0$ **to** $1.5 * |b| + 1$ **by** 1 **do**

$q, r \leftarrow r \text{ div } r_{new};$

/* q is the quotient and r is the remainder */

$flag \leftarrow \bigwedge_{i=0}^n \bar{r}_i \vee flag;$

$(r, r_{new}) \leftarrow (r_{new}, r);$

$(s, s_{new}) \leftarrow (s_{new}, s - q \times s_{new});$

$(t, t_{new}) \leftarrow (t_{new}, t - q \times t_{new});$

$s_{result} \leftarrow MUX(s, s_{result}, flag);$

$t_{result} \leftarrow MUX(t, t_{result}, flag);$

end

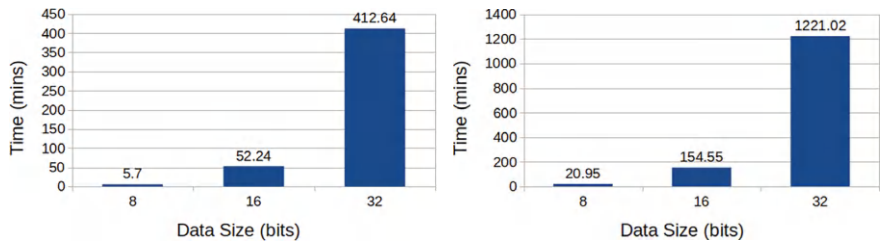


Fig. 3 Computational Time versus Data Size

4 Performance Evaluation

4.1 Experimental Environment

In this section, we will describe our experimental environment. We use a desktop PC with INTEL i7-10700kf and 32 GB RAM. The operating system is Ubuntu 22.04.3. The version of OpenFHE we used is v1.0.4. Note that OpenFHE provides a standard Hardware Abstraction Layer (HAL) designed to support different hardware acceleration technologies [4]. Unfortunately, there is no corresponding hardware in our lab. Therefore, the following experimental results are recorded purely based on the CPU computation.

4.2 Computational Time Versus Different Data Sizes

In this subsection, we will compare the computational time between different integer sizes. Since we use binary FHE, an integer is split into bits and each bit is encrypted independently. It is obvious that larger integer size will require much more time. In our experiments, we prepare three kinds of integer sizes, 8, 16 and 32 bits. Note that the integer size here is the memory size we use to store a number. The experiments include two operations, GCD calculation and modular multiplicative inverse calculation. The key size is 108 bits. The result is shown in Fig. 3.

In Fig. 3, we can see that 32 bits integer requires the longest computational time in both applications, which is the same as our guess. Extended Euclidean algorithm needs more time than GCD calculation because it requires additional operations in each iteration. Note that the experiments show that it takes lots of time to run these simple applications which implies that general FHE-based applications are not practical. Currently, an application which takes no conditional control and no loops is suitable to be implemented with FHE techniques. We believe that performance enhancement is an important issue in FHE technologies.

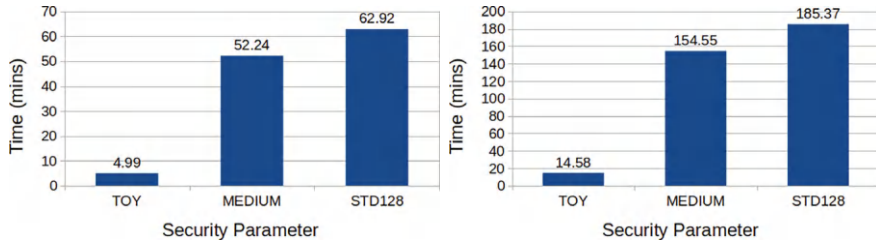


Fig. 4 Computational time versus key size

4.3 Computational Time Versus Different Key Sizes

In this subsection, we will compare the computational time between different key sizes. Key size is usually treated as a security parameter, which means an encryption system with larger key size is more secure. At the same time, larger key size requires more computational time. In our experiments, we use three different key settings, TOY, MEDIUM and STD128. These settings are defined by OpenFHE, where TOY has no security, MEDIUM is 108-bits, and STD128 is 128-bits [2]. The integer size is 16 bits. The experimental results are shown in Fig. 4.

In Fig. 4, we can see that the operational time is longer than user acceptance. Moreover, it is anticipated that the key size will be regulated to 256 bits in the near future. Therefore, FHE performance improvement is definitely a critical issue when deploying privacy-preserving out-sourcing services.

5 Applications: Proxy Re-encryption

In this section, we will introduce how our implementation can be applied in the practical scenario.

Proxy re-encryption is a cryptographic technique which allows a third party, which is generally called as a proxy, to transform a ciphertext to a new ciphertext which is for another party. The idea was first proposed by Blaze et al. [5]. This is a useful crypto primitive for data forwarding or sharing. In general, a proxy re-encryption can be defined as follows [3]:

- **Setup**(1^λ) $\rightarrow$ PP : Given a security parameter λ , the algorithm outputs a public parameter set PP .
- **KeyGen**(i, PP) $\rightarrow$ (sk_i, pk_i): Given a user identity i and PP , the algorithm outputs the user's key pair (sk_i, pk_i).
- **ReKey**(PP, sk_i, pk_j) $\rightarrow$ $rk_{i \rightarrow j}$: Given PP, sk_i, pk_j , the algorithm outputs a rekey $rk_{i \rightarrow j}$.
- **Enc**(PP, pk_i, m) $\rightarrow$ c_i : Given a message m and PP, pk_i , the algorithm outputs a ciphertext c_i which is for user i .

- **Dec**(PP, sk_i, c_i) $\rightarrow m$: Given a ciphertext c for user i and PP, sk_i , the algorithm outputs the original message m .
- **ReEnc**($PP, c_i, rk_{i \rightarrow j}$) $\rightarrow c_j$: Given a ciphertext c_i , which is for user i , PP and $rk_{i \rightarrow j}$, the algorithm outputs a ciphertext c_j which is for user j .

In the transformation process, the proxy should learn nothing about the user private key. With FHE and multiple encryption, proxy re-encryption will become trivial. Here we assume that all users share the same FHE environment, which we use pk, sk as the public key and the private key, and their own public key encryption schemes PKE, which we use pk_i, sk_i as the public key and the private key for user i . The proxy re-encryption algorithms can be defined as follows:

- **Enc**(PP, pk, pk_i, m) $\rightarrow c_i$:

$$c_i = \text{FHE.}\mathbf{Enc}(pk, \text{PKE.}\mathbf{Enc}(pk_i, m))$$

- **Dec**(PP, sk, sk_i, c_i) $\rightarrow m$:

$$m = \text{FHE.}\mathbf{Dec}(sk, \text{PKE.}\mathbf{Dec}(sk_i, c_i))$$

- **ReKey**(PP, sk_i, pk_j) $\rightarrow rk_{i \rightarrow j}$:

$$rk_{i \rightarrow j} = \{\text{FHE.}\mathbf{Enc}(sk_i), \text{FHE.}\mathbf{Enc}(pk_j)\}$$

- **ReEnc**($PP, c_i, rk_{i \rightarrow j}$) $\rightarrow c_j$:

$$\begin{aligned} c'_i &= \text{FHE.}\mathbf{Eval}(\text{PKE.}\mathbf{Dec}, \text{FHE.}\mathbf{Enc}(sk_i), c_i), \\ c_j &= \text{FHE.}\mathbf{Eval}(\text{PKE.}\mathbf{Enc}, \text{FHE.}\mathbf{Enc}(pk_j), c'_i) \end{aligned}$$

In this idea, there will not be C_2^n rekeys, where n is the user number. Instead, only $O(n)$ keys are necessary. Our proposed work can support the modular inverse computation over the ciphertext space and therefore can support ElGamal-based re-encryption approach mentioned above.

6 Conclusion and Future Works

In this paper, we propose an implementation design for extended Euclidean algorithm with FHE technologies. We evaluate the computational costs of our work. Frankly speaking, the computational time is not acceptable for practical use but we successfully prove the feasibility of this idea. We also propose a possible application which can use our implementation.

Our future work will include the following directions:

1. **Performance Enhancement.** As described above, OpenFHE has a hardware acceleration framework. We will re-implement our design with the help of GPU computational power and see how the performance improves.
2. **Equality Check.** One of the key issue in FHE technologies is that the third party has no idea about the computational result. This makes the conditional control hard to be implemented. Encryption with equality test is a special kind of encryption techniques [20]. With trapdoors, it is feasible to do the equality check between two ciphertexts. Based on this concept, if we can port this feature to FHE schemes, it will greatly improve the practicality of FHE technologies.

Acknowledgements This work was supported by the National Science and Technology Council, Taiwan, under Grant MOST 110-2221-E-003-002-MY3 and NSTC 112-2634-F-004-001-MBK.

References

1. Google privacy policy. <https://policies.google.com/privacy?hl=en-US>. Accessed 28 Aug 2023
2. Program listing for file binfhe-constants.h—openfhe documentation. https://openfhe-development.readthedocs.io/en/latest/api/program_listing_file_binfhe_include_binfhe_constants.h.html. Accessed 28 Aug 2023
3. Ateniese G, Benson K, Hohenberger S (2009) Key-private proxy re-encryption. In: Fischlin M (ed) Topics in cryptology—CT-RSA 2009. Springer, Berlin Heidelberg, Berlin, Heidelberg, pp 279–294
4. Badawi AA, Bates J, Bergamaschi F, Cousins DB, Erabelli S, Genise N, Halevi S, Hunt H, Kim A, Lee Y, Liu Z, Micciancio D, Quah I, Polyakov Y, Saraswathy RV, Rohloff K, Saylor J, Suponitsky D, Triplett M, Vaikuntanathan V, Zucca V (2022) Openfhe: Open-source fully homomorphic encryption library. Cryptology ePrint Archive, Paper 2022/915. <https://eprint.iacr.org/2022/915>
5. Blaze M, Bleumer G, Strauss M (1998) Divertible protocols and atomic proxy cryptography. In: Nyberg K (ed) Advances in cryptology—EUROCRYPT'98. Springer, Berlin Heidelberg, Berlin, Heidelberg, pp 127–144
6. Brakerski Z, Gentry C, Vaikuntanathan V (2012) (Leveled) fully homomorphic encryption without bootstrapping. In: Proceedings of the 3rd innovations in theoretical computer science conference, ITCS '12. Association for Computing Machinery, New York, NY, USA, pp 309–325. <https://doi.org/10.1145/2090236.2090262>
7. Brakerski Z, Vaikuntanathan V (2011) Efficient fully homomorphic encryption from (standard) LWE. In: 2011 IEEE 52nd annual symposium on foundations of computer science, pp 97–106. <https://doi.org/10.1109/FOCS.2011.12>

8. Brakerski Z, Vaikuntanathan V (2011) Fully homomorphic encryption from ring-LWE and security for key dependent messages. In: Rogaway P (ed) *Advances in cryptology—CRYPTO 2011*. Springer, Berlin Heidelberg, Berlin, Heidelberg, pp 505–524
9. Cheon JH, Kim A, Kim M, Song Y (2017) Homomorphic encryption for arithmetic of approximate numbers. In: Takagi T, Peyrin T (eds) *Advances in cryptology—ASIACRYPT 2017*. Springer International Publishing, Cham, pp 409–437
10. Chillotti I, Gama N, Georgieva M, Izabachène M (2020) The: fast fully homomorphic encryption over the torus. *J Cryptol* 33(1):34–91. <https://doi.org/10.1007/s00145-019-09319-x>. Jan
11. van Dijk M, Gentry C, Halevi S, Vaikuntanathan V (2010) Fully homomorphic encryption over the integers. In: Gilbert H (ed) *Advances in cryptology—EUROCRYPT 2010*. Springer, Berlin, Heidelberg, pp 24–43
12. Ducas L, Micciancio D (2015) FHEw: bootstrapping homomorphic encryption in less than a second. In: Oswald E, Fischlin M (eds) *Advances in cryptology—EUROCRYPT 2015*. Springer, Berlin, Heidelberg, pp 617–640
13. Evtushkin D, Riley R, Abu-Ghazaleh NC, ECE, Ponomarev D (2018) Branchscope: a new side-channel attack on directional branch predictor. *SIGPLAN Not.* 53(2):693–707. <https://doi.org/10.1145/3296957.3173204>
14. Fan J, Vercauteren F (2012) Somewhat practical fully homomorphic encryption. *Cryptology ePrint Archive*, Paper 2012/144. <https://eprint.iacr.org/2012/144>
15. Gentry C (2009) Fully homomorphic encryption using ideal lattices. In: *Proceedings of the forty-first annual ACM symposium on theory of computing, STOC'09*. Association for Computing Machinery, New York, NY, USA, pp 169–178. <https://doi.org/10.1145/1536414.1536440>
16. Gentry C, Sahai A, Waters B (2013) Homomorphic encryption from learning with errors: conceptually-simpler, asymptotically-faster, attribute-based. In: Canetti R, Garay JA (eds) *Advances in cryptology—CRYPTO 2013*. Springer, Berlin, Heidelberg, pp 75–92
17. Lamé G (1844) Note sur la limite du nombre des divisions dans la recherche du plus grand commun diviseur entre deux nombres entiers
18. Marcolla C, Sucasas V, Manzano M, Bassoli R, Fitzek FHP, Aaraj N (2022) Survey on fully homomorphic encryption, theory, and applications. *Proc IEEE* 110(10):1572–1609. <https://doi.org/10.1109/JPROC.2022.3205665>
19. Rivest RL, Adleman L, Dertouzos ML (1978) On data banks and privacy homomorphisms. *Foundations of secure computation*. Academia Press, pp 169–179
20. Yang G, Tan CH, Huang Q, Wong DS (2010) Probabilistic public key encryption with equality test. In: Pieprzyk J (ed) *Topics in cryptology—CT-RSA 2010*. Springer, Berlin, Heidelberg, pp 119–131

Cryptanalysis to Ciphertext-Policy Attribute-Based Encryption of Liu et al. and Amirthavalli et al.



Yi-Fan Tseng , Wen-Yi Chen, You-Qian Chen, and Jheng-Jia Huang

Abstract Attribute-based encryption is a fine-grained access control mechanism designed for one-to-many scenarios, making it particularly well-suited for contemporary data sharing within cloud environments. To mitigate computational overhead, numerous attribute-based encryption schemes founded on elliptic curve cryptography have surfaced in academic literature. Unfortunately, these schemes are either insecure or without security proof. In this manuscript, we first review on the pairing-free ciphertext-policy attribute-based encryption schemes introduced by Liu et al. (Entropy 25(7), 2023) and Amirthavalli et al. (Comput Syst Scid Eng 45(3):3079–3095, 2023), and then give a cryptanalysis on their schemes. More precisely, we demonstrate an attack method to their schemes, which allows an adversary to recover the master secret key given a set of private keys sharing at least a same attribute.

Keywords Attribute-based encryption · Elliptic curve cryptography · Pairing-free · Cryptanalysis

1 Introduction

Attribute-based encryption (ABE) was initialized by Sahai and Waters [1] in 2005 to enable fine-grained access control. Unlike identity-based encryption [2] or one-to-one encryption [3, 4], ABE serves as a one-to-many encryption mechanism. Moreover, ABE permits message encryption to non-designated recipients, eliminating the need for the encryptor to know recipients' identities in advance. These attributes make ABE well-suited for data sharing in cloud environments [5–9] nowadays.

ABE comes in two varieties: ciphertext-policy ABE [10–12] and key-policy ABE [13–15]. In a ciphertext-policy attribute-based encryption (CPABE) scheme, an encryptor encrypts a message with an access structure. Each user in a CPABE scheme registers themselves with a third party, referred to as Key Generation Center

Y.-F. Tseng (✉) · W.-Y. Chen · Y.-Q. Chen · J.-J. Huang
Department of Computer Science, National Chengchi University, Taipei, Taiwan
e-mail: yftseng@cs.nccu.edu.tw

(KGC), along with a set of attributes. Decryption works only when a user's attribute set satisfies the access structure of a ciphertext. Key-policy attribute-based encryption (KPABE) is the dual version of CPABE. In KPABE, a user's private key corresponds to an access structure, while a ciphertext is related to a set of attributes. Although CPABE and KPABE find applications in different scenarios, CPABE is considered more suitable for contemporary data sharing models.

The constructions of most of ABE schemes primarily rely on bilinear maps [2], also known as pairings, a potent mathematical tool for building cryptographic primitives. However, implementing bilinear maps is currently time-consuming compared to modular exponentiation or elliptic curve operations. Such resource-intensive tools are unsuitable for lightweight devices like IoT devices. Consequently, numerous "pairing-free" cryptographic primitives have been proposed in the literature, including pairing-free ABE [14, 16–25]. It's worth noting that although several ABE variants avoid using pairings, such as lattice-based ABE [26–28], "pairing-free" ABE typically refers to schemes constructed over elliptic curve groups. Regrettably, all existing pairing-free ABE schemes are either vulnerable or lack security proofs. In 2017, Herranz [29] demonstrated that [16, 17] were insecure. In 2020, Herranz [30] conducted a cryptanalysis on [14, 18, 20–23]. Also, in 2020, Tseng and Huang [31] revealed a collusion attack on [18, 19]. In 2021 and 2022, Tseng et al. [32, 33] further show the security flaws for the schemes of [34, 35].

In this manuscript, we show the security flaws for the CPABE schemes proposed by Liu et al. [36] and Amirthavalli et al. [37], respectively. Recently, two pairing-free CP-ABE schemes are proposed by Liu et al. [36] and Amirthavalli et al. [37]. Two schemes share the similar construction, except that Amirthavalli et al. [37] adds a revocation method in their scheme. In this work, we will show that, an adversary with at least two private keys for the same attribute is able to recover the master secret key.

The rest of this manuscript is organized as follows. In Sect. 2, we will introduce the notations used in this manuscript, and give the syntax and security definition for ciphertext-policy attribute-based encryption. In Sect. 3, the review on Liu et al.'s and Amirthavalli et al.'s scheme are given. We then demonstrate our attack in Sect. 4. Finally, we conclude our manuscript in Sect. 5.

2 Preliminaries

In this section, we give the notation used in this manuscript, and the background of CP-ABE.

2.1 Notations

The notations used in this manuscript are listed as follows. For an algorithm A , we denote by " $y \leftarrow A$ " that y is the output obtained by running A . For a set S , by

“ $x \in_R S$ ” we mean uniformly randomly choose an element x from S . We use $[a, b]$ to denote the set of consecutive integers $\{a, a + 1, \dots, b\}$. A function $f : \mathbb{N} \rightarrow \mathbb{R}$ is said negligible in n , if for every $k \in \mathbb{N}$, there is $n_0 \in \mathbb{N}$ such that for every $n \geq n_0$, $|f(n)| < \frac{1}{n^k}$. By PPT we mean “probabilistic polynomial-time”.

2.2 Ciphertext-Policy Attribute-Based Encryption

A CP-ABE scheme consists of the following four algorithms **Setup**, **Enc**, **KG**, **Dec**. **Setup**(1^λ). Taking as input the security parameter, the algorithm outputs the system parameter **params** and the master secret key **msk**. Note that **params** will be a implicitly input for the following algorithms.

Enc($\mathbb{A}, M$). Taking as inputs an access structure $\mathbb{A}$ and a message M , the algorithm outputs a ciphertext **CT**.

KG(**msk**, S). Taking as inputs the master secret key **msk** and an attribute set S , the algorithm outputs a private key D . Similarly, we do not regard S as a part of the private key when we are evaluating the length of the private key.

Dec(**CT**, D). Taking as inputs a ciphertext **CT** and a private key D , the algorithm outputs a message.

Remark 1 In common case, the access structure $\mathbb{A}$ used for encryption will be published with the ciphertext **CT**. However, $\mathbb{A}$ is not viewed as a part of the ciphertext when we are evaluating the length of the ciphertext. We hence omit the symbol $\mathbb{A}$ in **CT** for simplicity. Besides, the description for the universe $\mathcal{U}$, i.e., the set of possible attributes in the system, will be included in **params**.

Correctness. For $CT \leftarrow \text{Enc}(\mathbb{A}, M)$, $D \leftarrow \text{KG}(\text{msk}, S)$, we have $M \leftarrow \text{Dec}(CT, D)$ if S satisfies $\mathbb{A}$, denoted by $S \models \mathbb{A}$.

Below we show the IND-CPA security for CPABE, i.e. indistinguishability under chosen-plaintext attacks, which can be defined as a game between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

Setup. The challenger $\mathcal{C}$ runs the **Setup** algorithm and sends **params** to the adversary $\mathcal{A}$.

Phase 1. The adversary $\mathcal{A}$ makes polynomially many queries to the **KG** oracles.

– **KG**(S_i): Upon inputting an attribute set S_i , the oracle returns the corresponding private key D_{S_i} .

Challenge. The adversary submits two distinct message (M_0, M_1) of the same length, and a target access structure $\mathbb{A}^*$, where $\mathbb{A}^*$ does not satisfy the access structures queried in **Phase 1**. Then the challenger $\mathcal{C}$ chooses a bit $b \xleftarrow{\$} \{0, 1\}$, and returns $CT^* \leftarrow \text{Enc}(\mathbb{A}^*, M_b)$.

Phase 2. It is the same as **Phase 1** except that $\text{KG}(\mathbf{S}_j)$ such that $\mathbf{S}_j \models \mathbb{A}^*$ are not allowed.

Guess. The adversary $\mathcal{A}$ outputs a bit b' , and wins the game if $b' = b$.

For a stateful adversary $\mathcal{A}$, we define the advantage function

$$\text{Adv}_{\mathcal{A}}^{\text{CPA}}(\lambda) = \left| \Pr \left[\begin{array}{l} b' = b \\ \text{(params, msk)} \leftarrow \text{Setup}(1^\lambda); \\ (\mathbb{A}^*, \mathbf{M}_0, \mathbf{M}_1) \leftarrow \mathcal{A}^{\text{KG}(\text{msk}, \cdot)}(\text{params}); \\ b \in_R \{0, 1\}; \\ \text{CT}^* \leftarrow \text{Enc}(\mathbb{A}^*, \mathbf{M}_b); \\ b' \leftarrow \mathcal{A}^{\text{KG}(\text{msk}, \cdot)}(\text{CT}^*); \end{array} \right] - \frac{1}{2} \right|$$

with restriction that all queries $\mathbf{S}$ that $\mathcal{A}$ makes to $\text{KG}(\text{msk}, \cdot)$ satisfies $\mathbf{S} \not\models \mathbb{A}^*$. We say that a CPABE scheme is IND-CPA secure if for all PPT algorithm $\mathcal{A}$, $\text{Adv}_{\mathcal{A}}^{\text{CPA}}(\lambda)$ is negligible in λ .

2.3 Access Structure

The ABE schemes of [36, 37] apply linear secret sharing scheme (LSSS) to represent access structures, where an access structure is encoded into a matrix $\mathbf{M}$ and a labelling function ρ . Let $\mathcal{U}$ be the universe, and $\mathbf{M}$ be an $\ell \times m$ matrix. $\rho: [1, \ell] \rightarrow \mathcal{U}$ is a mapping from the indices of rows in $\mathbf{M}$ to attributes in $\mathcal{U}$. An attribute set $\mathbf{S} \subset \mathcal{U}$ is said to satisfy an access structure $\mathbb{A} = (\mathbf{M}, \rho)$, if there exist an index set $I \subset [1, \ell]$ and a constant set $\{c_i\}_{i \in I}$, such that

1. $I = \{i \mid \rho(i) \in \mathbf{S}\};$
2. $\sum_{i \in I} c_i \mathbf{M}_i = \underbrace{(1, 0, \dots, 0)}_n;$

where $\mathbf{M}_i$ is the i th row of $\mathbf{M}$.

3 Review on Pairing-Free Attribute-Based Encryption Schemes

In this section, we review on the ABE schemes of [36–38]. We may only show the essential information for the cryptanalysis shown later. The detailed description of the schemes is referred to [36–38].

3.1 Liu et al.'s Pairing-Free ABE Scheme

We first introduce Liu et al.'s ABE scheme. In their scheme, the **Setup** algorithm is separated into two algorithms, one is called **Setup**, which is responsible of generating public parameter, the other is **TAS**etup, which is performed by a trusted third party called TA and responsible of generating the attribute-related parameter and the master secret key. For simplicity, we combine these two algorithms into a single **Setup** algorithm in the following description.

Setup(1^λ). The algorithm performs as follows. Let the universe $\mathcal{U} = \{a_1, \dots, a_{|\mathcal{U}|}\}$.

1. Choose an elliptic curve group with prime order p and generator P .
2. Choose a secure hash function $H : \{0, 1\}^* \rightarrow \mathbb{Z}_p^*$.
3. Choose a secure symmetric encryption algorithm $E_K(\cdot)/D_K(\cdot)$, where K represents the symmetric key.
4. Choose a secure message authentication code algorithm $\text{MAC}_K(\cdot)$, where K is the key.
5. Choose a random number n from $\mathbb{Z}_p^*$ as msk, and compute $PK = nP$.
6. For $i \in [1, |\mathcal{U}|]$, choose $k_i \xleftarrow{\$} \mathbb{Z}_p^*$ and compute $PK_i = k_i P$.
7. Output $\text{params} = (p, P, H, \{PK_i\}_{i \in [1, |\mathcal{U}|]})$ and $\text{msk} = (n, \{k_i\}_{i \in [1, |\mathcal{U}|]})$.

KG(msk, $\mathbf{S}$, UID). Let UID be the identity of the user with the attribute set $\mathbf{S}$. The algorithm computes $D_{i, \text{UID}} = k_i + H(\text{UID})n$ for $i \in \mathbf{S}$, and outputs $\mathbf{D}_{\text{UID}} = \{D_{i, \text{UID}}\}_{i \in \mathbf{S}}$.

Enc($\mathbb{A}$, $\mathbf{M}$). The **Enc** algorithm of [36] consists of two parts, static attribute encryption and dynamic attribute encryption. Here we only review dynamic attribute encryption. Let $\mathbb{A} = (\mathbf{M}, \rho)$, where $\mathbf{M} \in \mathbb{Z}_p^{\ell \times m}$ and $\rho : [1, \ell] \rightarrow [1, |\mathcal{U}|]$.

1. Randomly choose $d \xleftarrow{\$} \mathbb{Z}_p^*$.
2. Compute $dP = (d_x, d_y)$.
3. Use d_x as the encryption key to encrypt $\mathbf{M}$ into $C_{\mathbf{M}}$ with a secure symmetric encryption scheme, i.e., $C_{\mathbf{M}} = E_{d_x}(\mathbf{M})$.
4. Use d_y as the key to compute the message authentication code $\text{MAC}_{\mathbf{M}}$, i.e., $\text{MAC}_{\mathbf{M}} = \text{MAC}_{d_y}(\mathbf{M})$.
5. Choose two random vectors $\mathbf{v}, \mathbf{w}$ from $\mathbb{Z}_p^m$ with d and 0 as the first entries of $\mathbf{v}$ and $\mathbf{w}$, respectively.
6. Compute $\lambda_i = \langle \mathbf{M}_i, \mathbf{v} \rangle, \omega_i = \langle \mathbf{M}_i, \mathbf{w} \rangle$ for $i \in [1, \ell]$, where $\langle \cdot, \cdot \rangle$ is the inner product operation.
7. For $i \in [1, \ell]$, compute

$$\begin{aligned} C_{1,i} &= \lambda_i P + \omega_i PK_{\rho(i)}, \\ C_{2,i} &= \omega_i P. \end{aligned}$$

8. Set the ciphertext $\text{CT}_d = (\mathbb{A} = (\mathbf{M}, \rho), C_{\mathbf{M}}, \text{MAC}_{\mathbf{M}}, \{C_{i,1}, C_{i,2}\}_{i \in [1, \ell]})$.

3.2 Amirthavalli et al.'s Pairing-Free ABE Scheme

The algorithms **Setup**, **KG**, **Enc** of [37] are almost the same as those in [36], except that Amirthavalli et al.'s work additionally considers the revocation issue in their scheme. We hence omit the description of Amirthavalli et al.'s scheme in order to avoid unnecessary repetition.

4 Cryptanalysis to [36, 37]

Due to the similarity of the schemes of [36, 37], we only show the attack to Liu et al.'s scheme here. In this section, we give an attack to Liu et al.'s ABE scheme. Our attack allows an adversary to recover the master secret key **msk** when it colludes with enough number of user. Note that, our attack can be successfully applied to Amirthavalli et al.'s scheme as well.

The algorithm for our first attack is shown in Algorithm 1. The crucial point of our first attack is that, the private key form of Liu et al.'s ABE scheme does not have enough entropy. A private key component $D_{x, \text{UID}}$ is of the form $k_x + H(\text{UID})n$. Observe that, no randomness is used to generate a private key, and each private key component contains the factor of n . Thus, given two private key components D_{x, UID_α} , D_{x, UID_β} for the same attribute a_x , one can easily recover k_x and n by solving the following equation system

$$\begin{cases} D_{x, \text{UID}_\alpha} = k_x + H(\text{UID}_\alpha)n \\ D_{x, \text{UID}_\beta} = k_x + H(\text{UID}_\beta)n \end{cases}.$$

Remark 2 Step 2 in Algorithm 1 can be done even if S_i s are not given. One can simply check S_i for D_{UID_i} by encrypting a randomly chosen message and performing a decrypt-then-check method.

Remark 3 Though our first attack only outputs a single k_j , once n is recovered, we can repeat Step 8 for each private key component $\{D_{i, \text{UID}_\alpha}\}_{i \in S_\alpha}$, $\{D_{i, \text{UID}_\beta}\}_{i \in S_\beta}$ to recover more master secret key components.

Remark 4 Obtaining private keys for other users is a reasonable assumption when analyzing the security for an ABE scheme. In real life, a user's private key may be leaked due to the malicious behavior of hackers or the user's own negligence. Users may also collude with attackers for their own benefit. In the security model of CPABE shown in Sect. 2.2, an adversary is allowed to query private keys of its choice. Thus, our attack falls in the security definition of CPABE.

Algorithm 1: Recovering msk Given Private Keys

Data: $\text{params} = (p, P, H, \{PK_i\}_{i \in [1, |\mathcal{U}|]});$ a set of private key $\{D_{\text{UID}_i}\}_{i \in [1, \delta]}$
for some integer δ

Result: (n, k_j) for some attribute $a_j \in \mathcal{U}$, or $\perp$

- 1 Let S_i be the attribute set related to D_{UID_i} ;
- 2 **if** $S_\alpha \cap S_\beta = \emptyset$ for all $\alpha, \beta \in [1, \delta], \alpha \neq \beta$ **then**
- 3 | **return** $\perp$;
- 4 **end**
- 5 Let $S_\alpha \cap S_\beta = \{a_j\}$;
- 6 Parse $D_{\text{UID}_\alpha} = \{D_{i, \text{UID}_\alpha}\}_{i \in S_\alpha}, D_{\text{UID}_\beta} = \{D_{i, \text{UID}_\beta}\}_{i \in S_\beta}$;
- 7 $n \leftarrow \frac{D_{j, \text{UID}_\alpha} - D_{j, \text{UID}_\beta}}{H(\text{UID}_\alpha) - H(\text{UID}_\beta)} \pmod{p}$;
- 8 $k_j \leftarrow D_{j, \text{UID}_\alpha} - H(\text{UID}_\alpha)n \pmod{p}$;
- 9 **return** (n, k_j) ;

5 Conclusion

In this manuscript, we show a security flaw of Liu et al.'s [36] and Amirthavalli et al.'s scheme [37], and give an attack method to their scheme. Our attack method allows an adversary to recover master secret key given at least two private keys with a same attribute. It is still an open problem to design an ABE scheme over elliptic curve groups. Although there is no conclusive evidence that this problem is unsolvable, we have a negative attitude towards solving this problem. Note that ID-based encryption (IBE) is a special case of ABE [29, 39]. As pointing out by Boneh et al. in their work [40], securing ID-based encryption based on discrete-log or RSA-like assumptions is a hard problem. Combining the implication from ABE to IBE with Boneh et al.'s impossibility result, we believe that designing a pairing-free ABE is difficult as well.

Acknowledgements This work was partially supported by the National Science and Technology Council of Taiwan, under grants NSTC 111-2221-E-004-005-, NSTC 111-2218-E-004-001-MBK, NSTC 112-2634-F-004-001-MBK, NSTC 112-2221-E-011-094-MY2, NSTC 112-2221-E-011-092-MY2, NSTC 112-2218-E-011-008-, NSTC 112-2634-F-011-002-MBK, NSTC 111-2218-E-011-016-, NSTC 113-2634-F-004-001-MBK, NSTC 113-2221-E-004-012-.

References

1. Sahai A, Waters B (2005) Fuzzy identity-based encryption. In: Cramer R (ed) Advances in cryptology—EUROCRYPT 2005. Springer, Berlin, Heidelberg, pp 457–473
2. Boneh D, Franklin M (2001) Identity-based encryption from the weil pairing. In: Kilian J (ed) Advances in cryptology—CRYPTO 2001. Springer, Berlin Heidelberg, pp 213–229
3. ElGamal T (1985) A public key cryptosystem and a signature scheme based on discrete logarithms. In: Blakley GR, Chaum D (eds) Advances in cryptology. Springer, Berlin, Heidelberg, pp 10–18

4. Rivest RL, Shamir A, Adleman L (1978) A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM* 21(2):120–126 <https://doi.org/10.1145/359340.359342>
5. Zhang Y, Deng RH, Xu S, Sun J, Li Q, Zheng D (2020) Attribute-based encryption for cloud computing access control: a survey. *ACM Comput Surv* 53(4). <https://doi.org/10.1145/3398036>
6. Liu S, Yu J, Xiao Y, Wan Z, Wang S, Yan B (2020) BC-SABE: blockchain-aided searchable attribute-based encryption for cloud-IOT. *IEEE Internet Things J* 7(9):7851–7867. <https://doi.org/10.1109/JIOT.2020.2993231>
7. Challagidad PS, Birje MN (2020) Efficient multi-authority access control using attribute-based encryption in cloud storage. *Procedia Comput Sci* 167:840–849. <https://doi.org/10.1016/j.procs.2020.03.423>, <https://www.sciencedirect.com/science/article/pii/S1877050920308899>, International conference on computational intelligence and data science
8. Zhang Z, Zeng P, Pan B, Choo KKR (2020) Large-universe attribute-based encryption with public traceability for cloud storage. *IEEE Internet Things J* 7(10):10314–10323. <https://doi.org/10.1109/JIOT.2020.2986303>
9. Ge C, Susilo W, Baek J, Liu Z, Xia J, Fang L (2021) Revocable attribute-based encryption with data integrity in clouds. *IEEE Trans Depend Sec Comput* pp 1–1. <https://doi.org/10.1109/TDSC.2021.3065999>
10. Premkamal PK, Pasupuleti SK, Alphonse PJA (2021) Dynamic traceable CP-ABE with revocation for outsourced big data in cloud storage. *Int J Commun Syst* 34
11. Xu S, Yuan J, Xu G, Li Y, Liu X, Zhang Y, Ying Z (2020) Efficient ciphertext-policy attribute-based encryption with blackbox traceability. *Inform Sci* 538. <https://doi.org/10.1016/j.ins.2020.05.115>
12. Yu Y, Guo L, Liu S, Zheng J, Wang H (2020) Privacy protection scheme based on CP-ABE in crowdsourcing-IoT for smart ocean. *IEEE Internet Things J* 7(10):10061–10071. <https://doi.org/10.1109/JIOT.2020.2989476>
13. Goyal V, Pandey O, Sahai A, Waters B (2006) Attribute-based encryption for fine-grained access control of encrypted data. In: *Proceedings of the ACM conference on computer and communications security*, pp. 89–98. <https://doi.org/10.1145/1180405.1180418>
14. Sowjanya K, Dasgupta M, Ray S, Obaidat MS (2020) An efficient elliptic curve cryptography-based without pairing KPABE for internet of things. *IEEE Syst J* 14(2):2154–2163. <https://doi.org/10.1109/JSYST.2019.2944240>
15. Zhang S, Li W, Wen Q, Zhang H, Jin Z (2020) A flexible KP-ABE suit for mobile user realizing decryption outsourcing and attribute revocation. *Wirel Pers Commun* 114(4):2783–2800
16. Odelu V, Das AK (2016) Design of a new CP-ABE with constant-size secret keys for lightweight devices using elliptic curve cryptography. *Secur Commun Netw* 9(17):4048–4059. <https://doi.org/10.1002/sec.1587>, <https://onlinelibrary.wiley.com/doi/abs/10.1002/sec.1587>
17. Odelu V, Das AK, Khurram Khan M, Choo KR, Jo M (2017) Expressive CP-ABE scheme for mobile devices in IoT satisfying constant-size keys and ciphertexts. *IEEE Access* 5:3273–3283
18. Ding S, Li C, Li H (2018) A novel efficient pairing-free CP-ABE based on elliptic curve cryptography for IoT. *IEEE Access* 6:27336–27345. <https://doi.org/10.1109/ACCESS.2018.2836350>
19. Wang Y, Chen B, Li L, Ma Q, Li H, He D (2020) Efficient and secure ciphertext-policy attribute-based encryption without pairing for cloud-assisted smart grid. *IEEE Access* 8:40704–40713. <https://doi.org/10.1109/ACCESS.2020.2976746>
20. Tan SY, Yeow KW, Hwang SO (2019) Enhancement of a lightweight attribute-based encryption scheme for the Internet of Things. *IEEE Internet Things J* 6(4):6384–6395. <https://doi.org/10.1109/JIOT.2019.2900631>
21. Yao X, Chen Z, Tian Y (2015) A lightweight attribute-based encryption scheme for the internet of things. *Future Gen Comput Syst* 49:104–112 <https://doi.org/10.1016/j.future.2014.10.010>, <https://www.sciencedirect.com/science/article/pii/S0167739X14002039>
22. Khandla D, Shahy H, Bz MK, Pais AR, Raj N (2019) Expressive CP-ABE scheme satisfying constant-size keys and ciphertexts. *Cryptology ePrint Archive, Report 2019/1257*. <https://ia.cr/2019/1257>

23. Karati A, Amin R, Biswas GP (2016) Provably secure threshold-based ABE scheme without bilinear map. *Arab J Sci Eng* 41:3201–3213
24. Sowjanya K, Dasgupta M, Ray S (2021) A lightweight key management scheme for key-escrow-free ECC-based CP-ABE for IoT healthcare systems. *J Syst Archit* 117:102108. <https://doi.org/10.1016/j.sysarc.2021.102108>, <https://www.sciencedirect.com/science/article/pii/S1383762121000849>
25. Cheng R, Wu K, Su Y, Li W, Cui W, Tong J (2021) An efficient ECC-based CP-ABE scheme for power IoT. *Processes* 9(7). <https://doi.org/10.3390/pr9071176>, <https://www.mdpi.com/2227-9717/9/7/1176>
26. Boyen X (2013) Attribute-based functional encryption on lattices. In: Sahai A (ed) *Theory of cryptography*. Springer, Berlin, Heidelberg, pp 122–142
27. Wang G, Liu Z, Gu D (2020) Ciphertext policy attribute-based encryption for circuits from LWE assumption. In: Zhou J, Luo X, Shen Q, Xu Z (eds) *Information and communications security*. Springer International Publishing, Cham, pp 378–396
28. Zhang J, Zhang Z, Ge A (2012) Ciphertext policy attribute-based encryption from lattices. In: *Proceedings of the 7th ACM symposium on information, computer and communications security, ASIACCS' 12*, Association for Computing Machinery, New York, NY, USA, pp 16–17
29. Herranz J (2017) Attribute-based encryption implies identity-based encryption. *IET Inform Sec* 11(6):332–337
30. Herranz J (2020) Attacking pairing-free attribute-based encryption schemes. *IEEE Access* 8:222226–222232. <https://doi.org/10.1109/ACCESS.2020.3044143>
31. Tseng YF, Huang JJ (2020) Cryptanalysis on two pairing-free ciphertext-policy attribute-based encryption schemes. In: *2020 international computer symposium (ICS)*, pp 403–407. <https://doi.org/10.1109/ICS51289.2020.00086>
32. Tseng, Y.F.: Cryptanalysis to Sowjanya et al.'s ABEs from ECC. In: Tsihrintzis GA, Wang SJ, Lin IC (eds) *2021 international conference on security and information technologies with AI, Internet computing and big-data applications*. Springer International Publishing, Cham, pp 287–294
33. Tseng YF, Huang JJ, Yang HY, Chien TY, Wu CH (2022) Cryptanalysis and discussion on two attribute-based encryption schemes. In: *2022 17th Asia joint conference on information security (AsiaJCIS)*, pp 24–28. <https://doi.org/10.1109/AsiaJCIS57030.2022.00014>
34. Sowjanya K, Dasgupta M, Ray S, Obaidat MS (2020) An efficient elliptic curve cryptography-based without pairing KPABE for internet of things. *IEEE Syst J* 14(2):2154–2163
35. Sethia D, Sahu R, Yadav S, Kumar R (2021) Attribute revocation in ECC-based CP-ABE scheme for lightweight resource-constrained devices. In: *2021 international conference on communication, control and information sciences (ICCISc)*, vol 1, pp 1–6. <https://doi.org/10.1109/ICCISc52257.2021.9485016>
36. Liu Y, Xu S, Yue Z (2023) A lightweight CP-ABE scheme with direct attribute revocation for vehicular ad hoc network. *Entropy* 25(7)
37. Amirthavalli M, Chithra S, Yugha R (2023) An improved pairing-free ciphertext policy framework for IoT. *Comput Syst Scid Eng* 45(3):3079–3095
38. Das S, Namasudra S (2023) MACPABE: multi-authority-based CP-ABE with efficient attribute revocation for IoT-enabled healthcare infrastructure. *Int J Netw Manag* 33(3):e2200
39. Tseng YF, Fan CI, Lin CW (2019) Provably secure ciphertext-policy attribute-based encryption from identity-based encryption. *J Univ Comput Sci* 25(3):182–202
40. Boneh D, Papakonstantinou P, Rackoff C, Vahlis Y, Waters B (2008) On the impossibility of basing identity based encryption on trapdoor permutations. In: *2008 49th annual IEEE symposium on foundations of computer science*, pp 283–292

Implementation and Analysis of Using Autoencoder to Suppress Noise Interference on Side-Channel Attacks



Kuo-Yu Tsai, Chung-Wei Kuo, Wei-Ming Weng, Guan-Ling Wang,
and Chun-Chang Lin

Abstract In recent years, Side-Channel Attack (SCA) has been widely used in hardware security analysis. Such an attack uses the power characteristics generated by the switching of internal logic gates during chip operation to extract the waveform characteristics of the encryption key. Analyzing the complete encryption key statistically, it poses a great threat to the security of the chip. However, these attacks can be prevented by adding noise to the signal output pins of the chip. These noises can be divided into naturally generated Gaussian noise and artificial noise. Artificial noise can effectively defend against SCA especially. In order to improve the attack success rate, we aim to discuss the use of Autoencoder to reduce the interference of Random delay noise on SCA. In the experiment, we used Denoising Autoencoder (DAE) to preprocess the power traces generated during chip operation. Then we applied Correlation Power Analysis (CPA) to resolve the encryption key. In addition, we implemented the DAE processing mechanism in the ChipWhisperer-Nano experimental board and used the actual measured original power traces and noised power consumption traces for training. After completing the training model, it can effectively preprocess the noised data and key analysis. According to the experimental results, we have successfully improved the efficiency and success rate of analyzing keys.

Keywords Data hiding · Steganography · Histogram shifting · Interpolation · Prediction errors

K.-Y. Tsai · C.-W. Kuo (✉) · W.-M. Weng · G.-L. Wang · C.-C. Lin
Department of Information Engineering and Computer Science, Feng Chia University, No. 100
Wenhua Rd., Xitun, Taichung 40724, Taiwan
e-mail: cwkuo@mail.fcu.edu.tw

K.-Y. Tsai
e-mail: kytsai@o365.fcu.edu.tw

C.-C. Lin
e-mail: M1205003@o365.fcu.edu.tw

1 Introduction

With the rapid development of the Internet of Things (IoT) application, many applications have become intelligent through IoT sensing components. For example, combining sensing components with the Internet of Vehicles system [1] enables the sharing of traffic information between vehicles and enables vehicles to plan their journeys and avoid congested roads automatically. However, an attacker can mount a Side-Channel Attack (SCA) which steals sensitive information, such as personal privacy and encryption keys, by calculating the power consumption or leaked electromagnetic radiation generated by IoT devices. SCA is a method of extracting secure information from a chip by measuring and analyzing physical parameters rather than attacking the encryption algorithm itself. During the execution of the encryption algorithm, the hardware device will generate various physical signals, such as electromagnetic radiation [2], power consumption [3], execution time, etc. These signals will be affected by the key information and form a correlation. An attacker can obtain essential characteristics of the signal by analyzing the correlation in these signals. According to different attack methods, SCA can be divided into two major categories: direct attacks and analytical attacks. Direct attacks analyze intercepted signals, such as Single Power Analysis (SPA) and Differential Power Analysis (DPA). Analysis attacks use known data sets to train an attack model and then analyze the actual measured signals. This includes template attacks (TA) [4] and machine learning-based attack methods [5–8]. Common countermeasures to protect chips from SCA are masking and hiding. The masking countermeasure is to split sensitive information into different parts to reduce dependency on the key [9, 10]. For example, block encryption technology splits the key into other blocks and encrypts each. Another hidden countermeasure is to add noise to the output signal so that the attacker cannot extract practical features to protect the safe transmission of the signal. For example, adding noise directly to the signal [11]. Therefore, although using SCA can extract features, the correlation of these features will decrease, and the attack will no longer be effective. To solve this problem, commonly used methods include low-pass filters [12], performing trajectory alignment [13], and various feature engineering [14, 15] methods to eliminate the interference of protection countermeasures. However, in recent years, researchers have explored deep learning technology for effective feature extraction [16] and restored the signal without adding noise by learning the correlation in the signal trace. Wu et al. and Gandara et al. propose using the autoencoder (AE) mechanism to eliminate noise interference [17, 18]. Although the protected noise is added, the noise can still be removed to extract the key according to the captured signal. Based on the research results, our study proposes to use AE to attack the real signal traces of physical circuits to verify whether they can cause consistent effects.

2 Related Works

Relevant results are adopted in this research, such as the implementation of the Advanced Encryption Standard (AES) algorithm, Correlation Power Analysis (CPA), and Denoising Autoencoder (DAE) mechanism for eliminating noise. Detailed descriptions will be given in this section.

2.1 Advanced Encryption Standard

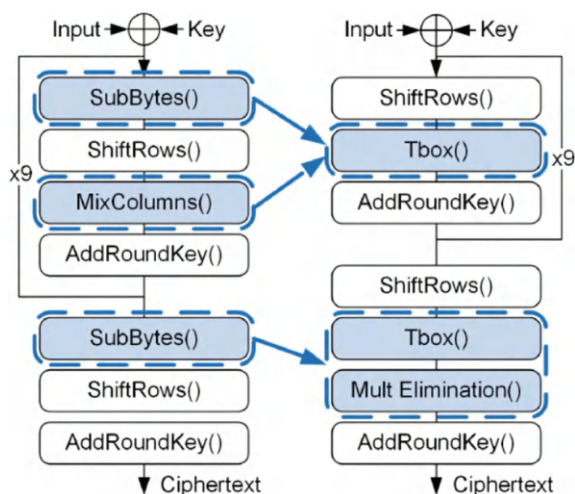
AES is a block cipher derived from the Rijndael cipher. It uses a fixed 128-bit plaintext block length and offers key lengths of 128-bit, 192-bit, or 256-bit. This study focuses on potential attacks on the initial key in AES-128 encryption. To do this, plaintext binary data is segmented into blocks of 16 bytes, matching the key length, called plaintext blocks. These blocks undergo 10 encryption rounds, transforming each plaintext block into a ciphertext block. The AES encryption process follows four steps as shown in Fig. 1. [19–22].

SubBytes: Each byte in the plaintext block gets substituted with a corresponding byte from S-Box, a pre-calculated table.

ShiftRows: The rows of the plaintext block are cyclically shifted by different offsets. This further increases the diffusion of data across the block, making it harder to analyze individual bytes.

MixColumns: A linear transformation is applied to every column within the block, intermixing the bytes and masking any lingering patterns. Notably, this step is skipped in the final encryption round.

Fig. 1 AES flowchart [23]



AddRoundKey: The present round key, derived from the initial key, is combined with the plaintext block using XOR operation.

The initial encryption round of AES-128 begins with the AddRoundKey step. This step involves an XOR operation between the plaintext block and the initial key, creating a unique power consumption pattern. The focus of this study is to analyze the correlation between the expected power consumption of different sub-keys and the actual power consumption. The strongest correlation was found with the 16-byte sub-key that was used in AES-128 encryption, out of all the outcomes that were examined.

2.2 Correlation Power Analysis

CPA is one of Side-Channel Attack (SCA) for analyzing secret key proposed by Brier et al. [23]. CPA is a chosen plaintext attack and is used to retrieve the security information in the encrypt system. The correlation coefficient can be calculated according to the power consumption as encrypting the same plaintext with different keys and the actual measured power consumption.

In the first step of AES, there is a linear relationship between the plaintext and the key. That is, the 16-byte key has not been mixed. In this research, we build a power consumption model in accordance with Hamming weight. The XOR operation is used to calculate two sets of binary signals bit by bit, such as $S1 = 0011\ 0000$ and $S2 = 0010\ 0011$. After the XOR operation, the result is $0001\ 0011$, and the Hamming weight is 3. Detail descriptions for the attack are shown in Fig. 2. We can calculate the correlation coefficient of each sampling point according to Eq. (1).

$$r = \frac{\sum x_i y_i - \frac{\sum \bar{x} \sum \bar{y}}{n}}{\sqrt{\sum x_i^2 - \sum x_i^2 / n} \sqrt{\sum y_i^2 - \sum y_i^2 / n}} \quad (1)$$

2.3 Denoising Autoencoder

DAE is initially applied to remove noise from images [18]. Wu and Picek [17] use Convolutional Autoencoder (CAE) to implement DAE and apply it to the noise removal for power consumption. Different from general Autoencoder (AE), the original hidden layer is replaced by a convolutional layer in Wu and Picek's CAE for a better correlation of the signals. In Wu and Picek's proposed work [17], a training set of 10,000 examples are used to train the model. Each data contains an original signal and a signal after adding noise. The learning goal of DAE is to minimize the difference between an original signal and a noise signal. After training the model, a testing

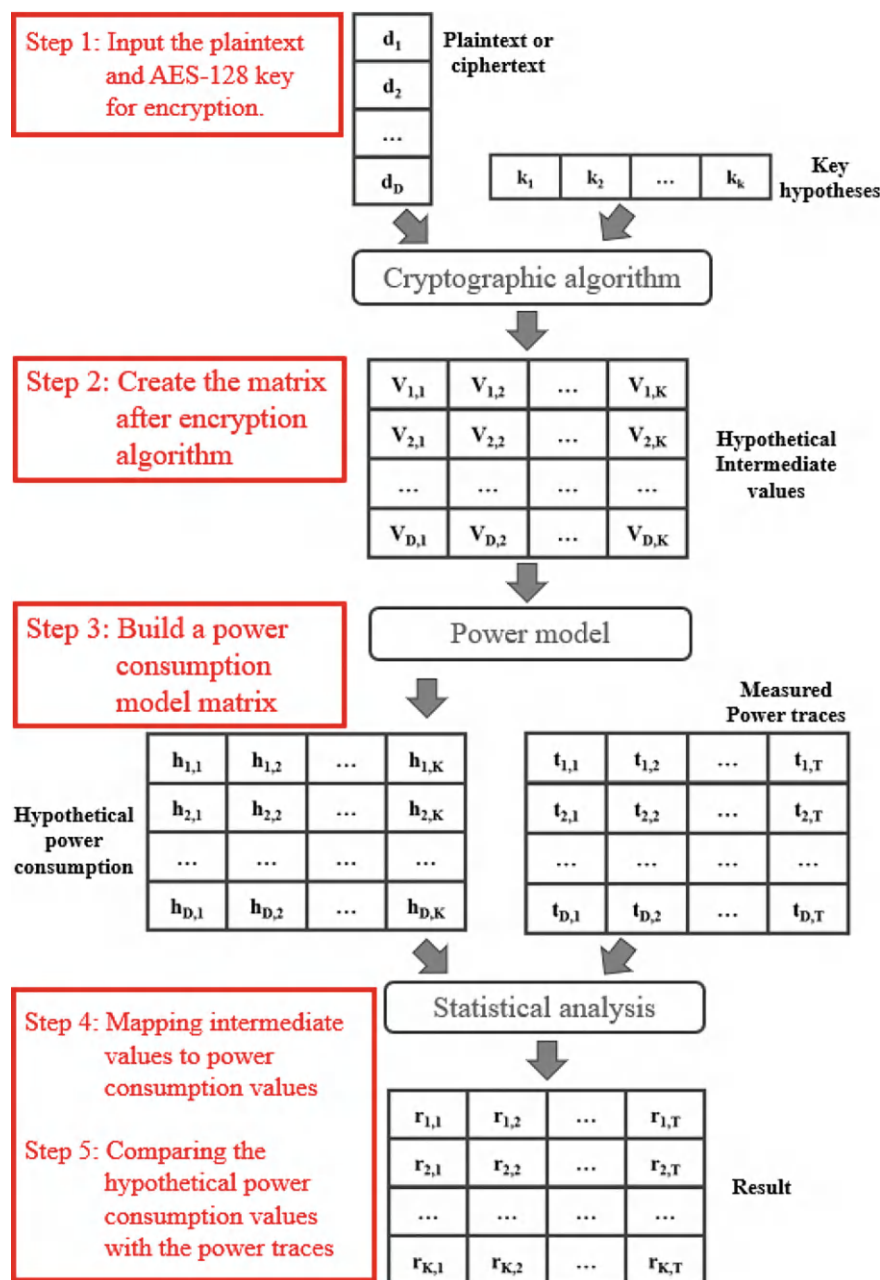


Fig. 2 Block diagram illustrating the steps 1 to 5 of a CPA attack

set of 5,000 examples with noise is applied to the model, and the corresponding 5,000 denoise power consumption signals can be obtained.

3 Our Proposed Method

We introduce how to implement a Side-Channel Attack (SCA) experiment and demonstrate the results of the attack after adding noise.

3.1 Detailed Descriptions for Experiment

In the experiment, we use ChipWhisperer-Nano experimental board produced by NewAE Technology Inc. The signal sampling rate is 20MS/s, and the sampling buffer size is 50,000 points. We can obtain the signal traces generated by the hardware for the 8-bit analog/digital converter (ADC) in the experimental board as shown in Fig. 3. We set the signal section to be observed for feature analysis. Figure 4 shows our experimental results of AES-128 encryption algorithm. The numbers 0~15 stand for 16 sub-keys, each sub-key can be retrieved according to the statistical results of the key guessing entropy (private key guessing entropy, PGE).

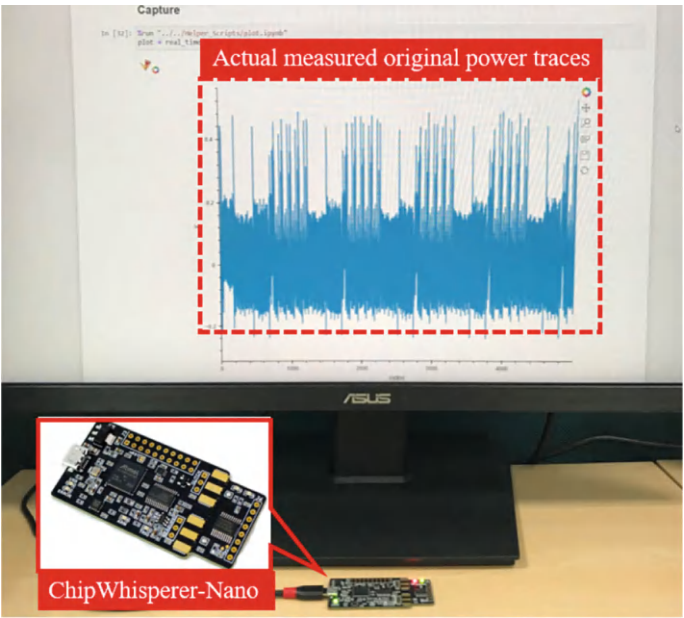


Fig. 3 Power consumption trace of signal

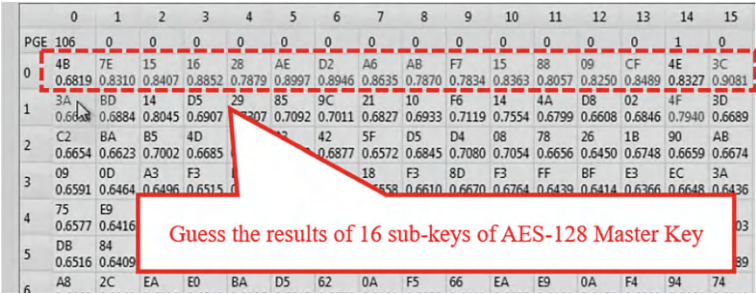


Fig. 4 Statistical result of AES-128 subkeys correlation

3.2 Original Signal Trace Analysis

Figure 5 shows the signal trace of ChipWhisperer-Nano experimental board that performs AES-128 encryption operation. The actual waveform of the hardware is captured through the built-in ADC of the experimental board. Based on this method, we start to input different plaintext data and repeatedly perform AES-128 so that different characteristic signals can be generated for an attack model.

After building the model, Correlation Power Analysis (CPA) is then used to analyze each signal trace measured from the hardware device. As shown in Fig. 4, the numbers of PGE on the left side represent the ranking of the correlation coefficients for sub-key guesses. Note that the number close to 0 means a higher correlation to the actual key. Without any defense mechanism, the attack in our experiment only needs less than 50 signal traces to retrieve all 16 sub-keys. Then, Autoencoder (AE) is used to preprocess the original signal traces. As shown in the results of [18], 831 signal traces are used to crack the sub-keys based on the Convolutional Neural Network (CNN) model, and the number of signal traces is reduced to 751 based on the AE model. Compared with CPA, these two methods do not get the best results.

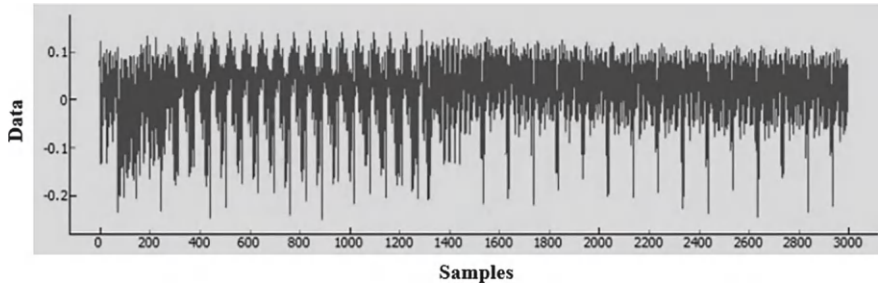
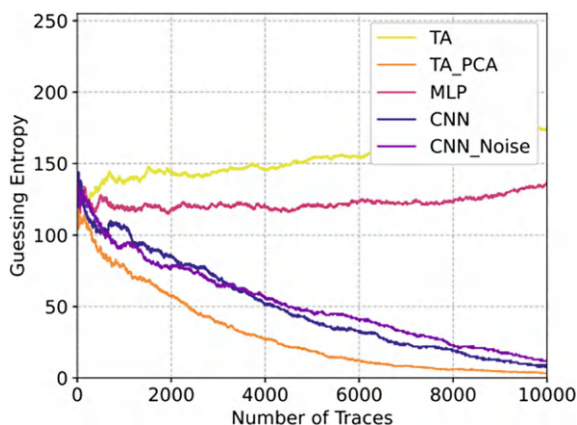


Fig. 5 Original signal trace

Fig. 6 Guessing entropy: denoise random delay interrupts with frequency analysis [18]



3.3 Signal Trace Analysis with Noise Added

Figure 6 shows the results for different attack methods as the original signal trace is mixed with Gaussian noise. We can see that Template Attack (TA) attacks and Multilayer Perceptron (MLP) attacks cannot work for the signal traces mixed with noise. Based on Principal Component Analysis (PCA), 10,000 signal traces are used to crack the sub-keys in the TA attack, and CNN can crack only 1,754 entries under the average trace method. The average of 10 signal traces is also taken based on AE. That is, AE can effectively improve the attack efficiency after eliminating Gaussian noise.

4 Conclusions

We implement the Autoencoder (AE) noise removal mechanism on the ChipWhisperer-Nano circuit board and increase the success rate of the attack. Generally, noise is added at the power supply end in encryption devices to prevent signal feature extraction to resist Side-Channel Attacks (SCAs). The Convolutional Autoencoder (CAE) used in our experiment successfully filters out the added noise and obtains a clean signal trace. Then, the Correlation Power Analysis (CPA) is used to obtain the correlation coefficient of each signal trace, and all sub-keys are retrieved according to correlation coefficients.

Acknowledgements This work is partially supported by the National Science and Technology Council (NSTC 112-2222-E-035-002-, NSTC 112-2221-E-035-049-, and NSTC 112-2634-F-005-001-MBK).

References

1. Chougule AR, Suganthi K (2018) IoT based smart car monitoring system. In: 2018 tenth international conference on advanced computing (ICoAC). Chennai, India, pp 281–285
2. Gandolfi K, Mourtlet C, Olivier F (2001) Electromagnetic analysis: concrete results. In: Cryptographic hardware and embedded systems—CHES 2001: third international workshop, Paris, France, May 14–16, 2001 proceedings 3. pp 251–261
3. Kocher P, Jaffe J, Jun B, Rohatgi P (2001) Introduction to differential power analysis. *J Cryptogr Eng* 1:5–27
4. Hanley N, Tunstall M, Marnane WP (2009) Unknown plaintext template attacks. In: Information security applications, 10th international workshop, WISA 2009, Busan, Korea, revised selected papers 10. pp 148–162
5. Hou S, Zhou Y, Liu H (2019) Convolutional neural networks for profiled side-channel analysis. *Radioengineering* 28(3):651–658
6. Cagli E, Dumas C, Prouff E (2017) Convolutional neural networks with data augmentation against jitter-based countermeasures: profiling attacks without pre-processing. In: Cryptographic hardware and embedded systems—CHES 2017: 19th international conference, Taipei, Taiwan, proceedings. pp 45–68
7. Picek S, Samiotis IP, Kim J, Heuser A, Bhasin S, Legay A (2018) On the performance of convolutional neural networks for side-channel analysis. In: Security, privacy, and applied cryptography engineering: 8th international conference, SPACE 2018, Kanpur, India, proceedings 8. pp 157–176
8. Maghrebi H, Portigliatti T, Prouff E (2016) Breaking cryptographic implementations using deep learning techniques. Security, privacy, and applied cryptography engineering: 6th international conference, SPACE 2016, Hyderabad, India, proceedings 6. pp 3–26
9. Barthe G, Dupressoir F, Faust S, Grégoire B, Standaert FX, Strub PY (2017) Parallel implementations of masking schemes and the bounded moment leakage model. In: Advances in cryptology—EUROCRYPT 2017: 36th annual international conference on the theory and applications of cryptographic techniques, Paris, France, proceedings, part I 36. pp 535–566
10. Chari S, Jutla CS, Rao JR, Rohatgi P (1999) Towards sound approaches to counteract power-analysis attacks. In: Advances in cryptology—CRYPTO’99: 19th annual international cryptology conference Santa Barbara, California, USA, Aug. 15–19, 1999 proceedings 19. pp 398–412
11. Clavier C, Coron JS, Dabbous N (2000) Differential power analysis in the presence of hardware countermeasures. In: Cryptographic hardware and embedded systems—CHES 2000: second international workshop Worcester, MA, USA, Aug. 17–18, 2000 proceedings 2. pp 252–263
12. Wei L, Luo B, Li Y, Liu Y, Xu Q (2018) I know what you see: power side-channel attack on convolutional neural network accelerators. In: Proceedings of the 34th annual computer security applications conference, NY, USA. pp 393–406
13. Thiebauld H, Gannerot G, Wurcher A, Clavier C (2018) Scatter: a new dimension in side-channel. In: International workshop on constructive side-channel analysis and secure design. pp 135–152
14. Picek S, Heuser A, Jovic A, Batina L (2019) A systematic evaluation of profiling through focused feature selection. *IEEE Trans Very Large Scale Integr (VLSI) Syst* 27(12):2802–2815
15. Zheng Y, Zhou Y, Yu Z, Hu C, Zhang H (2014) How to compare selections of points of interest for side-channel distinguishers in practice? In: Information and communications security: 16th international conference, ICICS 2014, Hong Kong, China, revised selected papers 16. pp 200–214
16. Zaid G, Bossuet L, Habrard A, Venelli A (2020) Methodology for efficient CNN architectures in profiling attacks. *IACR Transactions on Cryptographic Hardware and Embedded Systems* 1–36
17. Wu L, Picek S (2020) Remove some noise: on pre-processing of side-channel measurements with autoencoders. *IACR Trans Cryptogr Hardw Embed Syst* 389–415

18. Gondara L (2016) Medical image denoising using convolutional denoising autoencoders. In: 2016 IEEE 16th international conference on data mining workshops (ICDMW). Barcelona, Spain, pp 241–246
19. Pammu AA, Chong KS, Ho WG, Gwee BH (2016) Interceptive side channel attack on AES-128 wireless communications for IoT applications. In: 2016 IEEE Asia Pacific conference on circuits and systems (APCCAS). Jeju, Korea (South), pp 650–653
20. Lo O, Buchanan WJ, Carson D (2016) Power analysis attacks on the AES-128 S-box using differential power analysis (DPA) and correlation power analysis (CPA). *J Cyber Secur Technol* 1(2):88–107
21. Potestad-Ordóñez FE, Tena-Sánchez E, Acosta-Jiménez AJ, Jiménez-Fernández CJ, Chaves R (2022) Design and evaluation of countermeasures against fault injection attacks and power side-channel leakage exploration for AES block cipher. *IEEE Access* 10:65548–65561
22. Paar C, Pelzl J (2009) Understanding cryptography: a textbook for students and practitioners. Springer Science & Business Media
23. Brier E, Clavier C, Olivier (2004) Correlation power analysis with a leakage model. In: Cryptographic hardware and embedded systems—CHES 2004: 6th international workshop Cambridge, MA, USA, proceedings 6. pp 16–29

Design and Implementation of a Secure AIoT-Based Health Promotion Prescription Systems for Community-Based Healthcare Promotion Services



Chien-Lung Hsu, Yu-Ting Chang, Pei-Wei Chen, Chieh-Ni Chen,
and Che-Wei Su

Abstract In the evolving realm of health care juxtaposed with cryptographic advancements, the present inquiry endeavors to seamlessly integrate the sophisticated capabilities of Artificial Intelligence of Things (AIoT) into geriatric care contexts. Notably, the conveyance and computational procedures encompassing AIoT-infused physical activity directives necessitate formidable security apparatuses. These meticulously designed protocols are quintessential to safeguarding the cardinal principles of integrity, confidentiality, authenticity, and availability pertaining to the physical activity regimens. Moreover, they underpin the evidentiary validity of the said data. In this scholarly pursuit, we have conceptualized a temporally constrained attribute-based cryptographic technique, complemented by a signature continuum. This paradigm facilitates users in signing and ciphering their directives in accordance with their designated roles and prerogatives. It acts as a bulwark against potential adulteration or unwarranted disclosures from cloud interfaces to exercise equipment, thereby fortifying both confidentiality and integrity. Furthermore, exercise apparatuses are granted access to these salutary directives contingent upon their stipulated access credentials and temporal authorization. Intruders and unauthorized systems find themselves thwarted, ensuring unparalleled privacy and impenetrable data sanctity for users.

C.-L. Hsu (✉) · P.-W. Chen · C.-N. Chen · C.-W. Su
Department Information Management, Chang Gung University, Taoyuan 33302, Taiwan
e-mail: clhsu@mail.cgu.edu.tw

C.-L. Hsu
Healthy Aging Research Center, Chang Gung University, Taoyuan 33302, Taiwan

Department Visual Communication Design, Chi University of Technology, Ming New
Taipei 24301, Taiwan

Department Nursing, Taoyuan Chang Gung Memorial Hospital, 333, Taoyuan, Taiwan

C.-L. Hsu · Y.-T. Chang
Graduate Institute of Management, Chang Gung University, Taoyuan 33302, Taiwan

Keywords Health promotion prescription · Artificial intelligence of things · Attribute-based · Signcryption · Time-bound

1 Introduction

The use of Artificial Intelligence of Things (AIoT) devices in healthcare is becoming increasingly popular due to the many benefits they provide. One of these benefits is the ability to provide personalized health promotion prescriptions for elderly patients. As the population ages, it becomes increasingly important to take preventative measures to delay the onset of age-related functional decline. Regular exercise is one such measure that can help minimize the effects of aging. Healthcare providers can prescribe exercise as a means of behavior change to encourage patients to incorporate physical activity into their daily lives. In health promotion, exercise machines and physiological measurement devices can upload data to a storage platform after exercise or physiological measurement. However, Internet of Things (IoT) devices are vulnerable to attacks during transmission, such as man-in-the-middle attacks and replay attacks. Therefore, security and privacy are crucial issues in IoT devices. To address these issues, a time-bound attribute-based signcryption method with signature chain was designed for the transmission and management of physical activity prescriptions and data. This method allows users to sign and encrypt their prescriptions and data based on different roles and privileges, preventing tampering and disclosure of physical activity prescriptions and data through the cloud platform to the exercise machine. The exercise machine can access the health promotion prescriptions and data according to its access privileges and authorized time period, providing personalized smart health promotion services while protecting the privacy of users and the security of data.

2 Related Works

2.1 *Health Promotion Prescription*

Advancements in medicine and technology have increased the average lifespan worldwide, making it crucial to address disability and dementia prevention in the elderly. Regular exercise for seniors can reduce depression, enhance sleep quality, boost cognitive function, and alleviate functional decline, ultimately improving their quality of life [1]. Health plans typically consider age, gender, physical health, and exercise habits to tailor specific exercise routines. When creating these plans, it's essential to focus on motivation, individual capabilities, and interests to effectively achieve desired goals [2].

A complete health promotion prescription should include the following five basic elements [3]:

1. Modality: the chosen mode of exercise, such as walking, jogging or swimming.
2. Frequency: the number of exercise sessions per week, such as five times per week.
3. Time or Duration: the length of each exercise session, including warm-up and cool-down exercises, such as 30 min per session.
4. Intensity: the level of exercise intensity
5. Rate of Progression: The exercise content will be gradually established according to different physical and mental conditions.

Usually, health promotion prescriptions require adjustments based on each individual's response to exercise [4], and may also modify goals or directions due to different stages, with the main goal being to maintain regular exercise. The process of formulating an health promotion prescription includes four steps [Fang92]:

1. Assessment: including baseline surveys (including health status, exercise history, etc.) and physical fitness testing.
2. Prescription design: When creating the prescription, factors like personal physical and mental conditions, environment, and goals should be taken into account. The health promotion prescription follows a principle of gradual progress across three stages: initiation, improvement, and maintenance.
3. Prescription execution: During the health promotion prescription execution period, encourage participation, observe the response and adaptation to exercise, and record the process of executing the prescription at any time.
4. Evaluation and review: After using the health promotion prescription for a while, it's important to assess and review its effectiveness by considering satisfaction and improvements in physical fitness.

2.2 *AIoT*

Internet of Things (IoT) was originally a term and concept created by Kevin Ashton at MIT. It refers to a global intelligent network that connects many transactions and has the ability to perceive, compute, execute, and communicate with the Internet, achieving interaction between networks and physical objects [5]. IoT is also associated with sensors, software, or other technological devices that can transmit and receive data from other devices. With the advancement of network technology, Bluetooth, Wi-Fi, 5G, and the emergence of artificial intelligence, these emerging technologies help humans analyze complex systems and make decisions. The combination of artificial intelligence and IoT is called the Internet of Intelligent Things (IoIT). IoIT is an emerging concept that aims to understand and connect "smart things" to make communication between various networks and systems more efficient [6]. Through IoIT, data can be collected and transformed into useful information, laying the foundation for AIoT Data as a Service [7]. Nowadays, the application of IoIT is

becoming more and more common, such as in healthcare, wearable devices, smart homes, and environmental control sensors.

2.3 Signcryption

In 1997, Zheng introduced signcryption [Zheng97], which aims to ensure both message confidentiality and unforgeability. It merges encryption and signature into a single step, where the signcryptor simultaneously encrypts and signs the message, producing the ciphertext and signature. To decrypt the message and verify its accuracy, the signcryptor uses their private key and signature. Signcryption is a fusion of encryption and signature, offering reduced computational and communication complexity compared to the combined cost of separate encryption and signature processes.

2.4 Attribute-Based Signcryption

Attribute-Based Signcryption (ABSC) combines the functions of attribute-based encryption and attribute-based signature [8]. Users can access attributes through attribute-based encryption and signature, which can encrypt and protect messages while verifying their authenticity. In ABSC, the attributes used in signature and encryption are different.

The use of attributes results in very high computational costs, and most access structures use Shamir's threshold method. As the number of attributes increases, the computational cost increases exponentially. The use of Linear Secret-Sharing Scheme (LSSS) as the access structure for attribute-based signcryption can improve the computational cost problem.

3 Illustrations

3.1 Architecture

The system architecture in this chapter (e.g., Fig. 1) involves eight main roles: Attribute Authority (AA), User, Device, Agent, Gateway (GW), Server, Timestamp Server, and Storage Cluster. The Attribute Authority manages public and secret parameters, facilitates user and device registration, grants public parameters to devices for signcryption, provides attribute private keys to users, and handles course

application approvals. It also sets course validity intervals and generates course information. The Timestamp Server ensures accurate time. Devices receive health promotion prescriptions from the server. Agents installed in devices define access policies and signcrypt exercise data to produce signcryptured ciphertext, enabling attribute-based access control. Gateways verify received signcryptured ciphertext for accuracy. Servers store the signcryptured ciphertext in the storage cluster. Users retrieve signcryptured ciphertext from the server and decrypt exercise data using their attribute private key. If the attribute matches, they gain access to the decrypted exercise data.

Table 1 shows the corresponding algorithms and function descriptions for each stage, and Table 2 shows the notation explanations used in the system. The system consists of the system initialization stage, device registration stage, user registration stage, course registration stage, secure transmission stage of health promotion prescriptions, encryption and signing stage, verification stage, and decryption stage. Different algorithms are used to achieve security for different stages.

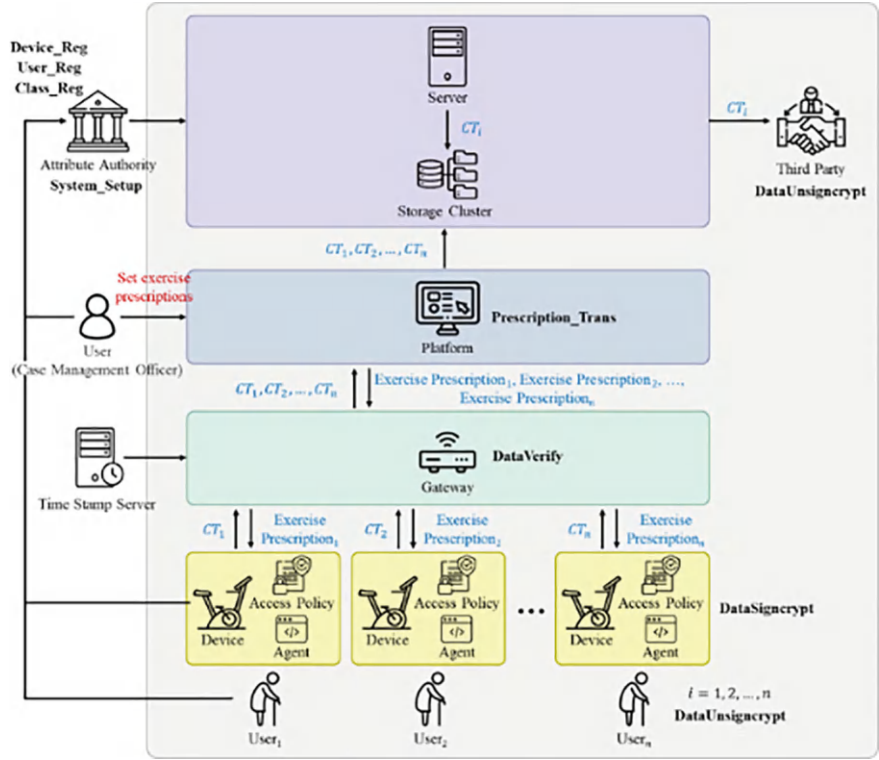


Fig. 1 System architecture

Table 1 Flow corresponding algorithm and function table

Stage	Algorithm	Description
System initialization phase	System_Setup	Generation of system parameters and public parameters
Device registration phase	Device_Reg	Registration of equipment and systems
User registration phase	User_Reg	User and system registration
Course registration phase	Class_Reg	Start-up phase, determination of start-up time and section
Secure transmission of health promotion prescription phase	Prescription_Trans	Health promotion prescriptions are transferred from the server to the device
Signcryption phase	Data_Signcrypt	Signcryption of exercise data into a ciphertext with a signature
Verification phase	Data_Verify	Verify the signcryption ciphertext
Decryption phase	Data_Unsigncrypt	Verify the signature and decrypt the ciphertext

Table 2 Explanation of notation

Notion	Descriptions	Notion	Descriptions
P, P	Public parameters	σ_{Ser}	Server's signature
MSK	Secret parameters	t	Time stamp
Y	System public key	IP	IP address
α	System private key	$H()$	A secure hash function
g^s	Signer's public key	$ECDSA()$	ECDSA signature generation algorithm
s	Signer's private key	$Verify_{ECDSA}()$	ECDSA signature verification algorithm
M	Health promotion prescription	$\vec{v}_e$	Secret vector
C	Ciphertext	BF	Boolean formula of access policy
σ_{CT}	Signature	x	Bit length of a row vector in a access control matrix
σ_{dev}	Signature of an AIoT device	ID_i	Identifier for user i
σ_{Gw}	Signature of a gateway	DID_i	Device identifier for device i

3.2 System Method

System Initialization Phase

The system initialization stage is set by the attribute organization, and the public and secret parameters are generated through the **System_Setup** algorithm, as well

as the randomly selected random numbers x_s and x_e . The **System_Setup** algorithm is defined as follows:

Input: Security parameters K

Output: PP, MSK, x_s, x_e

Algorithm

1. Choose a large prime q and determine the elliptic curve equation $E : y^2 = x^3 + ax + b \pmod{q}$.
2. Generate an additive group G_1 and a multiplicative group G_T such that its bilinear mapping function is $e : G_1 \times G_1 \rightarrow G_T$.
3. Set g to be a G_1 generator.
4. Set the system attribute domain $u_s = \{att_1, att_2, \dots, att_x\}$.
5. Select the corresponding random numbers $\{Q_1, Q_2, \dots, Q_x\} \in_R G_1$ according to the attribute u_s .
6. Define a safe one-way hash function $H : \{0,1\}^* \in G_1$.
7. Randomly select random numbers $\alpha, \beta \in_R Z_q$ and random numbers x_s, x_e .
8. Calculate the public key $Y = e(g, g)^\alpha$.
9. Calculate $B = g^\beta$.
10. System output $PP = (g, B, Y, H, Q_n, e, G_1, G_T)$, $MSK = (\alpha, \beta)$ and random numbers x_s, x_e .

Device Registration Phase

In the device registration phase, the device will be registered through the attribute authority. The attribute agency will execute the **Device_Reg** algorithm to obtain the attribute public parameters and send them to the device D_i through a secure channel. The **Device_Reg** algorithm is defined as follows:

Input: DID_i

Output: PP'

Algorithm

1. Send the device identifier DID_i to the attribute authority.
2. The attribute authority receives the device identifier DID_i and verifies the device with the identifier DID_i .
3. Output attribute public parameter $PP' = (g, B, Y, H, Q_n, e, G_1, G_T)$.

User Registration Phase

In the user registration phase, user U_i obtains the attribute private key SK_{ID_i} from the attribute authority with the identifier ID_i . The attribute authority will execute the **User_Reg** algorithm to generate the corresponding attribute private key SK_{ID_i} and send it to user U_i through the secure channel. The **User_Reg** algorithm is defined as follows:

Input: ID_i, PP, MSK

Output: SK_{ID_i}

Algorithm

1. When the attribute authority receives the identification code ID_i , it verifies the identity of this ID_i .
2. Generate $t_{ID_i} \in_R Z_q$ randomly for this identity.
3. Calculate K^i using the parameters $g, \alpha, \beta, t_{ID_i}$.
4. Use parameters g, t_{ID_i} to calculate L^i .
5. Use parameters t_{ID_i}', Q_x to calculate K_j^i .
6. Output attribute private key is $SK_{ID_i} = (K^i, L^i, K_j^i)$.

Class Registration Phase

In the class registration phase, user U_i uses the identification code ID_i to apply to the attribute authority to start a class. The attribute authority will execute the Class_Reg algorithm to obtain the class start information CL , and send it to the gateway for storage by the secure channel.

Input: ID_i

Output: CL

Algorithm

1. When the attribute authority receives the identification code ID_i , it verifies the identity of the ID_i .
2. Randomly generate $a \in_R Z_q$.
3. Calculate the class code Cl_{ID_i} using ID_i, a .
4. Select the access time t_1 and t_2 .
5. Use parameters x_s, t_1 to calculate T_1 and x_e, t_2 to calculate T_2 .
6. Output class information $CL = (Cl_{ID_i} || (H(H(T_1)) || H(T_2)))$.

Prescription Secure Transmission Phase

In the prescription security transmission phase, the server performs the Prescription_Trans algorithm to sign the health promotion prescription using the access policy boolean function BF and the public parameter PP that defines the access policy. After completing the signcryption, the signed ciphertext CT is generated and sent to the Gateway. The **Prescription_Trans** algorithm is defined as follows:

Input: BF, M, PP

Output: CT

Algorithm

1. Set the LSSS matrix m_e by accessing the policy boolean formula.
2. Randomly generate the random number r_j .

3. Generate the row matrix $\vec{v}_e$, and the matrix for randomly generated signature private key S , random number a_n .
4. Using matrix m_e and row matrix $\vec{v}_e$ to calculate λ_e .
5. Calculate C' using the signer's private key s and parameter g
6. Calculate C_j using parameter B , random number r_j and matrix λ_e .
7. Using the random number r_j and parameter g to calculate D_j .
8. Calculate T using the present time t and the random numbers x_s, x_e .
9. Calculate h by using the secure one-way hash function H for the prescription M .
10. Using the prescription M by using the system public key Y and the signer's private key s and T to calculate the health promotion prescription C for the signatory.
11. Calculate the health promotion prescription signature σ from the parameters h , system public key Y and the signer's private key s .
12. Output $CT = (\sigma, C, C_j, C', D_j, m_e)$.

Signcryption phase

In the signcryption phase, device D_i executes the **Data_Signcrypt** algorithm to sign the data using the access policy boolean function BF and the public parameter PP that defines the access policy. After completing the encryption, the ciphertext CT and signature σ_{dev} are generated and sent to the Gateway. The **Data_Signcrypt** algorithm is defined as follows:

Input: BF, M, PP

Output: CT, σ_{dev}

Algorithm

1. Setup the LSSS matrix as m_e .
2. Randomly choose a integer r_j .
3. Compute the column vector $\vec{v}_e$ and obtain s and a_n .
4. Use the matrix m_e and column vector $\vec{v}_e$ to compute λ_e .
5. Use s and g to compute C' .
6. Use the parameter $\{B, r_j, \lambda_e\}$ to compute C_j .
7. Use the random integer r_j and the parameter g to compute D_j .
8. Use the M and a secure one-way hash function H to compute h as $h = H(C)$.
9. Use the public key Y and the private key s to compute the ciphertext C .
10. Use $\{h, Y, s\}$ to compute the digital signature σ .
11. Use $\{\sigma_{CT}, IP_{dev}, t_{CT}\}$ to generate an ECDSA digital signature σ_{dev} .
12. Output $CT = (\sigma, C, C_j, C', D_j, m_e, IP_{dev}, t_{CT})$ and $\delta_{dev} = (\sigma_{dev}, IP_{dev}, t_{dev})$.

Verification Phase

In the verification phase, the gateway verifies the received signed ciphertext and uses the exposed parameters to Data_Verify the **GWCTPP** algorithm. When the signed ciphertext CT passes verification, the output *True* indicates that the verification is

successful, valid, and the signature belongs to the system and is transmitted to the server; otherwise, it is False. Algorithm Data_Verify is defined as following:

Input: CT, PP

Output: δ_{GW}

Algorithm

1. Verify if e .
2. Verify if ECDSA signature σ_{IoT} is valid.
3. Generate a ECDSA signature σ_{GW} by using the parameters $\{\sigma_{CT}, \sigma_{dev}, t_{GW}\}$.
4. Output $\delta_{GW} = (ECDSA_{GW}, t_{GW})$.

Data_Unsigncrypt Phase

In the de-signing phase, the user, U_i can only obtain the motion data ciphertext CT signed by the IoT device, $Device_i$, from the server server within the set valid interval. The user, U_i uses the attribute private key SK_{ID_i} and the public parameters PP to execute **Data_Unsigncrypt** algorithm to obtain plaintext. **Data_Unsigncrypt** algorithm is defined as follows:

Input: CT, SK_{ID_i}, PP

Output: M

Algorithm

1. Use the valid m_e to obtain the unique solution w_i .
2. The user U_i use the parameters $\{C, C', D_i, SK_{ID_i} = (K^i, L^i, K_j^i)\}$ to obtain the decryption key as Y^s .
3. Compute $h = H(C)$ by using the secure one-way hash function H .
4. Use the parameters $\{g, Y, C, C'\}$ to verify the digital signature σ_{CT} .
5. Use the decryption key Y^s to decrypt C and obtain the message M .

4 Security Analysis

4.1 Security Analysis Using AVISPA Tool

AVISPA is an automated tool for verifying internet security protocols and applications [ABCR06]. It utilizes modular and expressive HLPSP as its formal language and incorporates various backends in the verification process. These backends include real-time model checker (OFMC), constraint logic-based attack searcher (CL-AtSe), SAT-based model checker (SATMC), and security protocol analysis based on tree automata (TA4SP). However, SATMC and TA4SP are less commonly used as they lack the ability to leverage algebraic properties of modular exponentiation and XOR operators for protocol verification. AVI is a simulation comprising the AVISPA tool and Security Protocol ANimator (SPAN) to offer a user-friendly interface.

The simulator's HLPSL code defines three roles: Attribute Authority (A), Device and Sensor (S), and User (U). Symmetric keys protect messages during user registration, and to simplify, the attribute authority key is set as α . AVISPA supports three operators: concatenation, exclusive or, and exponentiation. Multiplication and pairing functions act as hash functions. Additionally, we assume ECDSA and $\text{inv}(\text{ECDSA})$ serve as the public and private keys for executing the ECDSA algorithm.

4.2 Semantic Security Proof

Our proposed protocol provides secure decryption keys, signature verification and data integrity, non-repudiation, tamper resistance, perfect forward secrecy, and data confidentiality. The specific semantic security proofs of the protocol are as follows:

1. Secure decryption keys: Based on the elliptic curve discrete logarithm problem (ECDLP), the adversary cannot retrieve the secret value s from C' . The secret value α is also hidden in K^i , which is even confidential to only the user and attribute authority. Therefore, the attacker cannot obtain g^s and g^α to calculate the decryption key $Y^s = e(g, g)^{s\alpha}$. The key can only be successfully calculated when a legitimate user performs the correct pairing operation with matching attributes. Thus, we can say that the proposed protocol achieves secure decryption keys.
2. Robust signature verification and data integrity: In our protocol, the signature of the ciphertext is verified to ensure authenticity. The correctness proof of the single signature σ_{CT_1} verification is as follows:

$$e(h, Y_{\cdot x} C') = e(h, Y_{\cdot x} g^s) = e(h, g^{Y_{\cdot x} s}) = e(h^{Y_{\cdot x} s}, g) = e(\sigma_{CT_1}, g)$$

Assuming there are two signatures in the batch signature, the following equation proves the correctness of batch signature verification (including σ_{CT_1} and σ_{CT_2}).

$$\begin{aligned} le(h_1 h_2, Y_{\cdot x} Y_{\cdot x} C'_1 C'_2) &= e(h_1 h_2, Y_{\cdot x} g^s Y_{\cdot x} g^s) \\ &= e(h_1 h_2, g^{Y_{\cdot x} s} g^{Y_{\cdot x} s}) = e(h_1^{Y_{\cdot x} s} h_2^{Y_{\cdot x} s}, gg) \\ &= e(\sigma_{CT_1} \sigma_{CT_2}, gg) = e(\text{BSig}, gg) \end{aligned}$$

Therefore, the signature can be verified correctly. After the verification and signature are sent by the device, integrity is achieved, and the conclusion holds.

3. Unforgeability of the signature: If an attacker wants to forge the signature $\sigma = h^{Y_{\cdot x} s}$, they must obtain the correct value of s . However, as mentioned earlier, the value s is protected by the elliptic curve discrete logarithm problem ECDLP, and the attacker cannot calculate s to forge the signature σ . Therefore, the unforgeability of the signature is achieved.

4. Data confidentiality: In order to decrypt the ciphertext, an attacker must obtain $SK_{ID_i} = (K^i, L^i, K_j^i)$ and compute Y^s to decrypt the ciphertext C . However, only users registered with the attribute authority have the correct attributes and the corresponding private key SK_{ID_i} . As mentioned earlier, the security of the decryption key Y^s is also guaranteed. In addition, attackers do not know $K = \alpha, \beta, u_s$, and cannot compromise the system, thus achieving complete confidentiality.
5. Non-repudiation: In the course of our agreed communication, the private key s is known only to the device, and the device cannot recognize the signature it has signed.
6. Perfect forward secrecy: During signing and encryption, the device picks the key “ s ” to compute the ciphertext and generate the decryption key Y^s . As “ s ” is randomly chosen, Y^s becomes a random number. So, if an adversary gets the current session’s decryption key, they can’t retrieve keys from previous sessions, ensuring perfect forward secrecy in our protocol.

5 Conclusion

Ensuring the security and data integrity of communication between IoT devices is necessary to consider as the IoT becomes more prevalent. This study proposes a method for safe data transmission between IoT devices using linear secret sharing, attribute-based signcryption, and BLS short signature methods. The trust chain is achieved through the immutability of blockchain.

The study proposes a trustworthy AIoT-based health promotion prescription system method that allows for secure transmission of health promotion prescriptions and exercise data between devices and the cloud. The system uses attribute-based signcryption to achieve access control mechanisms, ensuring the confidentiality of health promotion prescriptions and exercise data and achieving fine-grained access control. Additionally, the system can customize access rules based on different contexts and needs to enhance system flexibility. The study also uses AVIPSA tools and semantic security proof to demonstrate the security of the proposed method.

This study examines the use of AIoT devices in healthcare to provide personalized health promotion prescriptions for elderly patients. To ensure the security of AIoT-based physical activity prescriptions and data, a time-bound attribute-based signcryption method with signature chain was designed for the transmission and management of physical activity prescriptions and data. This method allows users to sign and encrypt their prescriptions and data based on different roles and privileges, preventing tampering and disclosure of physical activity prescriptions and data through the cloud platform to the exercise machine. The exercise machine can access the health promotion prescriptions and data according to its access privileges and authorized time period, providing personalized smart health promotion services while protecting the privacy of users and the security of data.

Acknowledgements This work was partially supported in part by the National Science and Technology Council in Taiwan under Grant No. NSTC-111-2410-H-182-007-MY2, Grant No. NSTC-112-2811-H-182 -002, No. NSTC-111-2221-E-155-038, and No. NSTC-111-2218-E-218-004-MBK. It was also supported in part by the Healthy Aging Research Center.

References

1. Weixuan L, Yuyun L, Junyi D (2009) Exercise prescription for the elderly. *Prim Med* 24(1):8–13
2. Lin J (2012) ACSM exercise testing and exercise prescription guidelines
3. Yang Y Physical activity and health. Accessed 12 2022. <http://stud.adm.ncku.edu.tw/hea/4work/wedu/e2/e232/pah.htm>
4. Cantu RC, Micheli LJ (1991) ACSM's guidelines for the team physician
5. Zheng Y (1997) Digital signcryption or how to achieve $\text{cost}(\text{signature} \ \& \ \text{encryption}) \ll \text{cost}(\text{signature}) + \text{cost}(\text{encryption})$. In: *Advances in cryptology—CRYPTO'97: 17th annual international cryptology conference Santa Barbara, California, USA August 17–21, 1997 Proceedings* 17. Springer Berlin Heidelberg, pp 165–179
6. Guo T, Yu K, Aloqaily M, Wan S (2022) Constructing a prior-dependent graph for data clustering and dimension reduction in the edge of AIoT. *Futur Gener Comput Syst* 128:381–394
7. Christensen G (2019) Artificial intelligence of things (AIoT), IoT Adenda. <https://internetofthingsagenda.techtarget.com/definition/Artificial-Intelligence-of-Things-AIoT>
8. Chen W (2019) Research on blockchain-based IoT log management system with attribute access control. Master's thesis of Chang Gung Memorial University
9. Zhang J, Tao D (2020) Empowering things with intelligence: a survey of the progress, challenges, and opportunities in artificial intelligence of things. *IEEE Internet Things J* 8(10):7789–7817

Financial Lending Application Based on Blockchain Technology



Zhe-You Liu, Ting-Yu Lin, Min-Teng Chen, and Pei-Yu Lin

Abstract Decentralized blockchain technology has revolutionized finance applications. It secures data through encryption and hashing in interconnected blocks, enabling multiple users to access and share information while preventing unauthorized alterations. This decentralized ledger removes the need for third-party verification and minimizes the risk of system failures or attacks, reducing reliance on centralized entities like governments and banks. Moreover, it lowers transaction costs, automates smart contracts, and provides access to off-chain data through solutions like ChainLink. In this article, we integrate smart contracts and ChainLink to implement a lending system on the Avalanche-C chain. This system allows users to lend stablecoins against collateral, with automatic liquidation triggered when the health factor exceeds a set threshold. Notably, it offers low transaction fees (gas < 0.1), provides liquidity without selling assets, and leverages ChainLink for off-chain information, enabling timed execution of internal contracts, among other benefits.

Keywords Blockchain · Smart contract · Defi · ChainLink · Avalanche-C

1 Introduction

The concept of blockchain was first introduced by Satoshi Nakamoto in 2008 [1]. Important information or transaction records can be considered sensitive data. To ensure the security and integrity of sensitive data, users need to use private keys to digitally sign the data. Miners then take the encrypted data, add timestamp information, and obtain a hash value through a hash function. This hash value and related data can be regarded as a block, connecting it with previous blocks, as shown in Fig. 1. Users can use mechanisms like Proof of Work or Proof of Stake [2] to ensure that the calculated hash value is correct, guaranteeing security and tamper resistance. Finally,

Z.-Y. Liu · T.-Y. Lin · M.-T. Chen · P.-Y. Lin (✉)

Department of Electrical Engineering, National Kaohsiung University of Science and Technology,
415 Jiangong Rd., Sanmin Dist., Kaohsiung City 807618, Taiwan

e-mail: pagelin@nkust.edu.tw



Fig. 1 Blockchain connections [1]

the block is broadcasted throughout the network in a peer-to-peer manner, achieving decentralized storage and transparency. Due to these characteristics of blockchain, it has brought significant changes and novel applications not only in the field of cryptocurrencies but also in areas such as financial services, smart contracts, supply chain management, and healthcare [3–5].

In recent years, technology developments related to blockchain have been continuously advancing. These advancements encompass fundamental technologies and applications, including smart contracts, decentralized finance (DeFi), and non-fungible tokens (NFTs). In response to the critical challenges of blockchain, such as achieving security and scalability while maintaining decentralization, several initiatives and technologies have been proposed. These efforts encompass blockchain interoperability, scalability solutions, and privacy protection measures.

The concept of smart contracts was originally introduced by Nick Szabo [6]. It involved the idea of encoding contracts as code, but due to technological limitations at the time, the rapid evolution of smart contracts didn't occur until the introduction of Ethereum. Smart contracts, once deployed on a blockchain, automatically execute when predefined contract conditions are met. The primary advantage of smart contracts lies in their autonomous operation, free from human intervention and third-party interference. Their decentralized nature ensures transparency and immutability of the contract's terms. Nevertheless, once deployed, correcting errors or vulnerabilities in a contract becomes a challenging task. Any flaws in a smart contract can lead to irreversible losses, as evidenced by the infamous case of TheDAO in June 2016, where a smart contract vulnerability was exploited, resulting in the theft of \$50 million worth of Ether [7, 8].

Traditional finance (Centralized Finance, CeFi) is a financial system in which financial institutions and central authorities control and manage financial operations, acting as intermediaries to process transactions and provide financial services while charging fees. It offers advantages such as stability, legal and regulatory protection, and high liquidity. However, it also has common drawbacks, including high transaction fees, slow transaction processing, trust risks associated with central financial institutions, potential fraud, and centralization of financial assets. These traditional issues have driven an increasing number of individuals in recent years to explore faster and more transparent decentralized finance (DeFi) [9].

DeFi leverages the concept of blockchain's distributed ledger to enable automated transactions through smart contracts, reducing both the time and cost of transactions and offering highly transparent public transaction records. Simultaneously, DeFi services can be accessed and used with a cryptocurrency wallet, without requiring

user identity verification (KYC/AML), making financial services more accessible to a broader audience. However, DeFi also comes with various risks, including smart contract vulnerabilities, significant market volatility, and regulatory challenges related to compliance.

In this article, we have implemented a financial lending protocol on Avalanche-C. Regarding the consensus mechanism, Ethereum [10] initially employed a Proof of Work (PoW) system similar to Bitcoin in its 1.0 version. However, Ethereum 2.0 introduced a Proof of Stake (PoS) mechanism. In contrast, Avalanche [11] uses a unique consensus mechanism, which involves multiple nodes conducting multiple random samples and verifies results by the majority. This approach offers a higher transaction throughput, making it suitable for DApps that require fast and large-scale transactions. In terms of transaction fees, Ethereum can experience substantial increases in gas fees due to transaction congestion. Conversely, Avalanche offers lower and relatively stable gas fees, which attracts a significant number of users and developers. As for scalability, Ethereum has made improvements recently by introducing sharding technology [10] in its 2.0 version. However, Avalanche surpasses Ethereum with a multi-chain design that includes Transaction Chains (X-Chain), Platform Chains (P-Chain), and Contract Chains (C-Chain). This design caters to high traffic loads and addresses the three major challenges faced by blockchain technology.

In the realm of smart contract languages, Ethereum primarily utilizes Solidity as the programming language for smart contracts and also supports Vyper, a more readable smart contract language. Avalanche, on the other hand, supports a variety of smart contract languages, including Solidity, Scilla, Ethereum Bytecode, providing a high degree of flexibility. In summary, Avalanche blockchain offers advantages such as faster transaction speeds, lower transaction costs, enhanced scalability, and greater flexibility, making it a suitable choice for developers to build DeFi projects.

In this article, we leveraged the concept of smart contracts to implement a lending system on the blockchain, achieving the goal of automated contract execution. The ChainLink [12, 13] is utilized to obtain real-time cryptocurrency valuations and periodically trigger internal contracts to determine whether liquidation is required. This lending protocol offers not only the advantage of low transaction costs but also enables users to access liquidity without selling their assets.

In the subsequent article structure, Sect. 2 will expound on the concept of ChainLink blockchain technology. Section 3 will delve into the framework and theory of the protocol we implemented. Section 4 will present the implementation results and analysis. Finally, Sect. 5 will encompass the conclusion and recommendations.

2 The Concept of ChainLink

To ensure the timely triggering of internal contracts for our designed financial lending application and to confirm the need for liquidation, we have utilized the blockchain technology known as ChainLink [12, 13]. This allows us to obtain real-time cryptocurrency values. ChainLink is currently the largest decentralized oracle

platform with the goal of acquiring real-world data (off-chain data) to bridge the gap between DeFi projects and the physical world within the blockchain. Given that each blockchain operates as a closed system unable to actively retrieve external data, the assistance of oracles is crucial to ensure the smooth operation of smart contracts. In the continually evolving DeFi landscape, the role of ChainLink oracles is paramount, as depicted in Fig. 2.

In general, centralized oracles may face the issue of a single point of failure, such as when an API data provider suddenly goes offline, when a data provider intentionally supplies incorrect data, or when a data provider falls victim to a hacker attack, resulting in tampered data. Due to these reasons, the data obtained may contradict the actual facts, leading to a loss of accuracy and reliability [14]. Decentralized oracles, on the other hand, operate in contrast to centralized oracles. They do not rely on a single data source but instead collect and verify data from multiple sources, thus enhancing the authenticity and reliability of the data.

ChainLink has a wide range of applications, including obtaining stock prices, currency exchange rates, supply chain inventory management, transportation conditions, and Internet of Things (IoT) data in the field of financial services. Common use cases include Aave [15], a DeFi platform that uses ChainLink to determine cryptocurrency prices, and the popular NFT game Axie Infinity [16], which also utilizes ChainLink to establish the prices of virtual assets. The decentralized legal contract platform OpenLaw [17] integrates ChainLink’s data to support the terms in legal contracts related to financial products and legal agreements.

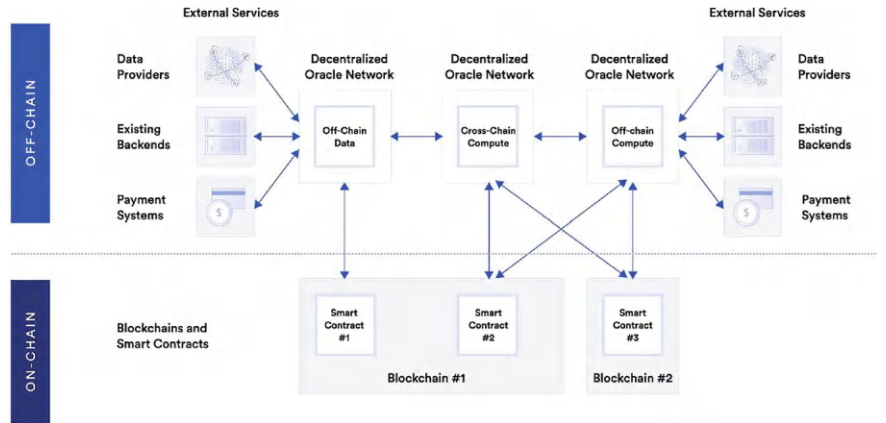


Fig. 2 Operations of ChainLink oracles [12]

3 The Framework of the Implementation System

In this section, the implementation and testing process of the financial lending system is described. Figure 3 shows the framework of the system. The financial system model is detailed in Sect. 3.1, whereas Sect. 3.2 is dedicated to the development and testing of smart contracts.

3.1 The Financial Model

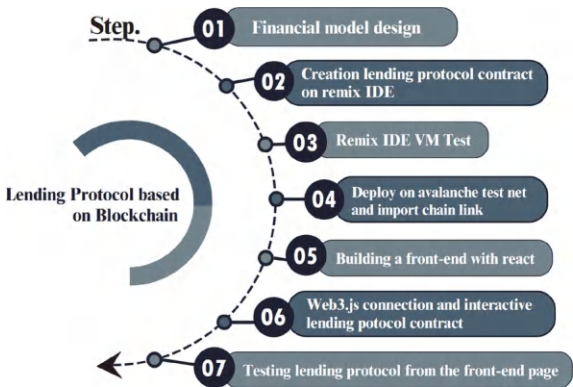
The financial lending application operates using AAVE technology [15]. Within the protocol, there are two types of tokens: USDC and AVAX. The financial system involves three distinct roles: the first is referred to as the liquidity provider, the second as the liquidator, and the third as the borrower. When a liquidity provider locks their USDC and AVAX into the protocol, it's referred to as Reserve. When a liquidator locks their USDC and AVAX into the protocol, it's called Liquidate.

When a borrower seeks to obtain a loan, they are required to deposit cryptocurrency into the protocol as collateral, acting as a lender themselves. At this point, the protocol calculates the collateral's market value for the borrower, referred to as Collateral (USD). The calculation method is as follows, with the AVAX price obtained through ChainLink:

$$Collateral(USD) = Supply_{AVAX} \times Price_{AVAX} + Supply_{USDC} \tag{1}$$

After calculating the collateral value, the following Eqs. (2) and (3) are compared based on the specific token for which the borrowing is required (If the Reserve for either of the two tokens is less than the corresponding token supply from the borrower, the loan cannot proceed directly because liquidators would be exposed to the collateral shortfall, leading to potential losses). Equation (2) checks if the

Fig. 3 The framework of the implementation system



current Reserve locked within the protocol is greater than 0.8 times the value of the borrower's collateral, with the parameter set at 0.8 to ensure over-collateralization, ensuring that the collateral's value exceeds the borrowing value. Equation (3) is employed to assess whether the Liquidate ($Liquidate_{AVAX}$ or $Liquidate_{USDC}$) locked by the liquidator exceeds 0.88 times the borrower's collateral value. The parameter value of 0.88 is flexible for adjustment but must be greater than 0.8 times the collateral value. This is primarily to facilitate the subsequent liquidation process, which will be explained in the later section.

$$\begin{cases} Reserve_{AVAX} \geq \frac{Collateral(USD) \times 0.8}{Price_{AVAX}}, \\ Reserve_{USDC} \geq Collateral(USD) \times 0.8. \end{cases} \quad (2)$$

$$\begin{cases} Liquidate_{AVAX} \geq \frac{Collateral(USD) \times 0.8}{Price_{AVAX}} \times 1.1, \\ Liquidate_{USDC} \geq Collateral(USD) \times 0.8 \times 1.1. \end{cases} \quad (3)$$

With the two previous comparisons, the protocol can determine and adjust the maximum amount that can be lent to the borrower, either $AVAX\_MAX$ or $USDC\_MAX$, with the choice between the two. When both Eqs. (2) and (3) are satisfied simultaneously, the maximum amount that the borrower can borrow is as follows:

$$\begin{cases} AVAX_{MAX} = \frac{Collateral(USD) \times 0.8}{Price_{AVAX}}, \\ USDC_{MAX} = Collateral(USD) \times 0.8. \end{cases} \quad (4)$$

And when Eq. (2) is satisfied but Eq. (3) is not, the following applies:

$$AVAX_{MAX} = \frac{Liquidate_{AVAX}}{1.1}, USDC_{MAX} = \frac{Liquidate_{USDC}}{1.1} \quad (5)$$

On the other hand, when Eq. (3) is satisfied but Eq. (2) is not, the following scenario applies:

$$AVAX_{MAX} = Reserve_{AVAX}, USDC_{MAX} = Reserve_{USDC} \quad (6)$$

Finally, once the borrower successfully obtains funds from the protocol, the borrower's collateral will be temporarily removed from the Reserve until it is fully repaid before being released back into the Reserve. Meanwhile they will receive a percentage value known as the health factor. This value serves to promptly inform and show the borrower the risk level of their collateral within the protocol. The calculation method for the health factor is as follows:

$$health\ factor = \frac{Borrowed(USD)}{Collateral(USD)} \times 100\% \quad (7)$$

Here, the value of $Borrowed(USD)$ can be generated as:

$$Borrowed(USD) = Borrowed_{AVAX} \times Price_{AVAX} + Borrowed_{USDC} \quad (8)$$

In an ideal scenario, when both Reserve and Liquidate are sufficient, the maximum borrowing limit for the borrower is 80%, which is the initial value upon borrowing. Due to the fluctuation in the price of AVAX and the periodic triggering of the interest rate increment function within the protocol by ChainLink, the health factor fluctuates. Therefore, the protocol is designed to automatically initiate liquidation when the health factor reaches 90%. The liquidation process is as follows:

Step 1: The protocol transfers all of the borrower's Collateral to the liquidator.

Step 2: The protocol automatically locks the liquidator's Liquidate assets in the agreement and charges a fee for purchasing Collateral below market value, calculated using Eq. (9). In this formula, "Borrowed" represents different funds borrowed by the borrower from the protocol.

$$Borrowed_{AVAX} \times 1.02 + Borrowed_{USDC} \times 1.02 \quad (9)$$

Step 3: The protocol utilizes the funds collected in Step 2 to replenish the Reserve, which covers the entire funding gap created when the borrower borrows from the protocol, calculated as

$$Borrowed_{AVAX} + Borrowed_{USDC} \quad (10)$$

Step 4: The remaining amount of funds collected in Step 3 is determined by multiplying the value of Eq. (9) by 0.02. These funds are distributed to the capital providers. This represents the profit generated, especially in the worst-case scenario when the borrower is liquidated. The primary source of income for capital providers is the interest paid by the borrower. Additionally, any extra profits from liquidation can also be decided through a public vote in the protocol.

After the completion of the liquidation process, the liquidator receives collateral acquired at a price lower than the market value, and the price difference constitutes the liquidator's profit. However, if the liquidator immediately sells this collateral on the market, their profit can be calculated by Eq. (11). The liquidator's actions may be influenced by market conditions and liquidity, leading to fluctuations in their profit. In extreme market conditions, their profit may even turn negative, typically stemming from insufficient demand among buyers, resulting in an increased price spread. It's worth noting that this protocol shifts the risk to the liquidator, facilitating risk diversification and relieving the liquidity providers from concerns about potential contract losses.

$$IProfit = Collateral(USD) - (Borrowed_{AVAX} \times 1.02 \times Price_{AVAX} + Borrowed_{USDC} \times 1.02) > 0 \quad (11)$$

3.2 *Smart Contract Creating and Testing*

The system first creates an ERC-20 compliant USDC contract on Remix IDE to facilitate subsequent testing with the ability to freely mint and manage tokens. Subsequently, USDC is imported into the newly created lending protocol contract for invocation. Following a pre-designed financial model, the lending protocol is established. Before deploying to the blockchain, initial internal testing is conducted within Remix IDE's Remix VM (Shanghai). These actions are simulated only locally and not deployed on the chain. At this stage, AVAX does not need to be transmitted into the protocol through ChainLink. An internal contract function can be created to allow external calls and manual adjustment of the AVAX price. The testing within Remix VM (Shanghai) primarily focuses on ensuring the normal operation of the protocol, including supply, borrowing, interest calculations, and liquidation processes.

When local Remix VM (Shanghai) testing operates smoothly, the implementation system switches the Environment to Injected Provider-MetaMask and then switches MetaMask to the Avalanche Test net. Using Remix, the lending protocol contract and the USDC contract are deployed on the Avalanche Test net. Subsequently, ChainLink is used to acquire LINK tokens on the Avalanche Test net. A new Upkeep is registered, and Time-based is selected as the Trigger. Our system submits the ABI generated by Remix after compiling the lending protocol contract to ChainLink. Finally, the LINK token fee is paid, and ChainLink is informed to trigger specific functions within the lending protocol contract at a designated time. This allows the protocol to execute periodic checks and interest calculations as planned. In Fig. 4, external AVAX pricing is introduced, with the Data feed contract address being: $0 \times 5498BB86BC934c8D34FDA08E81D444153d0D06aD$. In this scenario, the protocol has incorporated external pricing from ChainLink and configured the state to execute periodic checks and interest calculations, facilitating interaction and testing within the Remix IDE and deployment of the lending protocol contract on the Avalanche Test net.

4 **Implementation Results and Analysis**

4.1 *Front-End Interface and React*

In order to facilitate interaction between ordinary users and lending protocols on the blockchain, this implementation system creates a simple front-end interface that allows users to interact with the blockchain, as shown in Fig. 5. Here, we use React to build a basic front-end interface in VS Code and use the Web3.js [18] package to connect to the blockchain. The interface displays the user's wallet address, current AVAX price, Health Factor, and the user's asset status within the protocol, among

```
function borrowing_interest()public returns(bool){
    uint256 i=0;
    uint256 target=User_address.length;
    address user_address;
    uint256 temp=block.timestamp;
    while(i<target){
        user_address=User_address[i];
        if(interest_time[user_address]!=temp){
            interest_time[user_address]=temp;
            if(User_borrow_AVAX[user_address]>0){
                User_borrow_AVAX[user_address]=(User_borrow_AVAX[user_address]*110)/100;
            }
            if(User_borrow_USDC[user_address]>0){
                User_borrow_USDC[user_address]=(User_borrow_USDC[user_address]*110)/100;
            }
        }
        i+=1;
    }
    return true;
}
```

Fig. 4 Program function for interest calculation

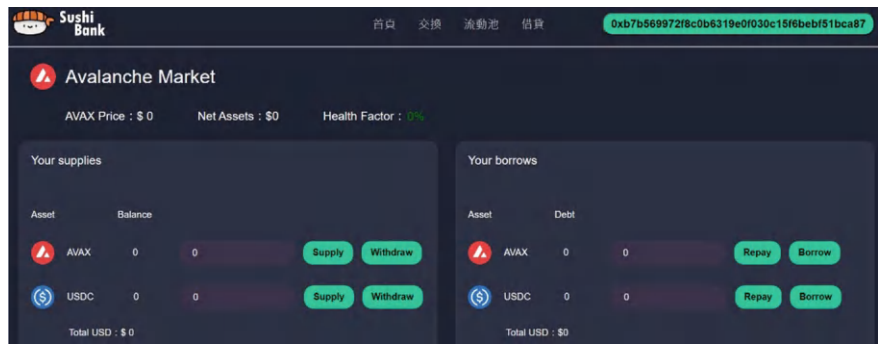


Fig. 5 Front-end implementation interface

other things. The above status information can be retrieved from the blockchain in real-time through Web3.js, achieving live updates. Interaction with the blockchain is also carried out in a similar fashion through Web3.js, ensuring sufficient functionality and information.

4.2 Smart Contract and Lending Development

In the blockchain financial lending system, we use three different addresses to simulate the protocol liquidation process. These addresses represent the fund provider, liquidator, and borrower as follows:

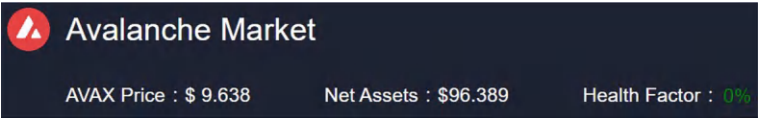


Fig. 6 Dashboard of borrower

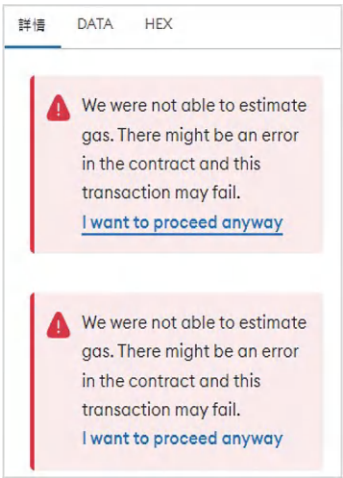
- **Fund Provider:** 0xF228C3EEcb4A3B0FDa7be846a9eBF4e260a891A2
- **Liquidator:** 0x 5b938B7EB9232a09b358568229cc9336CeAA7A6E
- **Borrower:** 0xB7B569972F8C0b6319E0F030c15F6bEBf51Bca87.

Firstly, to meet the conditions of the protocol (Eqs. (2) and (3)), we can choose alternative criteria. The fund provider supplies 500 USDC, the liquidator supplies 1000 USDC, and the borrower uses 10 AVAX as collateral. Based on the current AVAX price provided by ChainLink Oracle, the borrower can see relevant information on the protocol dashboard, as shown in Fig. 6.

According to the over-collateralization principle, the maximum borrowing limit is 80% of the collateral value. When calculated in USDC, it is approximately 77 USDC. However, when the borrower attempts to borrow 77 USDC, MetaMask may show a transaction failure, as shown in Fig. 7. If the borrowing amount is changed to 76 USDC, the transaction can be successfully submitted. Due to the lack of support for floating-point numbers in the Solidity language, precise calculations are not possible. A common solution is to multiply all values by a fixed multiplier to convert the decimal part into an integer before performing calculations. For example, if precision to two decimal places is needed, all values can be multiplied by 100 before calculation. This method helps improve calculation accuracy.

After lending 76 USDC, and a certain period of time elapses, if the interest rate continues to rise and the “Health Factor” reaches 90%, a liquidation will be triggered

Fig. 7 Transaction failure notification



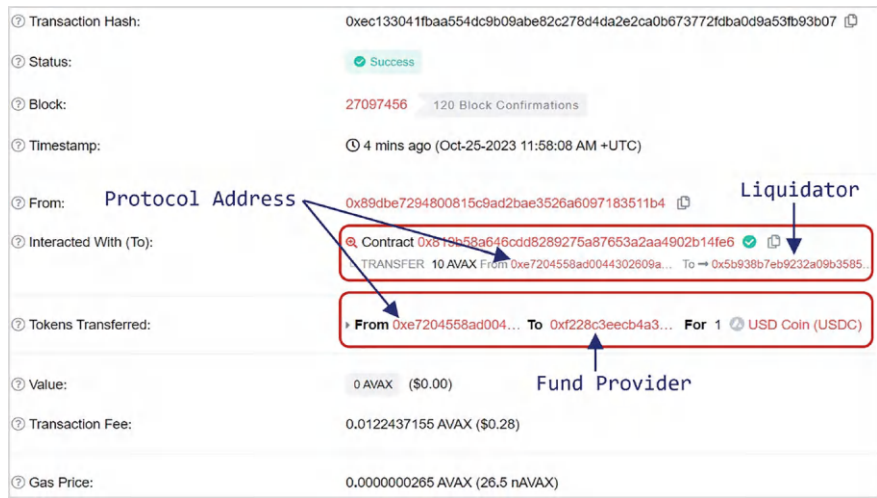


Fig. 8 Transaction details of liquidation execution

and executed. Figure 8 displays the transaction details viewable on the blockchain. It’s evident that the borrower’s collateral is entirely transferred to the liquidator. Within the protocol, a deduction is simultaneously made for 77 USDC from the liquidator (this detail is not displayed), and 1 USDC is transferred to the fund provider, which is considered as profit. Due to the previously mentioned decimal point issue, the decimal part is discarded. If the transfer amount is less than 1 USDC, the transfer will not take place, and this aspect requires optimization.

Through practical operational testing, the functionality of lending with excess collateral has been achieved. It involves transplanting traditional financial models onto the blockchain and implementing them through smart contracts. This empowers users to obtain funds without selling cryptocurrencies, allowing them to freely utilize the funds. While blockchain financial lending carries some risks, this model significantly enhances the liquidity of funds on the blockchain.

5 Conclusions

The financial lending agreements play a pivotal role in blockchain technology, offering a novel approach to financial services grounded in principles of openness and decentralization. These agreements extensively leverage smart contracts and blockchain technology, enabling borrowers and lenders to engage in direct transactions without relying on traditional financial intermediaries. This innovative

model positively impacts financial inclusivity and global financial market accessibility. However, it is crucial to address the risks associated with lending agreements, including smart contract vulnerabilities and market liquidity issues. Therefore, ensuring the safety and trustworthiness of both users and agreements becomes paramount.

Looking ahead, the digitization of real-world assets (RWAs) becomes a critical topic as centralization and decentralization converge. These physical assets represent tangible properties and interests in the real world, such as real estate, bonds, and equities. Transforming these tangible assets into digital form and conducting transactions on decentralized financial (DeFi) platforms brings about financial innovation and broadens investment and financing channels. This transformation holds particular significance in the DeFi sector. However, achieving the digitization and transactions of RWAs requires compliance and regulatory considerations to ensure adherence to relevant laws and regulations. Simultaneously, it presents opportunities and challenges for the financial market and traditional financial systems. This development expands asset diversity and practicality in the financial sector, driving widespread adoption of blockchain technology in traditional finance.

Acknowledgements This research was funded by the Ministry of Science and Technology, Taiwan, under Grant MOST 110-2221-E-992-097-MY3.

References

1. Nakamoto S (2008) Bitcoin: a peer-to-peer electronic cash system. Decentralized business review
2. Gramoli V (2020) From blockchain consensus back to Byzantine consensus. *Futur Gener Comput Syst* 107:760–769
3. Lee WM, Lee WM (2019) Using the metamask chrome extension. *Begin Ethereum Smart Contract Program: Ex Python, Solidity, JavaScript* 93–126
4. Zheng Z, Xie S, Dai HN, Chen X, Wang H (2018) Blockchain challenges and opportunities: a survey. *Int J Web Grid Serv* 14(4):352–375
5. Krawiec RJ, White M (2023) Blockchain: opportunities for health care. Deloitte
6. Szabo N (1996) Smart contracts: building blocks for digital markets. *EXTROPY: J Transhumanist Thought* (16), 18(2):28
7. Popper N (2016) A hacking of more than \$50 million dashes hopes in the world of virtual currency. *N Y Times* 17
8. Zheng Z, Xie S, Dai HN, Chen W, Chen X, Weng J, Imran M (2020) An overview on smart contracts: challenges, advances and platforms. *Futur Gener Comput Syst* 105:475–491
9. Werner S, Perez D, Gudgeon L, Klages-Mundt A, Harz D, Knottenbelt W (2022) Sok: decentralized finance (defi). In: *Proceedings of the 4th ACM conference on advances in financial technologies*. pp 30–46
10. Wang Z, Jin H, Dai W, Choo KKR, Zou D (2021) Ethereum smart contract security research: survey and future research opportunities. *Front Comp Sci* 15:1–18
11. Sekniqi K, Laine D, Buttolph S, Sirer EG (2020) Avalanche platform. *Netw Distrib Ledgers*
12. Ellis S, Juels A, Nazarov S (2017) Chainlink: a decentralized oracle network. 11(2018):1
13. Breidenbach L, Cachin C, Chan B, Coventry A, Ellis S, Juels A, Zhang F (2021) Chainlink 2.0: next steps in the evolution of decentralized oracle networks. *Chainlink Labs* 1:1–136

14. Egberts A (2017) The oracle problem-an analysis of how blockchain oracles undermine the advantages of decentralized ledger systems. SSRN 3382343.
15. Protocol AAVE (2020) AAVE protocol whitepaper. https://github.com/aave/aaveprotocol/blob/master/docs/Aave_Protocol_Whitepaper_v1_0.pdf [Stand 11.02. 2020]
16. Axie Infinity. Axie infinity: battle, collect, and trade collectible NFT creatures (2021). <https://axieinfinity.com/>
17. OpenLaw: Real world contracts for Ethereum. <https://www.openlaw.io/>
18. Panda SK, Satapathy SC (2021) An investigation into smart contract deployment on Ethereum platform using Web3.js and solidity using blockchain. In: Data engineering and intelligent computing: proceedings of ICICC 2020. Springer Singapore, pp 549–561