SpringerBriefs in Applied Sciences and Technology
PoliMI SpringerBriefs

Barbara Pernici · Tiziana Catarci ·
Matteo Palmonari · Giovanni Simonini *Editors*

Discount Quality for Responsible Data Science

Human-in-the-Loop for Quality Data



POLITECNICO
MILANO 1863

OPEN ACCESS



SpringerBriefs in Applied Sciences and Technology

PoliMI SpringerBriefs

Series Editors

Barbara Pernici, DEIB, Politecnico di Milano, Milano, Italy

Stefano Della Torre, DABC, Politecnico di Milano, Milano, Italy

Bianca M. Colosimo, DMEC, Politecnico di Milano, Milano, Italy

Tiziano Faravelli, DCHEM, Politecnico di Milano, Milano, Italy

Roberto Paolucci, DICA, Politecnico di Milano, Milano, Italy

Silvia Piardi, Design, Politecnico di Milano, Milano, Italy

Gabriele Pasqui, DASTU, Politecnico di Milano, Milano, Italy

Springer, in cooperation with Politecnico di Milano, publishes the PoliMI SpringerBriefs, concise summaries of cutting-edge research and practical applications across a wide spectrum of fields. Featuring compact volumes of 50 to 125 (150 as a maximum) pages, the series covers a range of contents from professional to academic in the following research areas carried out at Politecnico:

- Aerospace Engineering
- Bioengineering
- Electrical Engineering
- Energy and Nuclear Science and Technology
- Environmental and Infrastructure Engineering
- Industrial Chemistry and Chemical Engineering
- Information Technology
- Management, Economics and Industrial Engineering
- Materials Engineering
- Mathematical Models and Methods in Engineering
- Mechanical Engineering
- Structural Seismic and Geotechnical Engineering
- Built Environment and Construction Engineering
- Physics
- Design and Technologies
- Urban Planning, Design, and Policy

Barbara Pernici · Tiziana Catarci ·
Matteo Palmonari · Giovanni Simonini
Editors

Discount Quality for Responsible Data Science

Human-in-the-Loop for Quality Data

Springer

Editors
Barbara Pernici
DEIB
Politecnico di Milano
Milan, Italy

Tiziana Catarci
DIAG
Sapienza Università di Roma
Rome, Italy

Matteo Palmonari
DISCo
University of Milano-Bicocca
Milan, Italy

Giovanni Simonini
Dipartimento di Ingegneria Enzo Ferrari
University of Modena and Reggio Emilia
Modena, Italy



ISSN 2191-530X ISSN 2191-5318 (electronic)
SpringerBriefs in Applied Sciences and Technology
ISSN 2282-2577 ISSN 2282-2585 (electronic)
PoliMI SpringerBriefs
ISBN 978-3-032-29480-7 ISBN 978-3-032-29481-4 (eBook)
https://doi.org/10.1007/978-3-032-29481-4

This work was supported by Politecnico di Milano.

© The Editor(s) (if applicable) and The Author(s) 2027. This book is an open access publication.

Open Access This book is licensed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (http://creativecommons.org/licenses/by-nc-nd/4.0/), which permits any noncommercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if you modified the licensed material. You do not have permission under this license to share adapted material derived from this book or parts of it.
The images or other third party material in this book are included in the book's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the book's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.
This work is subject to copyright. All commercial rights are reserved by the author(s), 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. Regarding these commercial rights a non-exclusive license has been granted to the publisher. Unless otherwise expressly agreed by the Publisher in writing, the Publisher does not authorize use or reproduction of all or any part of this work in any manner for the purpose of training, fine-tuning, evaluating, or developing artificial intelligence models, technologies, or systems. In accordance with Article 4(3) of the EU Directive 2019/790 on copyright and related rights in the Digital Single Market, the Publisher expressly reserves all rights with respect to this work under the text and data mining exception of the directive.
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 Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

If disposing of this product, please recycle the paper.

Preface

Data preparation is a prerequisite in modern applications, in particular data analysis projects, artificial intelligence, and machine learning. The quality of data is crucial for achieving the desired results; however, due to the high volumes and the diversity of data sources, the data preparation activities that are needed to reach adequate quality levels may easily become unsustainable. Moreover, some preparation efforts may become unnecessary, if they result in negligible improvements or duplicate actions already performed in other contexts.

In this book, we propose a "discount" quality approach for scientific data analysis, discussing advanced techniques for pursuing responsible data science, i.e., to design sustainable data analysis pipelines based on Human-in-the-Loop (HITL) approaches for achieving quality data. The main results focus on developing task- and context-driven sustainable HITL approaches for data preparation, data quality analysis, and data visualization, based on methods and theories to reduce the annotations and the processing space and time, which starts from the estimation of the needed human computing effort evaluation and the requirements of the specific task.

The presented research focuses on the following topics:

- *Data ecosystems*: identifying reference models and architectures to specify and describe data and pipelines in a more transparent and interpretable manner, so that they can be better found and understood by third-party consumers (humans and machines, e.g., data search engines) and eventually reused, as recommended by several EU initiatives to foster both the data economy and Open Science. Data publishers must be supported with algorithms and tools that substantially reduce the effort they can afford to dedicate to a tedious and low-priority task.
- *Data preparation*: studying all aspects related to human intervention in data preparation pipelines. This includes several activities, and in particular, we focus our research on the management of scarce resources and user motivation in repetitive tasks such as labeling, data enrichment, and cleaning. Human involvement in these activities must be systematically evaluated to minimize effort while preserving the quality and significance of the outcomes.

- *Data quality evaluation*: evaluating the suitability of data to the task at hand in different stages of the data preparation pipelines (i.e., before and after data preprocessing, after data analysis) to evaluate the uncertainty of the results; classify pipeline improvements, toward methods for suggesting and minimizing human efforts in data preparation and profiling; providing a data quality assurance process for scientific data characterized by imprecise values and structures.
- *Information Visualization and Visual Analytics*: providing a flexible human-in the-loop system that, in a data preparation cycle, identifies potential performance, data quality, and critical issues, guided by the user, resolves them to generate a new model and continues this cycle until a satisfactory model is achieved. The visual analytics layer balances the trade-off between result accuracy and interactivity by minimizing latency, ensuring a smooth exploratory data analysis experience through predictive techniques for anticipating user interactions and progressive visualizations. These combined approaches enable a more efficient workflow, allowing analysts to explore data ahead of simulation completion, confirming promising runs and discarding unproductive ones, thereby conserving computational resources. Predicting user interactions also facilitates optimal resource allocation, focusing computational power on areas where the user prioritizes accuracy for critical decision-making.

The research presented in this volume is based on the results of the project "Discount quality for responsible data science: Human-in-the-Loop for quality data," funded by the Italian Ministry of University and Research under the PRIN Programme, a collaborative project coordinated by the RAISE (Research on Advanced Information Systems Engineering) group of Politecnico di Milano, Department of Electronics, Information, and Bioengineering, together with the DatAI Lab—Data for Artificial Intelligence of University of Milano-Bicocca, the DBGroup of University of Modena and Reggio Emilia, and the group of Sapienza Università di Roma.

The aim of the book is to present the main issues and discuss the challenges of developing sustainable approaches to data preparation for researchers and companies developing systems in the data-centric AI sector.

Milan, Italy
April 2026

Barbara Pernici

Acknowledgements

The project “Discount quality for responsible data science: Human-in-the-Loop for quality data” was funded by the Italian Ministry of University and Research under the PRIN Programme (Programma Nazionale di Ricerca e Progetti di Rilevante Interesse Nazionale), within the Italia Domani programme funded by the NextGenerationEU European Programme.

The authors thank and acknowledge the contribution of all participants in the project.

Contents

1 Sustainable Data Preparation: An Introduction 1
Barbara Pernici

2 Data-Centric AI: The Role of Data Quality Dimensions and Dataset Characteristics 11
Camilla Sancricca and Cinzia Cappiello

3 Don't Waste Resources on Data Preparation (Do it On-Demand) 23
Sonia Bergamaschi, Giovanni Simonini, Luca Zecchini, Angelo Mozzillo, and Giovanni Malaguti

4 Reconciliation-Based Data Enrichment 39
Flavio De Paoli, Matteo Palmonari, and Federico Belotti

5 Lightweight Pipelines: Good Enough is Sometimes Better 61
Camilla Sancricca and Cinzia Cappiello

6 Human Factors in Data Preparation 71
Marco Angelini, Matteo Filosa, and Tiziana Catarci

7 Key Insights and Future Opportunities for Sustainable Data Preparation 97
Barbara Pernici and Sonia Bergamaschi

Chapter 1
Sustainable Data Preparation: An Introduction



Barbara Pernici

Abstract The concept of sustainability of data preparation is discussed under different perspectives. In particular, we focus on data preparation activities for data-centric AI and the evaluation of sustainability, considering different quality dimensions. An analysis framework based on the circular economy paradigm is presented and discussed.

1.1 An Introduction to Sustainability

As discussed in [10], sustainability can be analyzed under many different points of view, and its definition, and in particular its evaluation metrics, have an important impact on the possible paths to be taken to improve a system based on the analysis of large amounts of data.

Three main perspectives are usually considered when defining sustainability: environmental, economic, and social. These perspectives form the basis for defining the goals of sustainability, for identifying the relevant assessment indicators, and for measuring the results of sustainability initiatives. The United Nations defined 17 Sustainable Development Goals (SDGs) as "a universal call to action to end poverty, protect the planet, and ensure that by 2030 all people enjoy peace and prosperity".[1]

From an *environmental perspective*, digital transformation has shown potential for significant positive contributions. For example, it can aid in reducing greenhouse gas emissions and resource consumption (Goal 6 Clean water and sanitation and Goal 7 Affordable and clean energy), mitigating climate change (Goal 13 Climate action and Goal 1 No Poverty), and protecting ecosystems and biodiversity (Goal 14 Life below water and 15 Life on land). Digital solutions such as smart grids, precision agriculture, and energy-efficient algorithms play critical roles in these areas.

[1] https://sdgs.un.org/goals.

B. Pernici (✉)
Politecnico di Milano - DEIB, Milano, Italy
e-mail: barbara.pernici@polimi.it

© The Author(s) 2027

B. Pernici et al. (eds.), *Discount Quality for Responsible Data Science*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-032-29481-4_1

However, the negative impacts of digital technologies cannot be overlooked. In particular, in recent years, generative Artificial Intelligence (genAI) platforms and other emerging innovations have often required substantial energy and natural resources, contributing to increased greenhouse gas emissions. Sustainable computing has become a relevant issue in the scientific debate [5]. The fast development pace of AI applications in many sectors is having a significant impact on the development of new energy systems [7]. Models for assessing the impact of genAI systems are being developed, considering also Life Cycle Assessment (LCA) as a basis for evaluating sustainability [2]. The concept of AI sustainability [15] is emerging in the literature, focusing in particular on its environmental sustainability. The short lifecycle of many digital devices exacerbates the issue of electronic waste, creating a mounting environmental challenge. Furthermore, the reliance on rare earth materials for these technologies raises ethical and environmental concerns. Mining operations in the Global South frequently occur under hazardous and inhumane conditions, leading to contaminated ecosystems, deforestation, and air pollution. These activities not only harm the environment but also perpetuate social injustices.

On the *social dimension*, digital technologies hold promises for driving transformative change. They can enhance healthcare in disadvantaged areas (Goal 3), foster education and social inclusion in underprivileged communities (Goal 1 No poverty, Goal 2 Zero hunger, Goal 5 Gender equality, and Goal 16 Peace, justice and strong institutions), and contribute to reducing food waste through better supply chain management. Additionally, digital innovations enable the transformation of economic systems by promoting sustainable business models and practices (Goal 8 Decent work and economic growth, Goal 9 Industry, Innovation, Technology and Infrastructure, and Goal 12 Responsible consumption and production). However, the social implications of digital technologies are multifaceted. Issues related to privacy, security, and human rights require careful consideration. The massive use of personal data by AI systems risks reinforcing inequalities and discriminatory practices.

Economic considerations are equally critical in the discourse on digital sustainability. Digital technologies can drive productivity, foster innovation, and create opportunities for sustainable economic growth. For example, AI and blockchain technologies are being leveraged to enhance transparency in supply chains and promote circular economy practices, e.g., reducing waste, extending the life cycle of products, and optimizing resource efficiency. However, the economic benefits must be weighed against the long-term costs associated with resource extraction, energy consumption, and waste management. Policymakers and organizations must adopt a holistic approach to ensure that economic growth does not come at the expense of environmental degradation or social inequities.

In conclusion, the integration of digital technologies into sustainability frameworks presents both opportunities and challenges. A comprehensive and inclusive approach is necessary to harness the full potential of digital tools while mitigating their adverse effects. Stakeholders must collaborate to establish guidelines and standards for digital sustainability, ensuring that technological innovation aligns with the broader goals of environmental preservation, social equity, and economic resilience. As the digital revolution continues to evolve, it is imperative to prioritize sustainable

practices that benefit all dimensions of society and the planet. Digital sustainability is particularly critical when considering Machine Learning (ML) and AI.

Electricity consumption trends show a fast-growing percentage of use of electricity for IT and data centers[2] and projections show a considerable increase in the coming years.[3,4] Analogous considerations have been made for water consumption since large AI models utilize a significant amount of water in order to cool the data centers [8, 18].

Machine Learning model repositories are being populated rapidly with new models; however, measuring and reporting their carbon footprint is still in the initial phase, lacking systematic approaches [4, 9]. Green IT, Green software [3] and, in particular, Green AI [14] and Sustainable AI [15] approaches are emerging, mainly focusing on energy efficiency and environmental impacts. Data preparation plays a central role in this context, as the growing demand for high-quality data to train models drives the emergence of data-centric AI (DCAI). DCAI emphasizes improving model performance and sustainability by optimizing data quality rather than focusing on model complexity and optimization, with the assumption that high-quality data can yield better results than collecting big data, improving the data quality and the model in parallel, in a series of iterations [6]. The emerging field of data-centric green AI [13, 16] focuses both on DCAI and Green AI, studying the energy impact of dataset modifications, and analyzing also the trade-offs between energy consumption and the model performance when the size of the dataset is reduced.

1.2 Data Preparation

Data preparation is central in all data analysis tasks and in particular in data-centric green AI, due to the growing demand for good quality data for training models. Sustainability issues arise in data preparation, considering the process of information production, pipelines to prepare and process data, and efforts in data curation.

While pre-processing input data to improve data quality, we consider the following **data preparation tasks** [11]:

1. *data profiling* tasks, to analyze the characteristics of the data;
2. *data transformation* tasks (including data cleaning, normalization and standardization, merging, splitting, dropping data, data imputation);
3. *data matching* tasks at the instance and schema level (including deduplication, entity matching and/or linking, annotations of columns and column pairs), and
4. *data augmentation* tasks to extend data with data from third-party sources.

[2] International Energy Agency, Report Electricity, 2024 https://www.iea.org, visited on 10 June 2025.

[3] EPRI Electric Power Research Institute White paper on Powering intelligence, 2024 https://www.epri.com/research/products/000000003002028905, visited on 10 June 2025.

[4] Amodei, D., Hernandez, D., Sastry, G., Clark, J., Brockman, G., Sutskever, I. (2018). AI and Compute. https://openai.com/index/ai-and-compute/, visited on 10 June 2025.

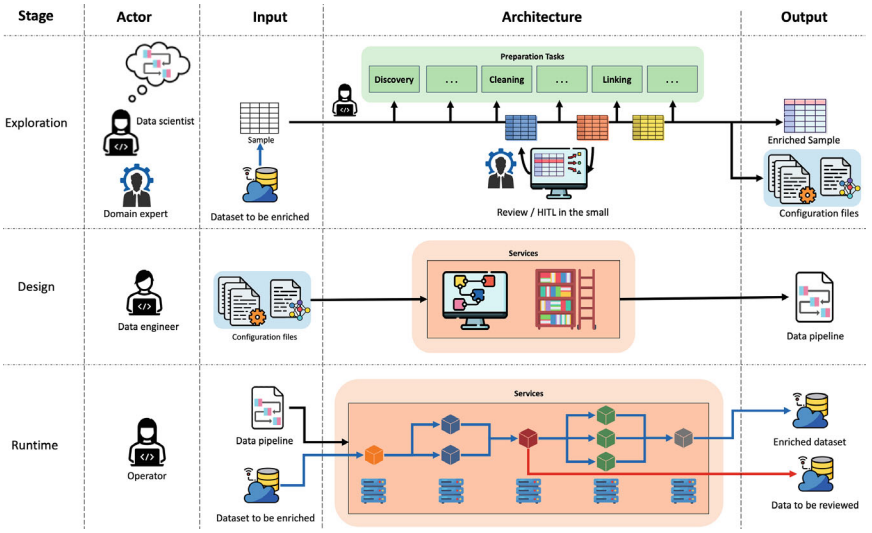


Fig. 1.1 Exploration and data preparation pipelines (*Source* [11])

A pipeline can be interpreted as a sequence or a more complex workflow of operations on the data. While in many application scenarios pipelines must be executed by data engineers on large data sets, this large-scale execution is just a final step of a more complex task, which, inspired by requirements collected for data enrichment pipelines, we conceptualize as composed of three phases (see Fig. 1.1):

1. In the *exploration phase*, the goal is to (1) understand the fitness for usage in downstream tasks and (2) identify operations that can improve their quality based on the intended usage. Typical actors involved in this phase are *data scientists* and professional figures with domain expertise (*domain experts*).
2. This phase is followed by the phase of *design* of the engineered pipeline, typically performed by *data engineers*.
3. The third phase consists of the actual *execution* (*runtime*), which also involves monitoring by operators.

Concerning the tasks in the exploratory phase, examples of questions to be answered are: what are the characteristics of the data? How can the data be transformed? Is it possible to enrich the data via integration with other data (matching and augmentation)? In this phase, users typically consider data samples and need some direct feedback on the results of the operations they explore to understand their effect, making the usage of proper interfaces very valuable. The output of this phase is the definition of a preparation pipeline and the specification of key elements, such as configurations for specific algorithms used within it. Suppose the pipeline needs to be applied to a large amount of data and/or replicated recurrently. In that case, it must usually be engineered to be efficient and, therefore, compliant with big data processing platforms (e.g., using distributed computation), which is the objective of

the design phase. While the output of the design phase is the specification of the engineered pipeline, the output of the execution phase is the cleaned and enriched data.

Within these phases, the main challenges concern the increasing need for computational, human, and energetic resources to be used in the process, particularly for AI and ML applications. Social implications are also increasingly arising when collecting and analyzing data. In particular, increasing demand for data should deal with social issues introduced by bias, lack of understanding, and coverage of all needed aspects in data being used for multiple purposes.

As a consequence, comprehensive approaches are needed to consider sustainability in data preparation, adding to efficiency-related considerations available in the state of the art, further considerations focusing not only on using resources efficiently for data preparation, with the goals of reducing time, computational effort, needed human resources, and storage, but also considering sustainability more in general, including also economic and social aspects relevant to the data preparation phases.

1.3 An Approach to Sustainable Data Preparation

For many years, it has been recognized that the preparation of information products has many characteristics in common with production processes [1].

Circular economy has been proposed as a way to achieve sustainable development in production processes in companies [17], therefore, the sustainability strategies presented in other production contexts appear as an interesting paradigm that we propose to adopt also in the field of data preparation.

In [12], nine strategies for circular economy are described, grouped in three Categories: (i) Smarter product use and manufacture (Design phase); (ii) Extend lifespan of product and its parts (Consumption phase); (iii) Useful application of materials. In Table 1.1, we present the original definitions by Potting in [12] and we introduce possible actions to be implemented to improve data quality during data preparation, considering the Design and the Consumption phases. Among the proposed strategies for other sectors, we have selected strategies R1–R5 as relevant for discussing approaches to make the data preparation phase more sustainable.

In particular, among the *Strategies for smarter product use and manufacture* in the Design phase, **R1 Rethink** aims to design systems that can be used more intensively and durably, and can be adopted both for facilitating the sharing of data products at a large scale and for adopting data sharing solutions that can perform multiple tasks with one model (AI); **R2 Reduce** calls for increasing efficiency in product manufacture or use by consuming fewer natural resources and materials. Considering data preparation within this strategy, the main goal is to create methods and techniques to make data preparation more efficient.

Considering *Strategies for extending the lifespan of products* in the Consumption phase, **R3 Reuse** by another consumer of a discarded product that is still in good condition can be applied in the field of data preparation either to a data source or

Table 1.1 Sustainability strategies in data preparation (adapted and extended from [12])

Category	Strategy	Description	Data preparation activities
Smarter product use and manufacture (Design phase)	R0 Refuse	Avoid unnecessary products or resource use by refusing non-essential consumption	Not relevant
	R1 Rethink	Make product use more intensive via new concepts or business models	Data spaces; Models to perform multiple tasks—LLMs
	R2 Reduce	Increase efficiency by consuming fewer resources in manufacture or use	On demand; Selecting relevant data/tables; Designing efficient pipelines
Extend lifespan of product and its parts (Consumption phase)	R3 Reuse	Reuse by another consumer of a discarded product that still fulfills its original function	Reuse curated datasets; Consider context; Enrich metadata
	R4 Repair	Repair and maintenance of a defective product so it can be used with its original function	Fix data errors; impute missing values; Inspection and pipeline design; Considering additional sources
	R5 Refurbish	Restore an old product and bring it up to date.	Extend the lifecycle of old datasets
	R6 Remanufacture	Use parts of discarded products to make a new product with the same function	Not considered
	R7 Repurpose	Use a discarded product or its parts in a new product with a different function	Not considered
Useful application of materials	R8 Recycle	Process materials to obtain the same or lower-quality raw material	Not relevant
	R9 Recover	Incinerate materials with energy recovery	Not relevant

a sequence of already implemented actions (a pipeline). Reuse involves the efforts needed to create accurate descriptions of (meta)data to facilitate reuse or to make data accessible using paradigms that improve semantic interoperability, and it strongly depends on the context of reuse; **R4 Repair** focuses on maintaining defective products so that they can be used with their original function. In data preparation, it is relevant when cleaning data with quality issues; **R5 Refurbish** refers to restoring and bringing an old product up to date. In the context of data preparation, we consider extending the data source with new data that has become available over time.

To assess and compare alternative techniques for each of these phases, we have proposed an assessment model that is based on the following evaluations:

- *Data quality improvement*. We consider the improvement of the quality of data evaluated along traditional data quality dimensions, quality dimensions in data-centric AI, and Visual analytics dimensions.
- *Cost evaluation*. Costs are evaluated considering both computational and human resources involved in the application of a given technique. We distinguish between *Assessment cost* for the data profiling and evaluation of the quality dimensions to be considered and *Improvement cost* for applying the data cleaning and transformation techniques.
- *Waste evaluation*. Waste evaluation is proposed, as each data improvement technique needs to be evaluated also in terms of the percentage of data being prepared that will be actually used and of the success rate of the technique in solving a given quality issue in a given context, considering resources used for preparing data which will not be used a waste and also unsuccessful repairs another type of waste.

Detailed technical details on these models and the associated evaluation formulas can be found in [10].

1.4 Evaluation of Data Preparation Methods Within a "Discount Quality" Approach

The proposed framework can be applied to assess the sustainability and related trade-offs of a selection of data preparation methods, chosen to cover all five strategies. Within the "Discount quality" project, we compared different methods for data preparation according to the framework described in the previous section, with the dual goal of improving data quality and of reducing waste within the process, taking also efficiency considerations into account.

In the following chapters of this book, first, we illustrate the quality dimensions in Chap. 2, with a focus on those relevant to data-centric AI and the impact of quality problems on the results. In the following chapters, we analyze different approaches that can be adopted for sustainable data preparation for the different strategies, as synthesized in Table 1.2.

Table 1.2 Methods for data preparation and their applicability to sustainability strategies

Methods	Strategies	Chapter
On demand approaches	Reduce, Reuse, Refurbish	3
Reconciliation-based data enrichment	Rethink, Reduce, Reuse, Repair	4
Pipeline design	Reduce, Reuse, Repair	5
Human Factors	All strategies	6

As shown in the table, four method categories are considered and are discussed in the next chapters. Each method category includes a set of methods that focus on a specific data preparation task, such as entity reconciliation or pipeline design, or an approach, such as on-demand approaches and considering human factors during visualization. Methods in each category can benefit several strategies. All methods can serve the purposes of Reduce and Reuse strategies, due to their general focus on developing a sustainable approach. Within on-demand approaches, there is a focus on identifying suitable data sources in order to Refurbish data. The other three categories can also support the Repair strategy. A detailed description of the proposed methods is provided in the following chapters, as indicated in the table.

References

1. Ballou, D.P., Tayi, G.K.: Methodology for allocating resources for data quality enhancement. Commun. ACM **32**(3), 320–329 (1989)
2. Berthelot, A., Caron, E., Jay, M., Lefevre, L.: Understanding the environmental impact of generative AI services. Commun. ACM **68**(7), 46–53 (2025)
3. Caballar, R.D.: We need to decarbonize software: the way we write software has unappreciated environmental impacts. IEEE Spectr. **61**(4), 26–31 (2024)
4. Castaño, J., Martínez-Fernández, S., Franch, X., Bogner, J.: Exploring the carbon footprint of Hugging Face's ML models: a repository mining study. In: 2023 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), pp. 1–12. IEEE, Piscataway, NJ (2023)
5. Chien, A.A., Schmeck, H.: Sustainability and computing - edited Special Section. Commun. ACM **68**(7), 43–107 (2025)
6. Jakubik, J., Vössing, M., Kühl, N., Walk, J., Satzger, G.: Data-centric artificial intelligence. Bus. Inform. Syst. Eng. **66**, 507–515 (2024)
7. Jin, D., Ocone, R., Jiao, K., Xuan, J.: Energy and AI **1**, 100002 (2020)
8. Li, P., Yang, J., Islam, M.A., Ren, S.: Making AI less "thirsty"' - uncovering and addressing the secret water footprint of AI models. Commun. ACM **68**(7), 54–61 (2025)
9. Luccioni, A.S., Viguier, S., Ligozat, A.L.: Estimating the carbon footprint of BLOOM, a 176B parameter language model. J. Mach. Learn. Res. **24**(253), 1–15 (2023)
10. Pernici, B., Cappiello, C., Bono, C., Sancricca, C., Catarci, T., Angelini, M., Filosa, M., Palmonari, M., Paoli, F.D., Bergamaschi, S., Simonini, G., Mozzillo, A., Zecchini, L.: Sustainable quality in data preparation. ACM J. Data Inform. Qual. **17**(4:23), 1–33 (2025)
11. Pernici, B., Cappiello, C., Ramalli, E., Palmonari, M., Belotti, F., Paoli, F.D., Mozzillo, A., Zecchini, L., Simonini, G., Bergamaschi, S., Catarci, T., Filosa, M., Angelini, M., Benvenuti, D.:

The future of sustainable data preparation. In: Atzori, M., Ciaccia, P., Ceci, M., Mandreoli, F., Malerba, D., Sanguinetti, M., Pellicani, A., Motta F. (eds.) Proceedings of the 32nd Symposium of Advanced Database Systems. CEUR Workshop Proceedings, vol. 3741, pp. 486–497. CEUR-WS.org (2024)
12. Potting, J., Hekkert, M.P., Worrell, E., Hanemaaijer, A.: Circular economy: measuring innovation in the product chain. PBL Netherlands Environmental Assessment Agency, 2544 (2017)
13. Salehi, S., Schmeink, A.: Data-centric green artificial intelligence: a survey. IEEE Trans. Artif. Intell. **5**(5), 1973–1989 (2023)
14. Schwartz, R., Dodge, J., Smith, N.A., Etzioni, O.: Green AI. Commun. ACM **63**(12), 54–63 (2020)
15. Van Wynsberghe, A.: Sustainable AI: AI for sustainability and the sustainability of AI. AI Ethics **1**(3), 213–218 (2021)
16. Verdecchia, R., Cruz, L., Sallou, J., Lin, M., Wickenden, J., Hotellier, E.: Data-centric green AI: an exploratory empirical study. In: International Conference on ICT for Sustainability (ICT4S), pp. 35–45. IEEE, Piscataway, NJ (2022)
17. Walker, A.M., Opferkuch, K., Roos Lindgreen, E., Raggi, A., Simboli, A., Vermeulen, W.J., Caeiro, S., Salomone, R.: What is the relation between circular economy and sustainability? Answers from frontrunner companies engaged with circular economy practices. Circ. Econ. Sustain. **2**(2), 731–758 (2022)
18. Zuccon, G., Scells, H., Zhuang, S.: Beyond CO_2 emissions: the overlooked impact of water consumption of information retrieval models. In: Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR '23, pp. 283–289. ACM, New York, NY, USA (2023)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (http://creativecommons.org/licenses/by-nc-nd/4.0/), which permits any noncommercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if you modified the licensed material. You do not have permission under this license to share adapted material derived from this chapter or parts of it.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Chapter 2
Data-Centric AI: The Role of Data Quality Dimensions and Dataset Characteristics



Camilla Sancricca and Cinzia Cappiello

Abstract Data-centric AI highlights the importance of high-quality input data in machine learning, as it is essential for achieving reliable and accurate results. To this purpose, traditional data quality assessment and improvement systems might help detect and address data errors, inconsistencies, or missing values. However, recent literature has demonstrated that other factors, besides standard data quality issues, could compromise the performance of machine-learning applications. These factors relate to the characteristics of the datasets under consideration, such as their structure and statistical and ethical aspects (e.g., potential biases or unfairness). This chapter presents the results of a literature survey and proposes a quality model to address these aspects in data-centric AI. Such a model includes all possible data characteristics that may undermine the execution of a machine-learning pipeline, along with their corresponding metrics.

2.1 Introduction: the Need for New Dimensions in Data-Centric AI

The field of Data Quality (DQ) has traditionally addressed issues such as outliers, inconsistencies, and missing values. More recent studies, however, have highlighted that additional data-related characteristics, such as dataset structure and the statistical distribution of attributes, can significantly influence the outcomes of Machine Learning (ML) analyses. In the next sections, we explore these characteristics and identify new metrics to capture previously unaddressed data characteristics for evaluating the suitability of datasets in ML pipelines [20].

As recent studies have highlighted, data characteristics that can compromise the reliability of ML-based systems and their outcomes, traditional DQ models (such as

C. Sancricca · C. Cappiello (✉)
Politecnico di Milano—DEIB, Milano, Italy
e-mail: cinzia.cappiello@polimi.it

C. Sancricca
e-mail: camilla.sancricca@polimi.it

© The Author(s) 2027

B. Pernici et al. (eds.), *Discount Quality for Responsible Data Science*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-032-29481-4_2

those proposed in [5, 6]) are no longer sufficient to comprehensively address these aspects.

This chapter introduces a novel quality model specifically designed to support the development of Data-Centric AI (DCAI) systems.

The proposed model targets tabular datasets and is the result of (a) a literature review aimed to identify data characteristics that influence the performance of ML models, and (b) a series of experiments conducted to empirically validate the impact of those characteristics [20].

The literature analysis focused on identifying contributions that examine how certain properties of data can affect the behavior and effectiveness of ML models (particularly those aspects not captured by conventional DQ dimensions, including features typically derived from data profiling). For each identified factor, we sought suitable metrics to quantify it.

The remainder of the chapter is structured as follows: Sect. 2.2 reviews the existing literature on data-centric approaches for identifying data properties relevant to ML pipelines and discusses the limitations of standard DQ models in this context. Section 2.3 presents the proposed quality model in detail, while Sects. 2.4 and 2.5 report the experiments conducted to evaluate its effectiveness.

2.2 State-Of-The-Art Contributions on Novel Dimensions

This section provides a comprehensive overview of related work focusing on data characteristics and their influence on the performance of ML algorithms, highlighting how factors such as data quality, distribution, dimensionality, and imbalance can significantly affect model behavior and outcomes (Table 2.1).

A high number of features (dimensionality) can lead to overfitting, especially when many attributes are irrelevant to the learning task. This not only reduces model accuracy but also increases training time and computational overhead. Additionally, categorical features (data types) often require appropriate encoding, which may introduce sparsity or affect feature relevance. These challenges are discussed in [9, 12], while [1, 3, 21] analyze the effect of dataset size, concluding that representativity is more critical than volume, while dimensionality may enhance generalization in certain scenarios.

Highly correlated or redundant features increase model complexity and risk of overfitting. The study in [4] evaluates the role of correlation and dimensionality in logistic regression and deep learning, showing that both are relatively robust to these issues. Similarly, [16] highlights that models such as support vector machines, naive Bayes, and random forests tend to be less sensitive to high dimensionality or noisy attributes.

Unbalanced distributions (either across the entire dataset or in the class labels) can bias the learning process and impact model performance [21]. This issue is analyzed in [23], which compares various strategies (pre-processing, algorithmic, hybrid) for handling imbalance. Ali et al. [2] focuses specifically on class imbalance in clas-

Table 2.1 Relevant contributions mapping with data characteristics [20]

	Dimensionality	Data types	Correlation	Feature relevance	Imbalance	Normalization	Homogeneity	Class Noise	Class Imbalance	Class Overlap
Ajiboye et al. 2015 [1]	✓									
Ali et al. 2019 [2]					✓					
Althnian et al. 2021 [3]	✓									
Bailly et al. 2022 [4]	✓		✓							
Chen et al. 2021 [8]						✓				
Emmert-Streib et al. 2020 [9]	✓	✓								
Gupta et al. 2019 [11]								✓		
Luca et al. 2022 [12]	✓	✓								
Ma et al. 2018 [13]	✓							✓		
Osisanwo et al. 2017 [16]		✓	✓	✓						
Patel et al. 2022 [17]			✓	✓			✓	✓	✓	✓
Patel et al. 2023 [18]								✓		✓
Sen et al. 2023 [21]	✓				✓					
Singh and B. Singh 2020 [22]						✓				
Vuttipittayamongkol et al. 2021 [23]					✓					✓

sification tasks. Data normalization is often employed to mitigate data imbalance; [22] investigates the impact of fourteen normalization techniques under different preprocessing strategies.

Errors or inconsistencies in the target class can significantly degrade model performance [11, 13] reviews the impact of both attribute and class noise, noting how mislabeled or overlapping instances decrease model generalization, particularly for algorithms that rely on class separability, such as support vector machines. Additionally, [23] explores the interplay between class imbalance and class overlap, and surveys methods to mitigate their impact.

2.3 A Quality Model for Data-Centric AI

From the contributions analyzed in the previous section, it emerges that, beyond traditional DQ issues, additional data-related aspects can directly influence the effectiveness of ML pipelines. The data characteristics introduced by our model are intended to (a) extend traditional DQ models (traditional models include standard DQ dimensions such as completeness, accuracy, consistency, and timeliness) and (b) provide quantifiable insights into further aspects of data that may either improve or affect

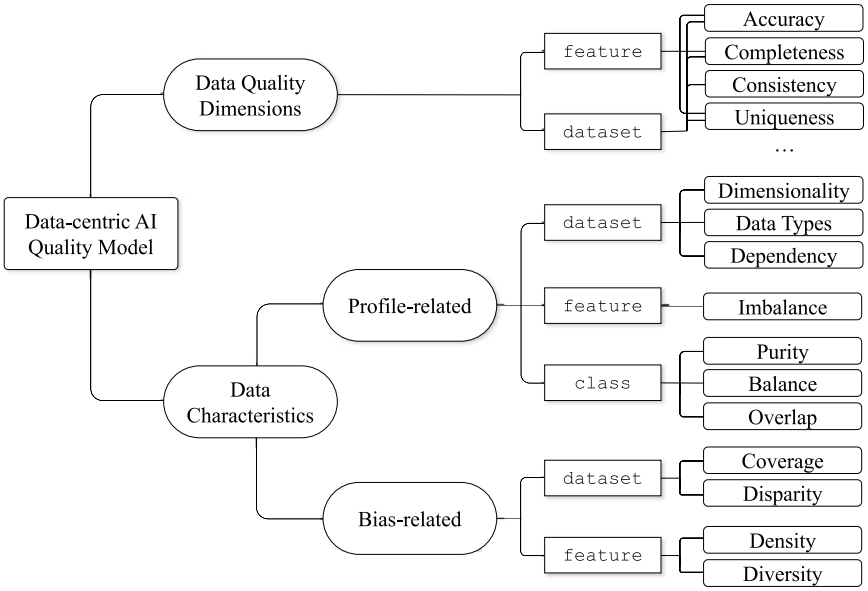


Fig. 2.1 New quality model for data-centric AI [20]

ML performance. Note that the metrics proposed for quantifying these aspects are already known in the literature, but they have typically been applied in different contexts.

In addition, the quality model incorporates a set of metrics for tabular data designed to detect bias early in the analysis pipeline, to help data scientists identify potential sources of unfairness and select datasets appropriate for sensitive decision-making contexts [20]. The proposed metrics are based on state-of-the-art contributions that discuss ethical concerns fundamental to ensuring proper DQ in sensitive domains [10, 19]. These aspects are related, for example, to unrepresentative data distributions, and are known to be primary sources of bias-related issues [15].

We organize the proposed data characteristics into two main categories: *Profile-related* and *Bias-related*. The former includes features that can be automatically extracted using standard data profiling techniques, while the latter encompasses ethical aspects of the data.

Figure 2.1 illustrates the DCAI quality model, including the granularity level at which each data characteristic is assessed: at the `dataset`, `feature`, or `class` level. It is important to note that characteristics defined at the class level are applicable only to classification tasks.

2.3.1 Profile-Related Dimensions

In this section, profile-related dimensions are defined. For each dimension, its impact on ML algorithms is discussed in the text boxes following the definition.

Dimensionality refers to the overall size of the dataset, defined by its number of rows (instances) and columns (features).

> *Overfitting.* As the number of *features* (i.e., columns) increases, the model's prediction ability can improve, potentially enhancing the accuracy but also resulting in overfitting.
> *Curse of dimensionality.* When analyzing data in a high-dimensional space, the model gives results that do not occur in low-dimensional settings. This phenomenon arises when an excessive number of non-informative features are introduced, leading to increased model complexity and a consequent decline in accuracy.
> *Computational effort.* Larger datasets or high-dimensional feature spaces require more computational costs and longer training time.
> *Generalization.* Larger datasets provide more representative samples of the underlying data distribution.

Data types quantify the distribution of feature types (numerical, categorical, and boolean) measured as their proportion over the total number of features.

> *Encoding.* Most ML models do not deal with categorical variables during training; choosing the proper encoding technique can be difficult. For example, choosing label encoding could affect the weights the model assigns to the features, while one-hot encoding could exaggerate the volume by increasing execution time and computational resources.
> *Distinctness.* Having many categorical variables may increase the possibility that some have a large number of distinct values. This can lead to *(i)* sparse feature representations, especially when they are one-hot encoded, and *(ii)* underrepresentation of rare categories, i.e., values with very few observations.

Dependency captures the extent of feature redundancy due to high correlation. It is assessed by computing correlation coefficients (e.g., Pearson, Spearman) and measuring the fraction of features that exceed a predefined threshold (e.g., Pearson > 0.7).

Computational effort. Redundant or highly dependent features can increase computational complexity, slowing the training time.
Overfitting. Redundant information can cause overfitting.
Interpretability. Redundant or highly dependent features can make it difficult to identify which features are most important and to understand the behavior of the model.
Multicollinearity. High dependency can worsen the problem of multicollinearity [7], where features treated as independent variables are highly correlated instead.

Imbalance evaluates the deviation of a feature's value distribution from a reference distribution, assumed here to be Gaussian. Several metrics are used to quantify this deviation:

- *Intersection Over Union*, which measures distributional overlap between the original distribution and an ideal Gaussian one.
- *Normalized Mean Square Error (NMSE)*, which estimates the difference between the original distribution and the Gaussian one, dividing their Mean Square Error (MSE) by the variance of the original distribution;
- *Wasserstein distance* between two probability distributions. It measures the effort required for transforming a distribution into another one:

$$Imbalance = \inf_{\gamma \sim \Pi(P,Q)} \mathbb{E}_{(x,y)\sim\gamma}[||x - y||]$$

where $\Pi(P, Q)$ is the set of all joint distributions mapping P into Q, and $\gamma(x, y)$ indicates the probability that is shifted from x to y to make x following the y distribution. inf is the greatest lower bound, indicating the goal of minimizing the transformation cost.

Class Purity estimates label correctness by quantifying label noise. We propose a k-means-based metric: the clustering algorithm is run excluding the class label, with k equal to the number of classes. A confusion matrix is then used to compute the fraction of correctly clustered instances.

Performance. Higher label purity generally leads to better model performance. Lower label purity can worsen performance, as incorrect labels may affect the prediction.

Class balance assesses the distribution of class labels, following [17]. Given the majority class and the minority classes, we define:

- *Absolute Class Balance* as the ratio between the size of the smallest and largest class.

- *Average Class Balance* computed as:

$$Class\ Balance = \frac{\frac{1}{|N|}\sum_{i \in N} n_i}{|M|}$$

where $|M|$ is the size of the majority class, and N is the set of minority class sizes.

> *Biased prediction.* Class imbalance can introduce bias; if present in the targeted label, predictions may be biased toward the majority class.

Class overlap measures how closely instances with different labels are positioned in feature space. We use a classifier (support vector machine or logistic regression) to train a decision function, then identify an overlapping region around the decision boundary using a threshold. The class overlap metric is defined as the ratio of points that fall within this region to the total number of instances.

> *Decision boundary.* High-class overlap can reduce the model's reliability as algorithms may struggle to find decision boundaries that separate the classes. *Misclassification*. High-class overlap can lead to higher misclassification. *Interpretability*. High-class overlap can affect the interpretability of ML models, making it difficult to detect relationships between features and classes.

2.3.2 Bias-Related Dimensions

Coverage represents the degree to which a dataset is representative of the real world. It was defined in a previous work as "*the proportion of entities represented in the dataset relative to the number of real-world entities*" [15]. It can be assessed as:

$$C = \frac{E}{RW}$$

where E is the number of entities represented in the dataset, and RW is the total size of the real-world entities. Note that the total size of the real-world entities is ground truth information and, if known, must be provided by the user.

Disparity assesses whether the label distribution differs across demographic groups. In this case, we considered the group fairness concept [14], which is commonly used in the literature to assess prediction fairness during the post-processing phase. However, in our model, we propose a metric for pre-processing that can be computed on the training dataset. Note that other fairness metrics can be used to estimate bias before the training phase. For a protected attribute (e.g., sex, age, etc.),

once the value of the protected group PG (e.g., female, younger than 25, etc.) has been established, it can be computed as:

$$DS = \frac{|P|}{|NP|}$$

where $P = \{x \mid y(x) = 1, x \in PG\}$ contains the samples x belonging to the protected group that have a favorable outcome in the training dataset and $NP = \{x \mid y(x) = 1, x \notin PG\}$ contains the samples x belonging to any group that have a favorable outcome.

Density is a measure of how densely concentrated certain values are within a feature; it was defined in a previous work as "*a measure of appropriate numerosity and intensity between different real-world entities available in the data*" [15]. It can be assessed as:

$$DN = \frac{OCC}{DIS}$$

where OCC is the number of occurrences of a specific value of a feature to the total number of distinct values DIS.

Diversity is a measure of the value heterogeneity within a feature; it was defined in a previous work as "*the degree to which different kinds of objects are represented in a dataset*" [10]. It is closely linked to the concept of entropy, which measures the amount of information contained within a dataset or a specific attribute. It can be evaluated using well-established metrics to quantify the information content, such as (1) Shannon's entropy and (2) the Gini index:

$$DV\ (1) = -\sum_{i=1}^{n} p(x_i) * log(p(x_i)) \qquad DV\ (2) = 1 - \sum_{i=1}^{n} (p(x_i))^2$$

where $p(x_i)$ is the relative frequency of the element i in a specific column.

Variability. In addition to its ethical implications, it is also relevant for performance as a higher diversity provides variability for models to learn from.

In particular, Coverage and Disparity can be computed at the dataset level, while metrics such as Density and Diversity can be used to estimate the presence of bias in a specific attribute of a table.

2.4 Impact of Profile-Related Dimensions on Machine-Learning Performance

This section describes the results we gathered from a set of experiments to validate (a) the effectiveness of the proposed metrics for quantifying the data characteristics of our model and (b) the impact of such characteristics on model performance [20].

Dimensionality Changes in the number of features had minimal impact on model performance, with the exception of k-nearest neighbors, whose accuracy declined as dimensionality increased. This behavior is expected due to the *curse of dimensionality*, which arises when data become increasingly sparse in high-dimensional spaces, making it more difficult for the model to generalize. In contrast, increasing the number of training instances led to improved generalization and overall performance.

Imbalance Greater imbalance consistently degraded performance, confirming that non-uniform distributions affect the model's ability to learn from underrepresented classes.

Data Types Model performance remained stable with few discretized features, but then declined. This can be attributed to the fact that continuous attributes retain more granular, discriminative information, which may be lost when they are discretized.

Dependency Increasing the number of highly correlated features had a negligible effect on model accuracy. This suggests that the presence of redundant information does not significantly affect learning in the tested scenarios.

Class Purity Both the validity of the proposed Purity metric and the positive correlation between Class Purity and model performance were confirmed.

Class Balance Class Balance is a key factor influencing model accuracy: balanced class distributions consistently lead to better results. The impact is particularly evident in low-dimensional datasets, such as heart attack, where a limited number of instances amplifies the effect of class imbalance on generalization.

Class Overlap A negative correlation between overlap and model performance was observed. The degradation is more significant in smaller datasets, where limited training data reduces the model's ability to generalize across ambiguous decision boundaries.

Discussion The experimental results confirm the effectiveness of the proposed metrics in capturing the presence of the data characteristics defined in our model. Specifically, the computed metric values on the perturbed datasets consistently reflected the degree of injected data pollution.

We further investigated how these characteristics affect the performance of various classification algorithms. The analysis revealed that characteristics such as Imbalance, class Purity, Balance, and Overlap have a substantial influence on model outcomes. As a consequence, data scientists should assess whether their datasets meet appropriate thresholds for these metrics (or apply repairing strategies) before deploying them in ML-based tasks.

Conversely, Dimensionality, Data Types, and Dependency exhibited limited or negligible impact on predictive accuracy. Nevertheless, because high dimensionality, strong feature dependencies, or a large number of categorical variables can

considerably increase computational cost and resource usage, we incorporate these factors into our quality model to support the design of more efficient pipelines.

2.5 Impact of Bias-Related Dimensions on Machine-Learning Performance and Fairness

This section presents experimental results assessing the influence of the proposed bias-related metrics on ML models. They aim to determine whether variations in these metrics' values in the training data affect model performance and, for the Disparity metric, also fairness outcomes [20].

Firstly, the experimental findings demonstrate that the proposed metrics effectively estimate the presence of bias in the training data, as their behavior closely aligns with the amount of bias introduced.

Secondly, a comprehensive analysis of all acquired results allows us to conclude that: (a) more representative samples, thus higher Coverage, can improve model performance, (b) greater Density provides more balanced and varied information, thus enhancing performance, and (c) increasing Diversity provides more valuable information, positively affecting performance.

Variations in the Disparity levels within the training dataset do not affect ML performance. Conversely, a clear correlation between the pre-processing bias assessment and the resulting fairness of classification model outcomes was observed. Specifically, datasets with Disparity values near 1 yield models that produce fairer predictions, whereas classifiers produce less fair results when Disparity values significantly deviate from 1. These observations highlight the potential value of the introduced Disparity metric as an initial indicator of bias.

References

1. Ajiboye, A.R., Arshah, R.A., Qin, H., Hauwau, I.K., et al.: Evaluating the effect of dataset size on predictive model using supervised learning technique. Int. J. Comput. Syst. Softw. Eng **1**(1), 75–84 (2015)
2. Ali, H., Salleh, M.M., Saedudin, R., Hussain, K., Mushtaq, M.F.: Imbalance class problems in data mining: a review. Indones. J. Electr. Eng. Comput. Sci. **14**(3), 1560–1571 (2019)
3. Althnian, A., AlSaeed, D., Al-Baity, H., Samha, A., Dris, A.B., Alzakari, N., Abou Elwafa, A., Kurdi, H.: Impact of dataset size on classification performance: an empirical evaluation in the medical domain. Appl. Sci. **11**(2), 796 (2021)
4. Bailly, A., Blanc, C., Francis, É., Guillotin, T., Jamal, F., Wakim, B., Roy, P.: Effects of dataset size and interactions on the prediction performance of logistic regression and deep learning models. Comput. Methods Programs Biomed. **213**, 106504 (2022). https://doi.org/10.1016/J.CMPB.2021.106504
5. Batini, C., Cappiello, C., Francalanci, C., Maurino, A.: Methodologies for data quality assessment and improvement. ACM Comput. Surv. **41**(3), 16:1–16:52 (2009)

6. Batini, C., Scannapieco, M.: Data and Information Quality—Dimensions, Principles and Techniques. Data-Centric Systems and Applications. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-24106-7
7. Chan, J.Y.L., Leow, S.M.H., Bea, K.T., Cheng, W.K., Phoong, S.W., Hong, Z.W., Chen, Y.L.: Mitigating the multicollinearity problem and its machine learning approach: a review. Mathematics **10**(8) (2022). https://doi.org/10.3390/math10081283
8. Chen, H., Chen, J., Ding, J.: Data evaluation and enhancement for quality improvement of machine learning. IEEE Trans. Reliab. **70**(2), 831–847 (2021)
9. Emmert-Streib, F., Yang, Z., Feng, H., Tripathi, S., Dehmer, M.: An introductory review of deep learning for prediction models with big data. Front. Artif. Intell. **3**, 4 (2020)
10. Firmani, D., Tanca, L., Torlone, R.: Ethical dimensions for data quality. ACM J. Data Inf. Qual. **12**(1), 2:1–2:5 (2020)
11. Gupta, S., Gupta, A.: Dealing with noise problem in machine learning data-sets: a systematic review. Procedia Comput. Sci. **161**, 466–474 (2019)
12. Luca, A.R., Ursuleanu, T.F., Gheorghe, L., Grigorovici, R., Iancu, S., Hlusneac, M., Grigorovici, A.: Impact of quality, type and volume of data used by deep learning models in the analysis of medical images. Inform. Med. Unlocked **29**, 100911 (2022)
13. Ma, X., Wang, Y., Houle, M.E., Zhou, S., Erfani, S.M., Xia, S., Wijewickrema, S.N.R., Bailey, J.: Dimensionality-driven learning with noisy labels. In: Dy, J.G., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10–15, 2018, Proceedings of Machine Learning Research, vol. 80, pp. 3361–3370. PMLR (2018). http://proceedings.mlr.press/v80/ma18d.html
14. Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., Galstyan, A.: A survey on bias and fairness in machine learning. ACM Comput. Surv. **54**(6), 115:1–115:35 (2022)
15. Naumann, F., Freytag, J.C., Leser, U.: Completeness of integrated information sources. Inf. Syst. **29**(7), 583–615 (2004)
16. Osisanwo, F., Akinsola, J., Awodele, O., Hinmikaiye, J., Olakanmi, O., Akinjobi, J., et al.: Supervised machine learning algorithms: classification and comparison. Int. J. Comput. Trends Technol. (IJCTT) **48**(3), 128–138 (2017)
17. Patel, H., Gupta, N., Panwar, N., Mittal, R.S., Mehta, S., Guttula, S.C., Mujumdar, S., Afzal, S., Bedathur, S., Munigala, V.: Automatic assessment of quality of your data for AI. In: Dasgupta, G., Simmhan, Y., Srinivasan, B.V., Bhowmick, S., Singhee, A., Ramanath, M., Batra, N., Prasad, A.S. (eds.) CODS-COMAD 2022: 5th Joint International Conference on Data Science & Management of Data (9th ACM IKDD CODS and 27th COMAD), Bangalore, India, pp. 354–357. ACM, New York, NY, USA (2022). https://doi.org/10.1145/3493700.3493774
18. Patel, H., Guttula, S.C., Gupta, N., Hans, S., Mittal, R.S., Nagalapatti, L.: A data-centric AI framework for automating exploratory data analysis and data quality tasks. ACM J. Data Inf. Qual. **15**(4), 44:1–44:26 (2023). https://doi.org/10.1145/3603709
19. Pitoura, E.: Social-minded measures of data quality: fairness, diversity, and lack of bias. J. Data Inf. Qual. (JDIQ) **12**(3), 1–8 (2020)
20. Sancricca, C., Castiglione, P., Cappiello, C.: Exploring the influence of data characteristics on machine learning outcomes. In: International Conference on Business Process Modeling, Development and Support, International Conference on Evaluation and Modeling Methods for Systems Analysis and Development, pp. 227–244. Springer (2025)
21. Sen, S., Singh, K.P., Chakraborty, P.: Dealing with imbalanced regression problem for large dataset using scalable artificial neural network. New Astron. **99**, 101959 (2023)
22. Singh, D., Singh, B.: Investigating the impact of data normalization on classification performance. Appl. Soft Comput. **97**(Part B), 105524 (2020). https://doi.org/10.1016/J.ASOC.2019.105524
23. Vuttipittayamongkol, P., Elyan, E., Petrovski, A.: On the class overlap problem in imbalanced data classification. Knowl. Based Syst. **212**, 106631 (2021). https://doi.org/10.1016/j.knosys.2020.106631

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (http://creativecommons.org/licenses/by-nc-nd/4.0/), which permits any noncommercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if you modified the licensed material. You do not have permission under this license to share adapted material derived from this chapter or parts of it.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Chapter 3
Don't Waste Resources on Data Preparation (Do it On-Demand)



Sonia Bergamaschi, Giovanni Simonini, Luca Zecchini, Angelo Mozzillo, and Giovanni Malaguti

Abstract Data preparation (meant as the process of cleaning and integrating data before use) is essential for modern analytics and AI. Yet, when organizations collect massive and frequently changing datasets, preparing them as they are collected can be prohibitively costly and often unnecessary. A significant portion of stored data may never be analyzed before it becomes obsolete, while only a subset of the data might be truly urgent or relevant for decision-making. This chapter argues for an on-demand approach to data preparation, in which only the data needed for a given task is cleaned and integrated. Such a strategy conserves resources and accelerates access to actionable information. However, on-demand data preparation raises non-trivial challenges: determining which records require cleaning is difficult, especially when "repairing" a record alters its value or relevance. We review methodologies and techniques that address these issues, providing an overview of state-of-the-art approaches to on-demand preparation of structured and semi-structured data in contexts such as data lakes.

S. Bergamaschi (✉) · G. Simonini · L. Zecchini · A. Mozzillo · G. Malaguti
Dipartimento di Ingegneria Enzo Ferrari, Università di Modena e Reggio Emilia, Modena, Italy
e-mail: sonia.bergamaschi@unimore.it

G. Simonini
e-mail: giovanni.simonini@unimore.it

L. Zecchini
e-mail: luca.zecchini@unimore.it

A. Mozzillo
e-mail: angelo.mozzillo@unimore.it

G. Malaguti
e-mail: giovanni.malaguti@unimore.it

© The Author(s) 2027

B. Pernici et al. (eds.), *Discount Quality for Responsible Data Science*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-032-29481-4_3

3.1 Introduction

Data preparation (cleaning [6] and integration [2]) is a prerequisite for modern analytics and AI [11], but it is also a major driver of computational cost, storage overhead, and human effort [13]. The "eager" way to do it, in which data is systematically prepared at ingestion time, has historically been justified by the desire to provide reliable curated datasets to downstream consumers. This paradigm becomes increasingly problematic in data lakes and large data ecosystems, where data is high-volume, high-variety, and frequently changing [2]. In such settings, preparing everything upfront often violates sustainability principles (Chap. 1): a significant portion of the curated data may never be used, while the cost of preparing it is paid anyway.

This chapter advocates an *on-demand* (a.k.a. *pay-as-you-go* [18]) approach to data preparation: clean and integrate only the subset of data that is actually needed to answer a query, support a decision, or train/evaluate a model. By avoiding unnecessary work, on-demand data preparation directly supports Strategy R2 (Reduce) and R3 (Reuse) from the circular-economy-inspired view of sustainable data preparation introduced in Chap. 1. It also aligns with the progressive and interactive nature of modern data work: analysts explore data iteratively, refine their hypotheses, and may discard directions that do not yield value. In these cases, cleaning only what is needed for the current step can shorten latency to insight and reduce wasted computation.

At the same time, on-demand data preparation is not simply "lazy execution" applied to cleaning operators. Deciding what to clean requires reasoning under uncertainty: the subset of relevant records is often unknown before cleaning, and repairing values can change their relevance and therefore the subset itself. Furthermore, preparing data on-demand must reconcile conflicting objectives: provide early useful outputs, preserve eventual correctness when resources allow, and avoid repeated work across successive requests.

We structure the chapter as follows. Section 3.2 motivates on-demand data preparation through real-world scenarios and identifies the sources of waste in eager pipelines. Section 3.3 formalizes the problem and highlights why "relevance can change after repair" is a central challenge. Sections 3.4 and 3.5 introduce core principles and methods for on-demand preparation, focusing on three fundamental works: Schema-agnostic Progressive Entity Resolution [14], BrewER (Entity Resolution On-Demand) [15], and RadlER (Deduplicated Sampling On-Demand) [21]. Section 3.6 discusses how Retrieval-Augmented Generation (RAG) and agentic AI systems can benefit from on-demand data integration. Section 3.7 reviews additional related work and positions on-demand data preparation in the broader landscape. Finally, Sect. 3.8 summarizes open research challenges and practical guidelines.

3.2 The Need for On-Demand Data Preparation

At scale, cleaning and integrating everything upfront systematically creates *waste*: resources are spent on records that might never be queried, on versions that might soon be superseded, and on integration choices that might not match the needs of specific tasks. This is not only an efficiency concern. It is a sustainability concern (Chap. 1), because preparation consumes computation, energy, storage, and human attention (often before any value is realized). From a systems viewpoint, the resulting costs resemble other "pay-per-use" services: organizations increasingly favor on-demand execution and pay-as-you-go models for large-scale analytics, rather than permanently maintaining fully curated data products for all ingested data [14].

On-demand data preparation adopts the same principle for data quality and integration: *prepare only what is needed, when it is needed, and to the level required by the task*. This shift is motivated by three structural trends.

- **Volume and velocity**. Data arrives continuously (streams, logs, IoT, clickstreams), and the cost of maintaining a fully clean integrated replica grows faster than the value extracted from it.
- **Variety and drift**. Sources evolve (schema drift, changing semantics, new attributes), so eagerly engineered pipelines require frequent maintenance and reprocessing, even if downstream tasks do not touch the changed portions.
- **Exploration and open-ended consumption**. Analysts, data scientists, and AI agents explore data iteratively and opportunistically; many intermediate directions are abandoned, so eager cleaning pays costs for unrealized value.

The following use case exemplifies these conditions and motivates the approach. Consider a large organization managing a data lake that ingests (i) transactional records from multiple business units, (ii) third-party reference data (e.g., product catalogs, locations), and (iii) logs from digital channels. The lake is used by different teams: business intelligence, risk and fraud detection, marketing, and data science. Data engineers may be tempted to enforce eager preparation to "stabilize" the lake: standardize formats, reconcile identifiers, deduplicate entities, and correct obvious errors upon ingestion.

In practice, the organization faces at least four recurring sources of waste:

- **Low utilization before obsolescence**. Portions of the ingested data are never queried before they become outdated (e.g., ephemeral logs, short-lived event streams, temporary third-party feeds).
- **Exploration-driven churn**. Data projects iterate on features, subsets, and join paths; many early experiments are abandoned. Cleaning everything upfront does not exploit this churn.
- **Uneven urgency**. Only a subset of the data is urgent for decisions (fraud alerts, supply disruptions, incidents), while most is background. Eager pipelines allocate resources uniformly.

- **Redundant preparation across tasks**. Different teams repeat similar preparation steps with different objectives; eager global preparation may not match any task well, while still incurring high cost.

These issues are amplified when preparation includes expensive tasks such as deduplication and integration across heterogeneous sources. In Entity Resolution (ER), for instance, even scalable workflows rely on blocking and still compare many candidate pairs. If only a small portion of entities will be used in a downstream query, resolving all duplicates upfront is a poor allocation of resources [15, 17].

3.3 Problem Setting and Key Challenges

3.3.1 From Eager Preparation to Task-Driven Preparation

Let D be a collection of raw datasets stored in a data lake. Eager preparation applies a pipeline P to all datasets of D, producing a curated dataset $P(D)$ (or multiple curated views), regardless of future tasks.

In on-demand preparation, the system receives a task T (e.g., a query, a model training step, a sampling request) and prepares only the subset of data needed to produce useful outputs for T. We denote by $D_T \subseteq D$ the data subset that is relevant to T *under cleaned semantics*. The goal is to minimize preparation cost while providing outputs that are sufficiently correct and informative for the user.

Two properties become central:

- **Improved early utility**. With limited time or budget, the system should provide useful partial results earlier than a batch approach that prepares everything (a progressive behavior).
- **Eventual correctness**. When resources allow, the result should converge to what would be obtained by fully cleaning and integrating the relevant data (a batch-equivalent outcome for the considered scope).

These properties echo classical notions in progressive processing, and in particular in Progressive ER [14].

3.3.2 The Central Difficulty: Relevance May Change After Repair

On-demand preparation is complicated by a feedback loop: to know what to clean, we need to know what is relevant, but relevance depends on cleaned values.

Selection predicates can change after cleaning. A query selects records where `city = 'Modena'`. In raw data, some records use `'Modena'`, `'MODENA'`, or

`'Moedna'`. Standardization and typo repair can move records into or out of the selection, so the relevant set is not fixed before cleaning.

Entity identity can change after deduplication. A query aggregates by company (e.g., total sales per supplier). If duplicates exist, aggregation boundaries change after ER. Deciding which records to compare and merge cannot rely solely on the raw representation.

Sampling can change after deduplication. If a data scientist asks for a sample with a target distribution across groups, duplicates can bias group counts. Producing a *clean* sample requires deduplication *within the sample*, while deduplicating the entire dataset is wasteful.

Because of these effects, on-demand systems must combine *task awareness* with *progressive data cleaning and integration*.

3.4 Principles for On-Demand Data Preparation

We distill four design principles that recur across on-demand approaches.

3.4.1 Principle 1: Prioritize What Matters for the Task

On-demand preparation is inherently *task- and context-driven* (Chap. 1). The system should focus cleaning and integration on records that are likely to influence the task outcome: those that contribute to top-ranked answers, belong to the target sample, or appear in the join and selection frontier.

In ER, this translates into prioritizing the comparisons that are most likely to produce duplicates early (progressive ordering) [14]. In query answering, it translates into prioritizing entities that affect query results first (e.g., groups with high weights, records near predicate boundaries) [15].

3.4.2 Principle 2: Progressive Outputs with Bounded Resources

Many tasks are exploratory: users prefer early, approximate, or partial results rather than waiting for full cleaning. Progressive methods aim to produce the *best possible partial solution* given limited resources, and to refine it over time [14]. This is compatible with sustainability goals, as the user can stop early when results are sufficient.

3.4.3 Principle 3: Avoid Repeated Work via Caching and Reuse

On-demand does not imply recomputing from scratch. A system should reuse intermediate results: partial clusters from ER, previously cleaned values, or computed statistics. This is a form of Strategy R3 (Reuse) applied to preparation artifacts.

3.4.4 Principle 4: Be Explicit About Uncertainty and Provenance

When outputs are produced under partial cleaning, they may be uncertain. A system should expose provenance and confidence information (e.g., which entities are fully resolved, which are provisional, how many comparisons were executed). This improves trust and allows users to decide whether further cleaning is needed.

3.5 Fundamental Methods for On-Demand Data Preparation

On-demand data preparation is not a single algorithm, but rather a way of *organizing* cleaning and integration so that the effort follows the user workload. The three contributions discussed in this section exemplify this idea in complementary task settings. They share a common stance: instead of treating data preparation as a mandatory batch stage that must complete before any analysis starts, they make preparation *progressive*, *selective*, and *interruptible*, so that useful outputs can be produced early and refined as needed. In practice, this means changing both the execution model (from "run to completion" to "anytime") and the optimization goal (from minimizing total runtime for full cleaning to maximizing utility under a given budget).

3.5.1 Progressive ER as a Building Block for On-Demand Integration

Entity Resolution (ER) is one of the most expensive integration tasks, because it must identify which records refer to the same real-world entity. When datasets are large and heterogeneous, running ER in batch can be infeasible or simply unjustified: many downstream tasks do not require a globally deduplicated dataset, and in exploratory scenarios users may only need a partial resolution to validate a hypothesis. Progressive ER addresses this gap by rethinking ER as an *anytime* process: given

limited time or resources, it aims to produce the best possible partial resolution by approximating an optimal ordering of comparisons, so that duplicates are discovered early [9, 12, 14, 18].

A central obstacle in bringing Progressive ER to data lakes is that classical methods rely on schema-aware blocking keys, which presuppose stable, curated attributes. Data lakes violate this assumption: they typically mix relational tuples with semi-structured records (e.g., JSON), knowledge graph fragments, and text fields, where "what counts as an attribute" may differ across sources. Schema-agnostic Progressive ER [14] removes this dependency on schema knowledge by exploiting blocking and redundancy signals that can be computed without assuming aligned schemas. Intuitively, the method leverages two complementary regularities: records that describe the same entity tend to share enough tokens or signatures to become "close" under schema-agnostic keys, and they also tend to co-occur in multiple blocks, producing strong redundancy signals in a blocking graph. By emitting comparisons in decreasing order of such signals, Progressive ER can focus early effort on high-yield candidate pairs and postpone low-yield comparisons.

This matters for on-demand preparation because Progressive ER provides a reusable *execution substrate*: it can be invoked only on the portion of data that a task touches, it can be stopped as soon as outputs are sufficient, and it naturally supports refinement if the user continues the analysis. In other words, it turns ER from an all-or-nothing prerequisite into a controllable resource consumer.

3.5.2 *BrewER: Entity Resolution On-Demand for Query Answering*

While Progressive ER provides a general mechanism to prioritize comparisons, BrewER shows how to embed on-demand ER *directly* into the evaluation of SQL queries over dirty data [15, 22]. The motivation is practical: analysts often want to issue selection and aggregation queries over operational data that contain duplicates, and they want answers that *behave as if* the data had been deduplicated, but without paying the cost of fully cleaning the entire dataset beforehand.

BrewER achieves this by shifting the unit of work from "resolve the dataset" to "resolve the entities that the query needs". Conceptually, it executes the query under a cleaned semantics in which duplicate records are grouped into entities; operationally, it constructs those entities on-the-fly, one cluster at a time, and progressively returns query results in the same order as an `ORDER BY` predicate would produce on the cleaned data. This turns query answering into an interactive process: the system can return a first prefix of results quickly, and then refine the answer as more entities are resolved. The design naturally supports top-k usage and "stop-and-resume" behavior, which is central when the user is exploring and does not know in advance how much cleaning is worth paying for.

A key advantage of BrewER is that it addresses the "relevance may change after repair" problem using the query itself as a guide. The `ORDER BY` predicate (and more generally the query workload) induces a notion of which entities are likely to matter first (e.g., those driving the highest aggregates). By binding ER effort to that notion, BrewER reduces the probability of spending resources on entities that will not affect the returned results, and it can adjust rankings as clusters are progressively consolidated.

A representative scenario is fraud monitoring. Suppose an analyst queries for the top merchants by refund volume, but merchant identifiers are fragmented across multiple records (name variants, multiple internal IDs, marketplace accounts). A batch ER run on the entire merchant universe is expensive, while only the top suspicious entities are urgent. BrewER resolves and consolidates exactly those entities that are candidates to appear in the top results first, enabling early investigation and avoiding unnecessary work when the analyst stops after inspecting the first results.

3.5.3 RadlER: Deduplicated Sampling On-Demand for Data Science

On-demand preparation is equally important when the task is not query answering but *sampling*. Sampling is pervasive in data science: practitioners use samples to prototype features, estimate distributions, audit bias, and iterate on models. However, if the dataset contains duplicates (or multiple representations of the same entity), naïve sampling can distort downstream conclusions: duplicates may inflate certain subpopulations, hide rare cases, and bias estimates. The straightforward solution is "deduplicate, then sample", but this is often too expensive and, again, unnecessary for exploratory work. RadlER [20, 21] tackles this tension by producing *deduplicated samples on-demand*: it incrementally builds a clean sample that follows a target distribution over subpopulations, while focusing ER effort only on the entities that are needed to appear in the sample.

The problem setting is intentionally aligned with practical analytics. Users often need a sample that is representative with respect to groups of interest (e.g., region, demographic group, product category) because the goal is to perform comparisons, bias checks, or cohort analyses. RadlER operationalizes this by treating the sample as the primary object and by making deduplication *local* to the sample construction process. As the sample grows, the system resolves potential duplicates among candidate entities, but it avoids spending resources on entities that will not be sampled at all. This yields two benefits that are crucial for sustainability: it provides a usable clean sample early, and it reduces the number of ER comparisons, which dominate the cost in large datasets.

A typical use case is bias assessment and cohort design. Assume an analyst wants a balanced sample of customer profiles across regions to evaluate a classifier. If duplicates are unevenly distributed, naïve sampling may overrepresent some regions and

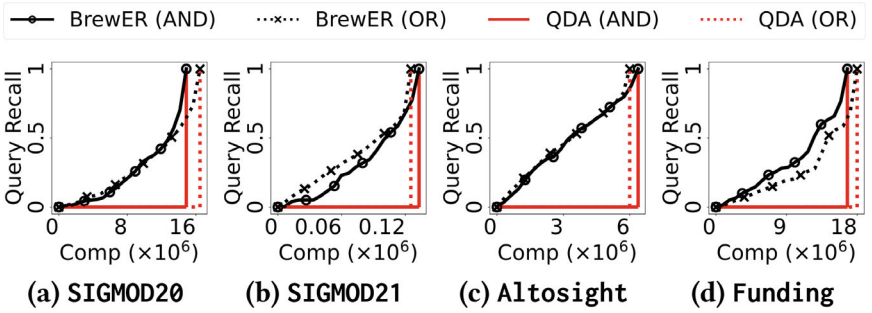


Fig. 3.1 Total comparisons compared to recall [15]

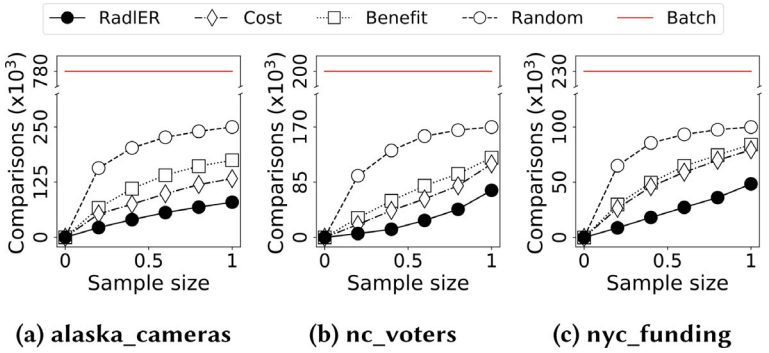


Fig. 3.2 Average number of comparisons to produce a clean sample of size x [21]

underrepresent others, leading to misleading fairness conclusions. RadlER allows the analyst to obtain a clean, distribution-aware sample quickly, without requiring full dataset deduplication, hence enabling earlier auditing with lower resource consumption.

3.5.4 Common Patterns Across the Methods

Although BrewER and RadlER target different tasks (query answering vs. sampling), they embody the same on-demand logic. They both make the task define *where* to spend cleaning effort first (entities that affect top-k results or entities required by a target sample), they both rely on incremental entity construction (so that work can stop and later resume), and they both turn ER from a monolithic preprocessing stage into a controlled, budget-aware component of the workflow. Figures 3.1 and 3.2 show how a progressive solution leads to partial results in a shorter time, since it does not require to analyze all records as the baselines.

Taken together, these ideas suggest how on-demand preparation can be generalized beyond ER to other preparation steps (standardization, error repair, enrichment): whenever the effect of an operation can be localized to the task-relevant subset, the system can prioritize, execute progressively, and stop when marginal gains are no longer worth the cost.

3.6 On-Demand Data Preparation for RAG and Agentic AI

Retrieval-Augmented Generation (RAG) and agentic AI systems [7, 19] make the connection between *data preparation* and *interactive consumption* explicit. Rather than relying on a single, fully curated dataset prepared in advance, these systems oftentimes assemble evidence at inference time: they retrieve relevant pieces of information from large and heterogeneous collections, align them into a coherent context, and often apply lightweight transformations (normalization, entity linking, small joins, aggregations) before producing an answer. In this setting, on-demand integration is not an optimization layered on top of a batch pipeline: it might be the operational default. In fact, the system cannot anticipate in advance which sources (and which fragments of those sources) will be needed for future questions.

3.6.1 Why Context Must be Chosen On-Demand (and Wisely)

In RAG, a model answers a question by retrieving relevant documents/chunks from an external corpus and injecting them into the prompt (or into an intermediate reasoning state). Two practical constraints make eager integration unrealistic. First, questions are open-ended: the space of possible future queries is broad, and precomputing a single globally integrated representation that supports all questions is infeasible in both engineering effort and maintenance cost. Second, retrieval and integration must operate under interactive latency and cost budgets: the system must minimize unnecessary context and avoid expensive global cleaning in order to remain responsive.

At first glance, long-context models might seem to weaken this argument: if the context window is large, why not simply provide much more material and let the model "find" what it needs? Empirical evidence suggests that this strategy is brittle. Needle-in-a-haystack evaluations formalize the risk by embedding a small relevant fact (the needle) into a large amount of distractor context (the haystack) and testing whether the model can still retrieve and use it. Results show that increasing context length does not guarantee robust in-context retrieval; performance depends on context size, the position of the needle, and the nature of distractors [4]. From a data preparation perspective, the implication is direct: dumping large amounts of semi-prepared information into the prompt is not a reliable alternative to targeted retrieval and selective integration.

Moreover, recent evidence indicates that model behavior can degrade as input tokens increase even when the underlying task is unchanged (a phenomenon often described as "context rot" [3]). This reinforces an operational guideline that aligns naturally with on-demand preparation: *context is a scarce resource and must be curated for utility*. Therefore, RAG-oriented preparation must continuously balance two competing objectives: (i) *coverage*, i.e., retrieving enough evidence to answer correctly, and (ii) *compactness*, i.e., excluding low-salience material that increases cost and can reduce reliability. In practice, this motivates the *right sized on-demand* context: prepare and provide only what is needed to support the question, and expand progressively only when the expected marginal benefit justifies the additional tokens and integration work.

3.6.2 Agentic AI as Multi-step On-Demand Preparation

Agentic AI systems amplify the need for on-demand preparation: they decompose a user goal into multiple steps (planning, tool use, iterative retrieval, verification), and each step creates a fresh, localized integration problem. An agent answering "Which suppliers in Italy delivered late more than twice last month?" may retrieve shipment tables from different systems, normalize supplier identifiers, deduplicate supplier names, compute aggregates, and finally explain the result with provenance. Attempting to precompute all such joins and cleanings eagerly would be wasteful: agent *trajectories* are contingent on intermediate outcomes, many branches are pruned, and the same question may never recur in exactly the same form.

Seen through the lens of data preparation, each agent step is a micro-instance of on-demand integration: the agent retrieves a small slice of data, prepares it just enough for the next decision, and either stops (if sufficient) or refines (if uncertain). This is precisely the execution model advocated by on-demand ER methods: effort follows the task frontier. The methods introduced in Sects. 3.5.2 and 3.5.3 suggest a robust blueprint for agentic preparation as well: task-aware prioritization (focus on what affects the current objective), progressive refinement (improve results over time), and explicit uncertainty management (so the agent can decide when more cleaning or evidence is required) [15, 21]. Operationally, this turns on-demand preparation into a control mechanism for agents: it limits wasted work, keeps the working context compact, and supports iterative verification when high-stakes decisions require stronger guarantees.

3.7 Related Work and Positioning

The on-demand approach to data preparation is rooted in several well-established research lines [1] that converge to the same core idea: when full correctness is expensive and often unnecessary, systems should trade exhaustive preprocessing for

incremental improvement guided by usage. From the database perspective, this idea appears in progressive and approximate data management [5], where systems return early results that are refined as more computation is performed, often accompanied by uncertainty estimates. Online aggregation and approximate query processing [8], for instance, were designed to shorten time-to-insight by producing progressively improving answers rather than forcing users to wait for full scans and exact computation. On-demand data preparation [16] extends the same philosophy to upstream steps: rather than approximating the final query answer, it approximates the ideal data preparation steps for the data at hand (e.g., imputing all the missing values) by preparing only the data needed for answering the query (e.g., imputing only the missing values that might affect the result). Progressive ER can be seen as the integration analogue of these approaches: it seeks to maximize early utility by prioritizing comparisons so that matches are discovered quickly under limited resources [14].

A second relevant line is pay-as-you-go data integration, originally motivated by web-scale heterogeneity [10]. Classical arguments in this area observe that many application contexts can obtain useful services without full integration, and therefore propose architectures where semantic mappings and integration semantics are introduced gradually as needed. In this sense, on-demand preparation is a concrete operationalization of pay-as-you-go principles for modern data lakes and AI-driven workloads: instead of insisting on global schema alignment and complete entity consolidation upfront, it invests effort where the workload demonstrates value, and it reuses intermediate results across iterations when possible.

Finally, the broader positioning of on-demand preparation in this book is captured by the notion of *discount quality*: quality improvements should be pursued in a task- and context-driven way, allocating resources where they yield the largest impact for the current objective [13]. On-demand preparation makes this principle actionable by narrowing the scope of preparation to task-relevant data and by enabling early stopping and progressive refinement; the result is not a different set of preparation operations, but a different *execution discipline* that explicitly trades universal readiness for measured, workload-driven value.

3.8 Open Challenges and Practical Guidelines

Despite recent progress, on-demand data preparation remains challenging in both theory and practice, because it replaces a single "prepare everything once" decision with a continuous sequence of choices about *what* to prepare, *when* to prepare it, and *how far* to go before stopping. A first open issue concerns **priority definition and learning**. Methods such as BrewER and RadlER can exploit explicit task signals (e.g., top-k answers, target strata for sampling), but in realistic deployments, priorities are often implicit and may vary across users and over time [15, 21]. A practical guideline is to make prioritization a first-class component of the system: start from workload-aware heuristics (e.g., predicates, ranking functions, group-by keys, recent access patterns), then progressively learn better policies from query logs, user interactions,

and downstream utility signals, while exposing controllable knobs (e.g., time budget, desired confidence, or "good enough" thresholds) so that users can trade quality for latency.

A second challenge is **stability under feedback loops**, i.e., the fact that relevance can change after repair. Cleaning may bring new records into scope (predicate repair), while deduplication may change aggregation and join boundaries (identity repair), and these shifts can trigger further cleaning. Designing incremental semantics that remain stable under such loops is non-trivial, especially when the system is expected to return intermediate answers and support early stopping. In practice, systems should *(i)* keep intermediate results explicitly labeled as provisional, *(ii)* track which parts of the answer are sensitive to unresolved entities or unclean values, and *(iii)* define termination criteria in terms of marginal impact on the task (e.g., "no change in the top-k for the last b resolved entities") rather than in terms of completing a fixed amount of cleaning.

A third issue is **reuse without drifting back to eager preparation**. On-demand systems gain much of their sustainability advantage by avoiding unnecessary work, but repeated tasks will naturally revisit overlapping data regions, so some caching of preparation artifacts is beneficial. The difficulty is deciding what to retain (e.g., standardized values, partial clusters, blocking indices, match decisions), how to invalidate these artifacts under updates, and how to share them across tasks that may have different quality requirements. A practical direction to handle this might be to cache at the *entity- and evidence-level* rather than at the dataset level: retain compact provenance-aware artifacts (e.g., resolved entity identifiers with supporting evidence and timestamps) and adopt conservative invalidation rules under drift, so that reuse accelerates future on-demand tasks without forcing a global re-curation cycle.

A fourth challenge concerns **interoperability in heterogeneous data lakes**. Schema-agnostic methods reduce dependence on curated schemas, but production systems still require robust connectors, scalable indexing, and lightweight metadata to support discovery and selective retrieval. In practice, on-demand preparation benefits from a thin layer of metadata that can be maintained cheaply (e.g., source descriptors, basic statistics, token indices, sampling synopses), enabling the system to localize work quickly without committing to full semantic integration.

Finally, for **RAG and agentic AI**, on-demand preparation must be coupled with trust mechanisms and context discipline. Long-context models are not reliably improved by simply adding more tokens: "needle-in-a-haystack" evaluations show that retrieval quality depends on context structure and distractors, and recent evidence suggests that performance can degrade as irrelevant context grows ("context rot") [3, 4]. Therefore, a practical guideline is to treat context as a scarce resource: retrieve and integrate evidence progressively, prefer compact structured intermediate representations (normalized entities, small tables, aggregated summaries) over raw dumps, and attach provenance so that the system can verify critical claims and decide when additional preparation is warranted. In this view, on-demand preparation is not only a cost-saving strategy, but also a reliability strategy: it limits noise, makes uncertainty visible, and focuses cleaning and integration where it most affects the correctness of decisions.

3.9 Conclusion

Traditional data preparation is increasingly misaligned with the scale, variety, and dynamism of modern data ecosystems. On-demand preparation provides a sustainability-oriented alternative: it reduces wasted computation and human effort by preparing only the data needed for a task, and often by producing useful partial results early. This chapter reviewed foundational methods that exemplify this paradigm: Schema-agnostic Progressive ER [14], BrewER for query-driven ER on-demand [15], and RadlER for deduplicated sampling on-demand [21]. We also argued that RAG and agentic AI systems require on-demand integration by definition, and that long-context limitations further motivate "right-sized" preparation [3, 4]. Overall, on-demand preparation is a key mechanism to operationalize "discount" quality in sustainable data preparation pipelines [13].

References

1. Alagiannis, I., Borovica, R., Branco, M., Idreos, S., Ailamaki, A.: NoDB: efficient query execution on raw data files. In: ACM International Conference on Management of Data (SIGMOD), pp. 241–252 (2012)
2. Dong, X.L., Srivastava, D.: Big Data Integration. Springer (2015)
3. Du, Y., Tian, M., Ronanki, S., Rongali, S., Bodapati, S.B., Galstyan, A., Wells, A., Schwartz, R., Huerta, E.A., Peng, H.: Context length alone hurts LLM performance despite perfect retrieval. In: Findings of the Association for Computational Linguistics: EMNLP, pp. 23281–23298 (2025)
4. Gao, Y., Xiong, Y., Wu, W., Huang, Z., Li, B., Wang, H.: U-NIAH: unified RAG and LLM evaluation for long context needle-in-a-haystack. ACM Trans. Inf. Syst. **44**(3), 55 (2025)
5. Haas, P.J., Hellerstein, J.M.: Ripple joins for online aggregation. ACM SIGMOD Rec. **28**(2), 287–298 (1999)
6. Ilyas, I.F., Chu, X.: Data Cleaning. ACM (2019)
7. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., Kiela, D.: Retrieval-augmented generation for knowledge-intensive NLP tasks. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Proceedings of 34th Conference on Neural Information Processing Systems (NeurIPS 2020). Advances in Neural Information Processing Systems, vol. 33, pp. 9459–9474 (2020)
8. Li, K., Li, G.: Approximate query processing: what is new and where to go? a survey on approximate query processing. Data Sci. Eng. **3**(4), 379–397 (2018)
9. Maciejewski, J., Nikoletos, K., Papadakis, G., Velegrakis, Y.: Progressive entity matching: a design space exploration. Proc. ACM Manag. Data **3**(1), 1–25 (2025)
10. Madhavan, J., Jeffery, S.R., Cohen, S., Dong, X., Ko, D., Yu, C., Halevy, A.: Web-scale data integration: you can only afford to pay as you go. In: Proceedings of CIDR, pp. 342–350 (2007)
11. Mohammed, S., Budach, L., Feuerpfeil, M., Ihde, N., Nathansen, A., Noack, N.S., Patzlaff, H., Naumann, F., Harmouch, H.: The effects of data quality on machine learning performance on tabular data. Inf. Syst. **132**, 102549 (2025)
12. Papenbrock, T., Heise, A., Naumann, F.: Progressive duplicate detection. IEEE Trans. Knowl. Data Eng. (TKDE) **27**(5), 1316–1329 (2015). https://doi.org/10.1109/TKDE.2014.2359666
13. Pernici, B., Cappiello, C., Bono, C., Sancricca, C., Catarci, T., Angelini, M., Filosa, M., Palmonari, M., Paoli, F.D., Bergamaschi, S., Simonini, G., Mozzillo, A., Zecchini, L.: Sustainable quality in data preparation. ACM J. Data Inf. Qual. **17**(4:23), 1–33 (2025)

14. Simonini, G., Papadakis, G., Palpanas, T., Bergamaschi, S.: Schema-agnostic progressive entity resolution. In: Proceedings of the IEEE International Conference on Data Engineering (ICDE), pp. 53–64 (2018)
15. Simonini, G., Zecchini, L., Bergamaschi, S., Naumann, F.: Entity resolution on-demand. Proc. VLDB Endow. (PVLDB) **15**(7), 1506–1518 (2022). https://doi.org/10.14778/3523210.3523226
16. Stonebraker, M., Bruckner, D., Ilyas, I.F., Beskales, G., Cherniack, M., Zdonik, S.B., Pagan, A., Xu, S.: Data curation at scale: the data tamer system. In: Biennial Conference on Innovative Data Systems Research (CIDR) (2013)
17. Wang, S., Zhao, S., Miao, S., Kwashie, S., Bewong, M., Hu, J., Nofong, V.M., Feng, Z.: Faster: on-demand entity resolution in property graphs. In: International Semantic Web Conference, pp. 177–195. Springer (2025)
18. Whang, S.E., Marmaros, D., Garcia-Molina, H.: Pay-as-you-go entity resolution. IEEE Trans. Knowl. Data Eng. (TKDE) **25**(5), 1111–1124 (2013). https://doi.org/10.1109/TKDE.2012.43
19. Xi, Z., Chen, W., Guo, X., He, W., Ding, Y., Hong, B., Zhang, M., Wang, J., Jin, S., Zhou, E., Zheng, R., Fan, X., Wang, X., Xiong, L., Zhou, Y., Wang, W., Jiang, C., Zou, Y., Liu, X., Yin, Z., Dou, S., Weng, R., Qin, W., Zheng, Y., Qiu, X., Huang, X., Zhang, Q., Gui, T.: The rise and potential of large language model based agents: a survey. Sci. China Inf. Sci. **68**(2), 121101 (2025). https://doi.org/10.1007/s11432-024-4222-0
20. Zecchini, L., Abedjan, Z., Efthymiou, V., Simonini, G.: RadlER: deduplicated sampling on-demand. Proc. VLDB Endow. (PVLDB) **18**(12), 5319–5322 (2025). https://doi.org/10.14778/3750601.3750661
21. Zecchini, L., Efthymiou, V., Naumann, F., Simonini, G.: Deduplicated sampling on-demand. Proc. VLDB Endow. (PVLDB) **18**(8), 2482–2495 (2025). https://doi.org/10.14778/3742728.3742742
22. Zecchini, L., Simonini, G., Bergamaschi, S., Naumann, F.: BrewER: entity resolution on-demand. Proc. VLDB Endow. (PVLDB) **16**(12), 4026–4029 (2023). https://doi.org/10.14778/3611540.3611612

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (http://creativecommons.org/licenses/by-nc-nd/4.0/), which permits any noncommercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if you modified the licensed material. You do not have permission under this license to share adapted material derived from this chapter or parts of it.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Chapter 4
Reconciliation-Based Data Enrichment



Flavio De Paoli, Matteo Palmonari, and Federico Belotti

Abstract Data enrichment is a crucial component of data preparation, aimed at improving the quality of input datasets by integrating external information from third-party sources. One approach to support this process is reconciliation-based data enrichment. In this approach, data providers organize their information around entities and expose reconciliation services for data consumers. Data consumers use these services to match tabular values in their data with entities in the source they want to access, enabling them to fetch additional information by querying the source with canonical identifiers. This approach reduces information asymmetry and facilitates access to external data, making enrichment more effective and user-friendly. The chapter introduces reconciliation-based enrichment, reviews current techniques for entity reconciliation in tabular data, and highlights the emerging role of large language models in supporting such tasks. Finally, it discusses the advantages of reconciliation-based enrichment with respect to sustainability principles, including reducing computational effort, limiting human intervention, and promoting resource efficiency in data preparation pipelines.

4.1 Introduction

Data collected from real-world sources is often raw, heterogeneous, and incomplete, and therefore cannot be directly exploited in AI applications or analytical workflows. Before meaningful analyses can be performed, data must undergo a preprocessing and enrichment phase aimed at improving quality, consistency, and usability. Data enrichment is the process by which existing datasets are augmented with additional

F. De Paoli (✉) · M. Palmonari · F. Belotti
University of Milan-Bicocca—DISCo, Milano, Italy
e-mail: flavio.depaoli@unimib.it

M. Palmonari
e-mail: matteo.palmonari@unimib.it

F. Belotti
e-mail: federico.belotti@unimib.it

© The Author(s) 2027

B. Pernici et al. (eds.), *Discount Quality for Responsible Data Science*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-032-29481-4_4

information derived from external sources, thereby making them more complete, informative, and actionable.

A common approach to enrich a source dataset with content from one or more third-party data sources is to obtain a local copy of that data and apply data integration techniques [2]. An alternative approach, aligned with sustainability strategies, is based on *entity reconciliation* and fetches only the data relevant to the input table, possibly through an API, thus avoiding the assumption of full control over the third-party data sources.

A reconciliation-based process involves identifying the entities present in a dataset–such as people, locations, or organizations–and annotating them with unique identifiers drawn from reference vocabularies. Using vocabularies in this way provides a shared semantic framework, which facilitates the integration of datasets managed by different organizations and structured according to heterogeneous schemas. In other words, vocabularies act as standardized "glue" that aligns data elements across sources.

Vocabularies define terms, labels, and sometimes relationships among concepts, and are critical for standardizing language and reducing lexical ambiguity. When combined with ontologies, which provide a formal, structured representation of domain knowledge including classes, relationships, and logical constraints, ambiguities can be further reduced and dataset quality enhanced. The adoption of reference vocabularies and ontologies also significantly increases the reusability of data across domains and applications. Examples of widely used vocabularies include Schema.org, which provides terms for describing web resources, and FOAF (Friend Of A Friend), which models social connections and person-related information.

Of particular importance in modern data-driven applications are Knowledge Graphs (KGs), which represent entities as graph nodes and relationships among them as labeled edges. Entities in a KG are often defined according to an ontology, providing semantic structure that enables reasoning and inference. Prominent examples include Wikidata, DBpedia, and the Google Knowledge Graph, which have been widely employed for data integration, enrichment, and semantic search.

A critical aspect of data enrichment is entity disambiguation, which resolves ambiguities when the same name or label may refer to different real-world entities. This process typically involves matching entities in source datasets to a target KG to assign unique identifiers, ensuring that each entity can be clearly distinguished. Although automated algorithms can perform much of this reconciliation, in many cases human expertise is necessary to resolve ambiguous or complex situations. Once entities are reconciled, datasets can be integrated seamlessly. For example, a dataset containing weather observations may be linked to a KG of cities, allowing weather data to be enriched with additional geographic, demographic, or temporal information through identifier alignment.

While there exists a wide array of tools to support data preprocessing, annotation, and enrichment, these tasks remain challenging and resource-intensive, often requiring substantial expertise, time, and computational resources. Many existing tools are designed for domain experts, limiting accessibility for less experienced users. Nonetheless, such enrichment processes are essential for any high-quality data

analysis pipeline, as they provide the semantic foundation upon which AI models, analytics, and decision-making systems can operate effectively.

This chapter introduces reconciliation-based enrichment, reviews current techniques for entity reconciliation in tabular data, and highlights the emerging role of large language models in supporting such tasks. Finally, it discusses the advantages of reconciliation-based enrichment with respect to sustainability principles, including reducing computational effort, limiting human intervention, and promoting resource efficiency in data preparation pipelines.

Section 4.2 introduces the enrichment process and its main tasks–Modification, Reconciliation, and Extension–through an illustrative example. Section 4.3 presents the semantic annotation of tabular data, with particular focus on techniques addressing the key problem of entity reconciliation. Section 4.4 compares reconciliation techniques in terms of accuracy, execution time, memory usage, and sustainability. Finally, Sect. 4.5 concludes the chapter.

4.2 Enrichment Process

Data enrichment can be regarded as a specialized form of data integration aimed at augmenting a source dataset with additional information retrieved from external data sources. Unlike classical integration scenarios, enrichment is typically asymmetric: only the source dataset is under the control of the integrator, while external sources are autonomous, independently evolving, and only partially known. Moreover, data originating from different sources are often highly heterogeneous in terms of structure, representation, and semantics. This asymmetry significantly affects both the problem formulation and the set of applicable techniques.

As a consequence, classical data integration techniques–such as *Record Linkage* and schema-level integration approaches–are often insufficient or not directly applicable to enrichment tasks. Instead, data enrichment requires methods capable of coping with limited knowledge about external sources, semantic heterogeneity, and the absence of global schemas, motivating the adoption of semantic-based approaches.

To this end, semantic enrichment can be decomposed into two core tasks: *reconciliation* and *extension*. Reconciliation can be regarded as a form of entity linking applied to tabular data, consisting in mapping table mentions to entities in a target system of identifiers (e.g., geographical coordinates for locations) or a Knowledge Graph (e.g., Wikidata IDs for popular persons or organizations) so as to assign unique and persistent identifiers. The task is typically carried out through automatic algorithms; however, due to the inherent ambiguity and contextual dependence of textual mentions, the resulting mappings may be uncertain or ambiguous, thus requiring human intervention to validate, disambiguate, or refine the associations.

Once mentions have been reconciled, the extension task exploits the assigned identifiers to retrieve additional information from external data sources and incorporate it into the original table. This process is commonly based on Linked Data princi-

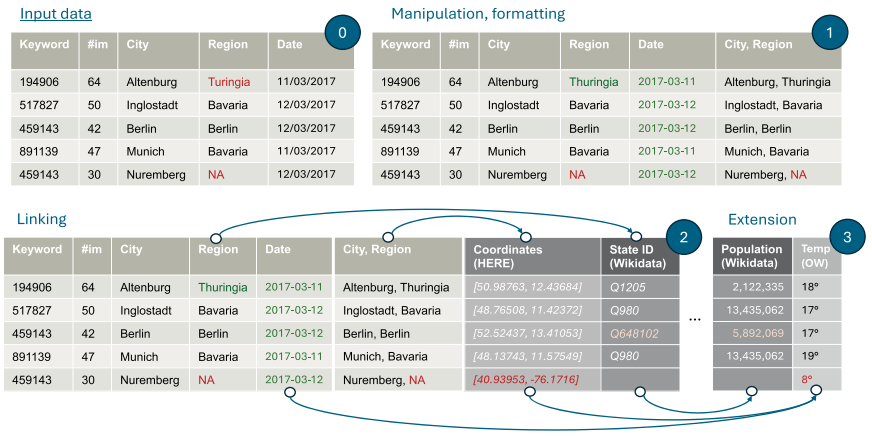


Fig. 4.1 Example of data preparation for optimizing digital marketing campaigns. Numbers identify data preparation steps. The background colors indicate data from different sources. Arrows trace how columns contribute to derived content. Text colors indicate data quality: red for missing or incorrect values; green for improved, reliable values

ples [5], allowing heterogeneous datasets to be combined through shared identifiers and semantic relationships.

To illustrate the process of enrichment, we consider the data preparation task in the digital marketing domain that has been discussed in [16]. With the objective of maximizing the performance of keyword-based marketing campaigns, we plan to study the correlation between impressions–the number of times the advertisement (AD) associated with the campaign has been displayed in search results–and weather conditions. The available data are the reports from the online advertising platforms, e.g., Google Ads, that include keyword identifiers, number of impressions, locations (city-region), and dates of the collection (Step 0 of Fig. 4.1). In this context, data preparation is a massive operation involving the management of 5 billion active keywords and the generation of more than 1M clicks per day in 234 countries and 15 languages.

As shown in the figure, accuracy and completeness issues may occur in the input data, such as misspellings, missing values, and so on. To prepare the data for analysis, assume that two data-enrichment tasks are required: (1) retrieving location information to normalize the data by population (state population in this case); and (2) retrieving weather information for the location on the given day (temperature in this example), as campaign optimization depends on weather conditions, which are known to influence user purchasing behavior [8].

Data on the population at the state level can be collected from open data sources such as Wikidata, while weather data can be obtained from weather services such

Output data

Keyword	#i	City	Region	Date	City, Region	Coordinates (HERE)	State ID (Wikidata)	Population (Wikidata)	Temp (OW)
194906	64	Altenburg	Thuringia	2017-03-11	Altenburg, Thuringia	*[50.98763, 12.43684]*	Q1205	2,122,335	18°
517827	50	Inglostadt	Bavaria	2017-03-12	Inglostadt, Bavaria	*[48.76508, 11.42372]*	Q980	13,435,062	17°
459143	42	Berlin	Berlin	2017-03-12	Berlin, Berlin	*[52.52437, 13.41053]*	Q64	3,782,202	17°
891139	47	Munich	Bavaria	2017-03-11	Munich, Bavaria	*[48.13743, 11.57549]*	Q980	13,435,062	19°
459143	30	Nuremberg	Bavaria	2017-03-12	Nuremberg, Bavaria	*[49.45421, 11.07752]*	Q980	13,435,062	12°

Fig. 4.2 The output table for the example of data preparation for digital marketing campaigns

as OpenWeather.[1] Population data may suffer from timeliness issues, whereas both temperature and population data can be affected by the presence of outliers.

Additional data issues related to weather services concern the use of specific formats –geocoordinates for locations and ISO 8601 for dates– which may differ from those adopted in the input data. Moreover, locality information can be incomplete or inaccurate, thus requiring disambiguation actions. For instance, geocoding services can be used to retrieve city coordinates (e.g., HERE[2]), while Wikidata reconciliation services such as OpenRefine[3] can support the disambiguation of regions. These matching operations are inherently subject to uncertainty and potential errors.

As illustrated in Fig. 4.1, the data preparation process is structured as a pipeline composed of three main tasks–Manipulation and Formatting, Reconciliation, and Extension–followed by a validation and refinement phase aimed at detecting, assessing, and correcting anomalies in order to produce a high-quality enriched table.

In the remainder of this section, each phase is described in detail to clarify how the final output reported in Fig. 4.2 is systematically obtained through successive enrichment and refinement cycles.

4.2.1 Step 1: Manipulation and Formatting

The first step of the enrichment process focuses on the manipulation and formatting of the input data to improve data quality and ensure compatibility with subsequent enrichment operations. This step primarily addresses syntactic aspects of data preparation, targeting quality dimensions such as accuracy, consistency, and interoperability, which are prerequisites for effective semantic linking and data extension.

Manipulation activities include the correction of spelling errors in textual attributes, the normalization of character encoding, and the harmonization of naming conventions for locations. City and region names, in particular, may exhibit substantial lexical variability due to abbreviations, alternative spellings, or language-dependent forms. Standardizing these values mitigates ambiguity and increases the

[1] https://openweathermap.org/.

[2] https://www.here.com/platform/geocoding.

[3] https://wikidata.reconci.link/.

robustness of downstream matching procedures. When location information is provided in separate columns (e.g., city and region), these attributes can be concatenated to form a composite location field, thereby facilitating subsequent disambiguation and geocoding.

Formatting operations focus on achieving syntactic interoperability with external services. Temporal attributes are converted into standardized representations, such as ISO 8601, to comply with the input requirements of weather data services. Similarly, spatial attributes may need to be reformatted to match the expected structure of geocoding services, for instance, by enforcing a consistent ordering of city and region names or by removing unnecessary textual elements. These transformations are essential to prevent query failures and misinterpretations caused by mismatches.

Although this step does not introduce new information, it plays a critical role in the enrichment pipeline. Issues left unresolved at this stage may propagate to later phases, negatively affecting the effectiveness of semantic linking and the reliability of the enriched data. For this reason, manipulation and formatting constitute a foundational preparation phase that directly impacts the quality and robustness of the linking and extension steps discussed in the following sections.

4.2.2 Step 2: Reconciliation

The second step of the enrichment process concerns the linking of values appearing in the input table to structured identifiers that enable integration with external data sources. Reconciliation establishes explicit connections between input values and entities, records, or identifiers provided by third-party systems, such as knowledge graphs, geocoding services, or domain-specific databases. As illustrated in Fig. 4.1, this step marks the transition from syntactic data preparation to semantic interpretation.

Reconciliation operations are typically performed using matching and linking techniques and are inherently subject to uncertainty. Ambiguity, lexical variability, and incompleteness in the input data may lead to incorrect or suboptimal associations. The downstream impact of such errors varies depending on the nature of the mismatch and on the role of the linked identifiers in subsequent enrichment steps.

For example, a geocoding service may mislink the city name “Nuremberg” to a location in the United States, resulting in coordinates that refer to a completely different geographical area and, consequently, to unrelated weather conditions. In contrast, Wikidata reconciliation may associate “Berlin” with the Berlin–Brandenburg Metropolitan Region[4] rather than with the city entity.[5] While the city could be considered a more precise match, the resulting discrepancy may have a limited effect on downstream analysis, as several attributes of the two entities are closely related.

[4] https://www.wikidata.org/wiki/Q648102.

[5] https://www.wikidata.org/wiki/Q64.

Different reconciliation strategies can be adopted depending on the characteristics of the input data and the target sources. *String-based approaches* rely on lexical similarity measures, such as exact matching, edit distance, or token-based similarity, and are generally efficient but sensitive to spelling variations and ambiguity. *Semantic approaches* exploit background knowledge, such as entity types, relationships, or contextual information available in knowledge graphs, to perform disambiguation beyond surface forms. *Hybrid approaches* combine lexical similarity with semantic constraints or contextual signals, aiming to balance efficiency and accuracy. These strategies differ in terms of robustness, computational cost, and error profiles, and are discussed in detail in the next section.

Given the inherent uncertainty of automated reconciliation, human-in-the-loop mechanisms play a crucial role in ensuring data quality. A first class of corrective techniques is based on rule-driven validation and correction. Domain rules, integrity constraints, and plausibility checks can be used not only to identify inconsistencies—such as links that violate geographic, temporal, or semantic constraints, or that produce implausible attribute values—but also to systematically correct them. In this way, human intervention can be captured in the form of explicit rules that prevent the recurrence of the same errors in future executions.

For example, in the case of the value "Berlin", human feedback may establish a context-specific rule stating that, for the purposes of the considered analysis, the city entity should be preferred over the corresponding metropolitan region. Once defined, such a rule can be applied automatically to resolve similar ambiguities, thereby improving consistency and reducing the need for repeated manual intervention. Rules of this kind can be used either to guide automated re-linking or to constrain the set of candidate entities considered during the matching process.

Beyond rule-based validation, active learning techniques can be employed to guide human intervention more effectively. In this setting, the system selectively presents uncertain or ambiguous linking cases to human experts, whose feedback is then used to refine matching models, update similarity thresholds, or enrich training data. By focusing human effort on the most informative cases, active learning enables iterative improvement of linking quality while limiting manual effort. Over time, corrections provided by human operators can be exploited to improve automated linking strategies and reduce the frequency of similar errors in subsequent executions.

4.2.3 Step 3: Extension

The extension step enriches the input table by incorporating additional attributes retrieved from external data sources using the identifiers and normalized values produced in the previous steps. At this stage, new information is effectively introduced into the dataset, with the goal of supporting downstream analytical tasks that cannot be addressed using the original input alone.

Extension operations typically involve querying third-party services, knowledge graphs, or domain-specific datasets to retrieve attributes associated with the linked

identifiers. In the running example illustrated in Fig. 4.1, this includes the retrieval of population data at the state level and weather information for the specified location and date. More generally, extension can introduce a wide range of contextual, statistical, or descriptive attributes, depending on the analytical objectives and the available external sources.

Since extension relies entirely on the correctness of previously established links, errors introduced during the reconciliation step are propagated at this stage. Incorrect or imprecise identifiers may result in the retrieval of irrelevant, misleading, or inconsistent data, which can negatively affect downstream analysis. Therefore, extension represents a critical amplification point in the enrichment pipeline: small inaccuracies in reconciliation may systematically propagate through attribute retrieval and aggregation, producing structurally consistent but semantically incorrect data.

In addition to error propagation, extension introduces new sources of incompleteness and inconsistency. External data sources may differ in terms of temporal coverage, update frequency, granularity, and reliability, and may return missing, outdated, or partially conflicting values. These issues do not necessarily invalidate the enrichment process, but they require explicit treatment before the data can be reliably used for analysis.

For this reason, extension naturally leads to subsequent validation and refinement steps. Once additional attributes have been incorporated, the enriched dataset must be examined to identify missing values, outliers, and inconsistencies introduced or amplified during enrichment.

4.2.4 Step 4: Iterative Validation and Refinement

Following the extension step, the enriched dataset may exhibit missing values, inconsistencies, implausible values, or partially derived attributes that require further assessment before the data can be considered analytically reliable. Such issues may originate from incomplete external sources, temporal misalignment across datasets, service-level failures, ambiguity in entity linking, or error propagation from previous enrichment stages.

This step consolidates value imputation, inference, augmentation, and validation activities within a unified iterative refinement process. The objective is to detect potential quality issues, apply corrective or enrichment operations, and reassess the dataset until a satisfactory level of consistency and completeness is achieved.

Value imputation techniques are adopted to handle missing or incomplete data. Basic strategies include default assignments, aggregation at higher granularity levels, or statistically derived central tendencies, whereas more advanced approaches may exploit correlations among attributes, temporal dependencies, or learned predictive models. Similarly, outlier detection mechanisms can be applied to identify anomalous or implausible values, which may subsequently be removed, corrected, or replaced through imputation. These operations are typically automated but may generate borderline cases where corrective actions are not unequivocally justified.

Human-in-the-loop intervention plays a central role in this iterative refinement phase. Human supervision may be triggered when confidence scores fall below predefined thresholds, when inferred or augmented values have a significant impact on downstream analytical results, or when domain expertise is necessary to resolve semantic ambiguities. The intervention may include identifying anomalies, validating candidate corrections, revising linking decisions, or re-executing portions of the enrichment pipeline under updated constraints. For example, reconciliation for "Nuremberg" should be performed again once the correct region is specified to address inaccuracies in coordinates and weather data. Importantly, validated corrections can be formalized into explicit rules or parameter adjustments, thereby progressively improving the automated components of the process.

Through successive validation cycles, inconsistencies are progressively reduced, confidence scores are recalibrated, and derived attributes are consolidated.

The outcome of this step is a semantically enriched and internally consistent table that integrates reconciled identifiers, extended attributes, inferred values, and validated augmentations. The final result of the iterative validation and refinement process is illustrated in Table 4.2, which represents the consolidated and quality-assessed output of the enrichment pipeline.

4.3 Semantic Annotation of Tables

Semantic technologies enable the identification of a shared space of identifiers across heterogeneous datasets, thereby supporting data linking, integration, and enrichment. When dealing with tabular data, semantics can be associated both with the content of table cells and with the table schema. According to the "Semantic Web Challenge on Tabular Data to Knowledge Graph Matching" (SemTab) [13], the former corresponds to Cell Entity Annotation, while the latter includes Cell Type Annotation and Cell Property Annotation.

Cell Entity Annotation (CEA) is a cell-level annotation task, also referred to as instance-level annotation. Its objective is to associate textual values occurring in table cells (mentions) with entities in a target Knowledge Graph (KG)–for instance, Wikipedia entities for real-world objects–or, more generally, with unique identifiers drawn from reference systems such as geographical coordinates for location entities. The task can be formulated as an entity linking problem, whose goal is to disambiguate mentions and map them to the corresponding entities in the target KG or ide columns with classes defined in a KG ontology, thus capturing the semantic type of the values contained in each column. Cell Property Annotation (CPA) is another schema-level task that identifies semantic relations between pairs of columns by mapping them to properties defined in the KG ontology.

Schema-level annotations are typically inferred from the results of cell-level annotation, for instance, by analyzing the frequency and distribution of types and properties associated with the entities identified during reconciliation [7, 14]. Consequently, the quality of schema-level annotations–and, more broadly, of the overall

table enrichment process–critically depends on the accuracy of cell entity linking algorithms [6, 8, 9].

Entity linking, therefore, represents a central and critical step in the semantic annotation and enrichment process. Ambiguity, lexical variations, incomplete contextual information, and schema heterogeneity make reconciliation decisions inherently uncertain and error-prone, with potential cascading effects on downstream annotation and data extension tasks.

From an algorithmic perspective, existing entity linking approaches can be broadly categorized into three main families: heuristic-based, machine-learning-based, and language-model-based methods. These families differ in terms of decision mechanisms, supervision requirements, interpretability, and scalability characteristics. They are not mutually exclusive and do not represent orthogonal design choices; rather, they capture dominant modeling paradigms that are frequently combined with collective inference techniques, constraint-driven validation, and human-in-the-loop refinement strategies in practical systems.

4.3.1 Heuristic-Based Approaches to Reconciliation

Heuristic-based entity linking methods rely on manually designed rules and similarity functions to match mentions to candidate entities. Typical techniques include string similarity measures (e.g., edit distance, token overlap, phonetic similarity), normalization and canonicalization rules, and the exploitation of structured constraints such as type compatibility, geographic containment, or functional dependencies. These approaches are generally transparent, interpretable, and computationally efficient, which makes them particularly suitable for large-scale settings and interactive data preparation tools. However, their effectiveness is highly dependent on domain-specific tuning and they tend to degrade in the presence of noisy inputs or highly ambiguous mentions. Representative systems include SILK, which expresses linking rules using a declarative language combining similarity metrics and aggregation operators [18], as well as early reconciliation services integrated into tools such as OpenRefine.[6] More often, they are used in combination with feature-based machine learning algorithms [14].

4.3.2 Machine-Learning-Based Approaches to Reconciliation

Machine-learning-based approaches formulate entity linking as a classification or ranking problem, in which candidate entities are scored using learned models. Features typically capture lexical similarity, contextual coherence, structural properties

[6] https://openrefine.org/

of the knowledge graph, and, when available, collective signals across multiple mentions. Both supervised and semi-supervised techniques have been proposed, ranging from probabilistic models inspired by record linkage theory to more recent neural architectures based on distributed representations.

These approaches generally achieve higher accuracy than purely heuristic methods, particularly in ambiguous cases, but require labeled training data and introduce additional complexity in model training, deployment, and maintenance. A well-known example is DBpedia Spotlight, which combines statistical disambiguation with contextual evidence [15]. A more recent feature-based approach is Alligator [1], which uses a neural network operating on local feature vectors to transform engineered features into a normalized matching score. In practice, this step re-ranks a set of candidate entities by considering signals such as textual similarity and schema-level properties. Another approach explores deep learning architectures specifically designed for tabular data. One notable example is TURL [10], which adapts BERT [11] to explicitly model tabular structure and is subsequently fine-tuned to perform downstream tasks, including entity disambiguation.

4.3.3 Language-Model-Based Approaches to Reconciliation

Recent approaches leverage large language models (LLMs) as a distinct paradigm for entity linking, exploiting their parametric knowledge and contextual reasoning capabilities. For instance, TableLlama [20] is based on the autoregressive LLaMA 2-7B model [17], which is fine-tuned using instruction tuning techniques [19] to perform table-related tasks. These models are typically trained and evaluated on benchmark datasets designed to support their general-purpose objectives, and they are reported to achieve state-of-the-art performance on the evaluation tasks.

In this setting, entity linking is often formulated as a prompt-based inference task, where mentions and their surrounding context are provided as input to the model, which then generates, ranks, or selects the most plausible entity identifiers. Such methods can be particularly effective in low-resource or zero-shot scenarios, as they are able to handle lexical variations and implicit contextual cues more robustly than traditional models.

At the same time, LLM-based approaches raise important concerns regarding reproducibility, explainability, controllability, and long-term stability, since their behavior may depend on prompt formulation, model versioning, and opaque internal representations. Moreover, scalability remains a key challenge when processing large tabular datasets, as commonly encountered in industrial data preparation pipelines. As a result, recent research increasingly explores hybrid solutions that combine LLM-based inference with symbolic constraints, candidate generation mechanisms, and validation steps.

Overall, the three families of approaches described above represent different trade-offs in terms of accuracy, interpretability, supervision requirements, and scalability.

For this reason, practical semantic annotation and data enrichment pipelines often adopt hybrid architectures that integrate heuristic filtering, learned ranking models, language-model-based components, and human-in-the-loop validation mechanisms.

4.4 Experimental Evaluation

In this section, we compare different families of methods for entity linking, focusing in particular on approaches based on machine learning algorithms and large language models. The objective of the evaluation is to assess the level of accuracy achievable by these approaches and to relate it to their operational sustainability in practical data enrichment workflows.[7]

To this end, we consider four tools that are representative of commonly adopted solutions in the literature: Alligator, TURL, TableLlama, and GPT-based approaches. Alligator represents feature-based machine-learning approaches relying on engineered features and ranking models; TURL exemplifies deep learning architectures specifically designed for tabular data; TableLlama represents recent approaches based on instruction-tuned large language models for table understanding; finally, GPT-based systems represent general-purpose language models used through prompt-based inference.

The first three tools are characterized by the availability of information about their training data and execution mechanisms. In contrast, models from the GPT family do not expose details about their internal architecture, training data, or optimization procedures. They therefore behave as black-box systems. At the same time, they represent some of the most powerful general-purpose large language models currently available, and can thus be reasonably expected to provide highly competitive performance.

By comparing these systems, our goal is not only to measure their performance in terms of entity linking accuracy but also to analyze the trade-offs they introduce with respect to computational requirements, reproducibility, and integration within scalable data enrichment pipelines.

4.4.1 Evaluation Datasets

Table 4.1 lists the general-purpose datasets used in the known pre-training phases of the considered approaches. For TableLlama, we use the model released by the authors on HuggingFace, while TURL and Alligator are trained on the same datasets used in their original papers.

For Alligator, the pre-training datasets are derived from the SemTab challenge and are primarily composed of tables synthetically generated by querying KGs. These tables can be large and closely resemble those used in business processes. However,

[7] The content of this section is a revised version from [3].

Table 4.1 Lists of pre-training datasets

Approach	Dataset	Entities	Source	Entities domain
Alligator [1]	SemTab2021 - R2	47.4K	Graph queries	Cross
	SemTab2021 - R3	58.9K	Graph queries	
	SemTab2022 - R2 (2T)	994.9K	Graph queries/web tables	
	SemTab2020 - R4	667.2K	Graph queries	
	SemTab2019 - R3	390.4K	Graph queries	
	SemTab2019 - R1 (T2D)	8K	Graph queries/web tables	
	Total	2.16M		
TURL [10]	TURL WikiTable w/o WikiGS [12]	1.23M	Web Tables	Cross
TableLlama [20]	TableInstruct [20]	2.6M	Web Tables	Cross

Table 4.2 Statistics of the datasets used to fine-tune and evaluate models. †indicates datasets that have been sub-sampled so that they can be entirely processed by TableLlama within its 8192 context

Dataset	Split	Tables	Cols (min | max | $\bar{x}$)	Rows (min | max | $\bar{x}$)	Entities
SemTab2022 2T-red	Train†	91	1 | 8 | 4.86	5 | 369 | 98.61	14674
	Test†	26	1 | 8 | 4.65	7 | 264 | 74.58	4691
SemTab2022 HT-R1	Train	3691	2 | 5 | 2.56	4 | 8 | 5.68	26189
	Test	200	2 | 5 | 2.59	4 | 8 | 5.74	1406
SemTab2022 HT-R2	Train†	4344	2 | 5 | 2.56	4 | 8 | 5.57	20407
	Test	426	2 | 5 | 2.53	4 | 8 | 5.56	1829
SemTab2023 WikidataTables-R1	Train	9917	2 | 4 | 2.51	3 | 11 | 5.65	64542
	Test	500	2 | 4 | 2.46	3 | 11 | 6.95	4247
SemTab2021 BioDiv-red	Test†	11	1 | 26 | 17.45	26 | 100 | 58.90	1232
TURL-120k	Train	13061	1 | 43 | 5.43	1 | 624 | 12.98	120000
TURL-2k-red	Test	1295	1 | 14 | 1.03	6 | 257 | 32.95	1801

they generally lack the rich textual context typically associated with web tables, such as those found on Wikipedia. In contrast, generalistic LLM approaches have been trained primarily on web-based tables [4], which are relatively small, contain rich contextual information, and are crafted by humans for human use.

The term *pre-training* refers to the main training phase performed by the model authors, and we use it to distinguish this phase from the additional *fine-tuning* performed in our experiments. Thus, in these experimental settings, a model may first be pre-trained on one or more datasets, then further fine-tuned on different datasets, and finally evaluated on a third dataset. Table 4.2 reports the statistics of the datasets used for fine-tuning and testing the models.

To evaluate the models' generalization capabilities, we consider the relationship between the data distribution of the datasets used for pre-training (see [3] for details)

Table 4.3 Test datasets categorisation in terms of In-Domain (IN), Out-Of-Domain (OOD), and Moderately-Out-Of-Domain (MOOD) w.r.t. the pre-trained datasets

			Approaches		
Dataset	**Source**	**Domain**	**Alligator**	**TURL**	**TableLlama**
SemTab2022 2T-red	Graph queries/web tables	Cross	IN	MOOD	MOOD
SemTab2022 HT-R1	Graph queries	Cross	IN	MOOD	MOOD
SemTab2022 HT-R2	Graph queries	Cross	IN	MOOD	MOOD
SemTab2023 WikidataTables-R1	Graph queries	Cross	IN	MOOD	MOOD
SemTab2021 BioDiv-red	Biodiversity Tables	Specific	OOD	OOD	OOD
TURL-120k	Web Tables	Cross	MOOD	IN	IN
TURL-2K-red	Web Tables	Cross	MOOD	IN	IN

and that of the datasets used for testing. When the pre-training and test datasets can be regarded as drawn from similar distributions (e.g., when both are constructed from graph queries over cross-domain data), we consider the evaluation to be *in-domain*. When the two distributions differ substantially (e.g., graph queries over cross-domain data versus biodiversity tables with domain-specific languages), we consider the evaluation to be *out-of-domain*. We further introduce a *moderately out-of-domain* setting for cases in which the two distributions are neither clearly similar nor clearly divergent, such as graph queries over cross-domain data versus web tables containing cross-domain data. Observe that GPT-4o models, as well as the backbone LLama model used in TableLlama, may have seen web tables in the test set during the language modeling pre-training phase (for TableLlama: before fine-tuning on tabular tasks). This potential data leakage cannot be controlled in our setting, as in prior experiments, and is orthogonal to the distribution-aware analysis. Table 4.3 reports the testing conditions for the different tools considered.

4.4.2 Evaluation Results

In this section, we evaluate and compare the considered approaches in terms of accuracy. In addition to predictive performance, we also analyse their computational efficiency along two dimensions: (i) execution time, measured as the average wall-clock time required to link a mention (excluding entity retrieval); and (ii) memory usage, measured as peak GPU memory for TURL and TableLlama, and as virtual memory size for Alligator.

All experiments were conducted on the following infrastructure. Testing and fine-tuning for TableLlama and TURL were performed on an Azure NC24ads_a100_v4

instance.[8] Alligator was executed on a machine equipped with an Intel(R) Xeon(R) Gold 6238R CPU @ 2.20GHz with 64 cores and 96GB of RAM.

Table 4.4 reports the accuracy obtained by the evaluated algorithms on the considered datasets. Accuracy values are expressed as "$Accuracy^{DataTypology''}$", where the superscript indicates the relationship between the evaluation dataset and the training data. In particular, $^{\text{IN}}$ denotes *in-domain* data, $^{\text{MOOD}}$ denotes *moderately out-of-domain* data, $^{\text{OOD}}$ denotes *out-of-domain* data, and $^{\text{UNK}}$ unknown data provenance. The superscript $^{\text{MOOD}\rightarrow\text{IN}}$ indicates that the model has been fine-tuned in the MOOD setting, resulting in a transition of the evaluation condition from moderately out-of-domain to in-domain.

Overall, the results show that large language models achieve the highest accuracy across most datasets, with GPT-4o consistently ranking among the top-performing approaches. TableLlama also shows strong performance, confirming the effectiveness of LLM-based approaches specifically tuned for tabular tasks. In contrast, TURL performs well mainly when the evaluation data closely matches its training distribution, while the feature-based Alligator approach remains competitive in several in-domain settings. These results suggest that while LLM-based solutions currently provide the highest accuracy, traditional and feature-based approaches can still offer robust performance when the structure and domain of the data are well aligned with the training conditions.

To complement the accuracy analysis, we also compare the computational characteristics of the considered approaches. Figure 4.3 summarizes the average accuracy, execution time, and memory consumption across the evaluated datasets.

In this figure, the reported time corresponds to the elapsed time required to link a mention, excluding entity retrieval. The occupied memory corresponds to GPU memory for TURL and TableLlama, and to the "Virtual Memory Size" for Alligator. *{Approach}-FT* denotes the respective approach fine-tuned in the MOOD setting.

Overall, the figure highlights a clear trade-off between predictive performance and computational cost. LLM-based approaches achieve the highest average accuracy but require significantly larger memory resources and longer execution times compared with the other methods. In contrast, Alligator exhibits very low computational requirements, running on CPU with limited memory usage, although with lower accuracy in several datasets. TURL represents an intermediate solution, showing moderate computational cost but also a stronger dependency on the similarity between training and evaluation data. These results suggest that the choice of approach should be guided not only by accuracy requirements but also by the available computational resources and the expected deployment scenario.

[8] https://learn.microsoft.com/en-us/azure/virtual-machines/sizes/gpu-accelerated/nca100v4-series?tabs=sizebasic.

Table 4.4 Accuracy of the evaluated algorithms. For each test dataset, the best result is highlighted in **bold**, while the second-best is *underlined*

Dataset	Alligator	TURL	TableLlama	GPT-4o	GPT-4o-mini	Alligator-FT	TURL-FT	TableLlama-FT
SemTab2022 2T-red	0.7156$^{\text{IN}}$	0.1343$^{\text{MOOD}}$	0.8243$^{\text{MOOD}}$	**0.8783**$^{\text{UNK}}$	0.7907$^{\text{UNK}}$	*0.8531*$^{\text{IN}}$	0.3323$^{\text{MOOD}\rightarrow\text{IN}}$	0.8400$^{\text{MOOD}\rightarrow\text{IN}}$
SemTab2022 HT-R1	**0.8890**$^{\text{IN}}$	0.3997$^{\text{MOOD}}$	0.7873$^{\text{MOOD}}$	*0.8542*$^{\text{UNK}}$	0.8087$^{\text{UNK}}$	0.8165$^{\text{IN}}$	0.7454$^{\text{MOOD}\rightarrow\text{IN}}$	0.8001$^{\text{MOOD}\rightarrow\text{IN}}$
SemTab2022 HT-R2	**0.8218**$^{\text{IN}}$	0.2763$^{\text{MOOD}}$	0.6619$^{\text{MOOD}}$	*0.7516*$^{\text{UNK}}$	0.6565$^{\text{UNK}}$	0.7212$^{\text{IN}}$	0.6018$^{\text{MOOD}\rightarrow\text{IN}}$	0.6778$^{\text{MOOD}\rightarrow\text{IN}}$
SemTab2023 WikidataTables-R1	**0.8248**$^{\text{IN}}$	0.3391$^{\text{MOOD}}$	0.7426$^{\text{MOOD}}$	0.8034$^{\text{UNK}}$	0.7554$^{\text{UNK}}$	*0.8086*$^{\text{IN}}$	0.7061$^{\text{MOOD}\rightarrow\text{IN}}$	0.7530$^{\text{MOOD}\rightarrow\text{IN}}$
SemTab2021 BioDiv-red	0.5674$^{\text{OOD}}$	0.8109$^{\text{OOD}}$	0.9513$^{\text{OOD}}$	**0.9748**$^{\text{UNK}}$	0.9497$^{\text{UNK}}$	0.8547$^{\text{OOD}}$	0.6347$^{\text{OOD}}$	*0.9610*$^{\text{OOD}}$
TURL-2K-red	0.6017$^{\text{MOOD}}$	0.7118$^{\text{IN}}$	*0.9051*$^{\text{IN}}$	**0.9095**$^{\text{UNK}}$	0.8629$^{\text{UNK}}$	0.7456$^{\text{MOOD}\rightarrow\text{IN}}$	0.5347$^{\text{IN}}$	0.9045$^{\text{IN}}$

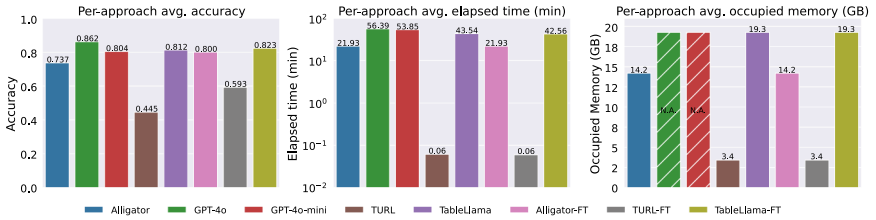


Fig. 4.3 Average accuracy, elapsed time (min), and memory usage (GB)

4.4.3 Sustainability

Beyond predictive performance, data enrichment approaches should also be evaluated in terms of their sustainability. In practice, the preparation and enrichment of tabular data may require substantial computational resources, human effort, and interaction with external services. These aspects become particularly relevant when enrichment pipelines operate at scale or must be repeatedly executed on evolving datasets.

Sustainability in data enrichment should account for the balance between improvements in data quality and the resources required to achieve them. Enrichment workflows typically involve two types of activities: *assessment*, aimed at identifying missing or erroneous values in the input data, and *improvement*, aimed at completing or correcting them.

Assessment costs quantify the effort needed to analyse the dataset and prepare enrichment actions, including understanding external sources and configuring matching processes. *Improvement costs* measure the resources required to execute the enrichment itself, such as performing matching and retrieving additional information. Since not all enrichment actions are fully effective, the portion of resources spent on unsuccessful operations or remaining errors is captured by *waste costs*.

The key characteristic of reconciliation-based enrichment pipelines–namely, linking table values to shared identifiers and using them to retrieve only the relevant information from external sources–reduces the need for extensive data discovery, schema understanding, and pairwise matching across heterogeneous datasets, which are typically the most resource-intensive phases of traditional data integration.

The comparison reported in Table 4.5 illustrates these trade-offs for the enrichment task described in Sect. 4.2. The table reports the application of the entity linking approaches discussed in Sect. 4.4, together with the specialized HERE API service, to link city names in the digital marketing dataset with geographic coordinates. For comparison, we also evaluated LLMs in a pairwise matching setting, reporting the corresponding results using the F1[9] measure.

The evaluation follows the sustainability model introduced in [16], which assesses data preparation strategies by jointly considering their effectiveness and the resources required to achieve the improvement. Accordingly, the table reports the parameters

[9] F1 score is the harmonic mean of precision and recall, providing a single measure of a method's accuracy in binary or multi-class classification tasks.

Table 4.5 Comparison of Methods, Assessment and Improvement Costs, and Waste. *Source* [16]

Method type	Matching method	LLM	Meas.	$p_{j,k}$	$c_{j,k}$ x1000	Ass. costs	Impr. costs	Total costs	Waste
Pairwise	GPT4o	✓	F1	0.8774	12.125	120	3891	4011	477
Pairwise	GPT4o-mini	✓	F1	0.8084	0.728	120	233	353	44
Reconc.	HERE	✗	Acc	0.9980	0.300	0	96	96	0
Reconc.	Alligator + Geo	✗	Acc	0.8000	0.266	0	85	85	17
Reconc.	TableLlama + Geo	✓	Acc	0.8230	1.367	0	438	438	77
Reconc.	GPT4o + Geo	✓	Acc	0.8620	5.200	0	1668	1668	230
Reconc.	GPT4o-mini + Geo	✓	Acc	0.8040	0.312	0	100	100	19

$p_{j,k}$ and $c_{j,k}$, together with the derived cost indicators. The parameter $p_{j,k}$ denotes the effectiveness of method k for quality dimension j, i.e., the proportion of erroneous or incomplete values that are successfully corrected or enriched by the method. The parameter $c_{j,k}$ represents the unit cost associated with performing the corresponding improvement action, capturing computational resources, external service usage, and human effort when applicable. The values of $c_{j,k}$ are based on assumed standard costs and should be interpreted for comparative purposes rather than as absolute measures.

As expected, the results indicate that reconciliation strategies differ significantly in their sustainability profiles. Lightweight methods and specialized APIs typically require minimal computational resources, offering a favourable trade-off between effectiveness and cost. In contrast, large language models achieve very high matching performance but impose substantially higher computational demands, which may limit their feasibility in large-scale or resource-constrained settings. Overall, reconciliation incurs lower costs than traditional integration methods such as pairwise matching.

These findings suggest that reconciliation-based enrichment represents a promising approach for sustainable data preparation pipelines. By leveraging shared identifiers and interoperable knowledge sources, these methods reduce redundant processing and enable more efficient reuse of existing information infrastructures.

4.5 Conclusions

Reconciliation-based data enrichment is a data preparation technique that does not require direct control or prior integration of the third-party datasets to be used. When such datasets expose access interfaces–such as APIs, SPARQL, or SQL endpoints–it is sufficient to reconcile the source data, i.e., link them to the same identifier system adopted by those datasets, in order to retrieve the information of interest.

The main limitation of reconciliation and linking techniques lies in the intrinsic ambiguity of the natural language expressions used in tabular data. For this reason, a variety of approaches–ranging from heuristic methods to machine learning and

language models–have been developed. As discussed in this chapter, these techniques achieve good levels of accuracy while exhibiting different cost–performance profiles.

The evaluation highlights several trade-offs among current approaches to entity linking in tabular data. Extra-large API-based LLMs show strong generalization capabilities and consistently achieve the highest accuracy across datasets, albeit at substantially higher computational and economic costs. Medium-sized API models such as GPT-4o-mini provide an effective compromise, reaching performance levels close to larger models while maintaining a more favourable accuracyâŁ“cost balance. Non-LLM approaches remain relevant in specific scenarios: when the structure and domain of the data align with the assumptions of the method, they can offer a competitive trade-off between effectiveness and resource consumption.

Recent reasoning-oriented models suggest that higher accuracy may be achievable with fewer parameters; however, current implementations suffer from very large inference times, which may limit their applicability in practical pipelines. Finally, the experiments highlight the importance of task-specific training: matching the performance of a fine-tuned 7B model may require a general-purpose model of roughly 70B parameters when no task-specific training is available.

Acknowledgements This research was partially funded by the European Innovation Actions projects *enRichMyData* (HE 101070284) and *DataPACT* (HE 101189771).

References

1. Avogadro, R., Ciavotta, M., De Paoli, F., Palmonari, M., Roman, D.: Estimating link confidence for human-in-the-loop table annotation. In: 2023 IEEE/WIC International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), pp. 142–149 (2023)
2. Batini, C., Scannapieco, M.: Data and Information Quality - Dimensions, Principles and Techniques. Data-Centric Systems and Applications. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-24106-7. https://doi.org/10.1007/978-3-319-24106-7
3. Belotti, F., Cremaschi, M., Dadda, F., Avogadro, R., Palmonari, M.: How good are LLMs in disambiguating entities in tabular data? A comprehensive study. Data & Knowledge Engineering, p. 102596 (2026). https://doi.org/10.1016/j.datak.2026.102596. https://www.sciencedirect.com/science/article/pii/S0169023X26000431
4. Bhagavatula, C.S., Noraset, T., Downey, D.: Tabel: entity linking in web tables. In: International Semantic Web Conference, pp. 425–441. Springer (2015)
5. Bizer, C.: The emerging web of linked data. IEEE Intell. Syst. **24**(5), 87–92 (2009). https://doi.org/10.1109/MIS.2009.102
6. Ciavotta, M., Cutrona, V., De Paoli, F., Nikolov, N., Palmonari, M., Roman, D.: Supporting semantic data enrichment at scale. In: Technologies and Applications for Big Data Value, pp. 19–39. Springer (2022)
7. Cremaschi, M., Avogadro, R., Chieregato, D.: s-elBat: a semantic interpretation approach for messy tables. In: Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab@ISWC), CEUR-WS. org, pp. 59–71. (2022)
8. Cutrona, V., De Paoli, F., Košmerlj, A., Nikolov, N., Palmonari, M., Perales, F., Roman, D.: Semantically-enabled optimization of digital marketing campaigns. In: Ghidini, C., Hartig, O., Maleshkova, M., Svátek, V., Cruz, I.F., Hogan, A., Song, J., Lefrançois, M., Gandon, F. (eds.)

The Semantic Web - ISWC 2019–18th International Semantic Web Conference Proceedings, Part II, pp. 345–362. Springer, Cham (2019)
9. De Paoli, F., Ciavotta, M., Avogadro, R., Hristov, E., Borukova, M., Petrova-Antonova, D., Krasteva, I.: An interactive approach to semantic enrichment with geospatial data. Data & Knowl. Eng. **153**, 102341 (2024)
10. Deng, X., Sun, H., Lees, A., Wu, Y., Yu, C.: Turl: table understanding through representation learning. ACM SIGMOD Rec. **51**(1), 33–40 (2022)
11. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (2019)
12. Efthymiou, V., Hassanzadeh, O., Rodriguez-Muro, M., Christophides, V.: Matching web tables with knowledge base entities: from entity lookups to entity embeddings. In: The Semantic Web – ISWC 2017: 16th International Semantic Web Conference. Vienna, Austria, 21–25 Oct 2017, Proceedings, Part I, pp. 260–277. Springer-Verlag, Berlin, Heidelberg (2017)
13. Jiménez-Ruiz, E., Hassanzadeh, O., Efthymiou, V., Chen, J., Srinivas, K.: Semtab 2019: resources to benchmark tabular data to knowledge graph matching systems. In: European Semantic Web Conference, Springer (2020)
14. Liu, J., Chabot, Y., Troncy, R., Huynh, V.P., Labbé, T., Monnin, P.: From tabular data to knowledge graphs: a survey of semantic table interpretation tasks and methods. J. Web Semant. **76**, 100761 (2023)
15. Mendes, P.N., Jakob, M., García-Silva, A., Bizer, C.: DBpedia spotlight: shedding light on the web of documents. In: Proceedings of the 7th International Conference on Semantic Systems, I-Semantics '11, p. 1–8. Association for Computing Machinery, New York, NY, USA (2011). https://doi.org/10.1145/2063518.2063519. https://doi.org/10.1145/2063518.2063519
16. Pernici, B., Cappiello, C., Bono, C., Sancricca, C., Catarci, T., Angelini, M., Filosa, M., Palmonari, M., Paoli, F.D., Bergamaschi, S., Simonini, G., Mozzillo, A., Zecchini, L.: Sustainable quality in data preparation. ACM J. Data Inf. Qual. **17**(4:23), 1–33 (2025)
17. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C.C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Koura, P.S., Lachaux, M.A., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E.M., Subramanian, R., Tan, X.E., Tang, B., Taylor, R., Williams, A., Kuan, J.X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S., Scialom, T.: Llama 2: Open foundation and fine-tuned chat models (2023). https://arxiv.org/abs/2307.09288
18. Volz, J., Bizer, C., Gaedke, M., Kobilarov, G.: Discovering and maintaining links on the web of data. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) The Semantic Web - ISWC 2009, pp. 650–665. Springer, Berlin Heidelberg, Berlin, Heidelberg (2009)
19. Wei, J., Bosma, M., Zhao, V., Guu, K., Yu, A.W., Lester, B., Du, N., Dai, A.M., Le, Q.V.: Finetuned language models are zero-shot learners. In: International Conference on Learning Representations, (2022)
20. Zhang, T., Yue, X., Li, Y., Sun, H.: TableLlama: towards open large generalist models for tables. In: Duh, K., Gómez-Adorno, H., Bethard, S. (eds.) Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024, pp. 6024–6044. Association for Computational Linguistics (2024)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (http://creativecommons.org/licenses/by-nc-nd/4.0/), which permits any noncommercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if you modified the licensed material. You do not have permission under this license to share adapted material derived from this chapter or parts of it.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Chapter 5
Lightweight Pipelines: Good Enough is Sometimes Better



Camilla Sancricca and Cinzia Cappiello

Abstract Data-centric AI prioritizes data-centric strategies for providing high-quality input data, since high data quality is fundamental for ensuring the reliability of AI systems. It recognizes the critical role of data preparation, an often time-consuming task that is a significant challenge for data scientists. While various methods have been proposed to support and automate data preparation, most focus on optimizing model performance without considering the high energy and resource requirements for computing complex pipelines on large-scale datasets. This chapter introduces an approach to designing lightweight data preparation pipelines: it selectively addresses only the data quality issues that significantly impact analysis performance. We present a framework that leverages historical knowledge from previous pipelines to recommend an effective sequence of preparation actions for a specific analysis, prioritizing efficiency while preserving quality. Our preliminary experiments show that improving only the most impactful data quality errors yields high performance while significantly reducing time and resources.

5.1 Introduction: Data-Centric Green AI

A variety of methods have been proposed in recent years to support data preparation. Nonetheless, these methods are frequently applied to large-scale datasets and rely on complex ML-based techniques to transform and clean the data. As a result, data preparation pipelines can become highly resource-demanding, requiring substantial computational power, consuming large amounts of energy, and ultimately contributing to increased gas emissions.

To mitigate these issues, the concept of green AI [3] has been introduced, emphasizing energy efficiency and sustainability alongside predictive performance. More recently, a few studies have begun exploring data-centric strategies to support green

C. Sancricca · C. Cappiello (✉)
Politecnico di Milano—DEIB, Milano, Italy
e-mail: cinzia.cappiello@polimi.it

C. Sancricca
e-mail: camilla.sancricca@polimi.it
© The Author(s) 2027

B. Pernici et al. (eds.), *Discount Quality for Responsible Data Science*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-032-29481-4_5

AI [1], showing that sustainability can often be achieved without significant, or even any, deterioration in model accuracy [4]. This is made possible by designing lightweight pipelines through careful data selection, reduced dataset sizes, or the application of feature selection and dimensionality reduction techniques.

In this chapter, we explore how we can contribute to the design and development of lightweight data preparation pipelines. We started by presenting a framework that prioritizes preparation actions based on their impact on model performance and identifies and addresses the most influential Data Quality (DQ) issues.

Preliminary findings demonstrate the potential to leverage this framework's recommendations to reduce energy consumption. Specifically, we show that addressing only the most impactful DQ dimension can yield ML performance comparable to the ones obtained by executing the full preparation pipeline, while significantly reducing computational effort and promoting more sustainable pipelines.

The framework is presented in Sect. 5.2, while Sect. 5.3 discusses preliminary experimental results that evaluate the effectiveness and sustainability of this approach.

5.2 A Framework for Data-Centric Green AI

This section presents our approach for a**d**apt**i**ve d**a**ta-ce**n**tric **AI** (DIANA) [2], a framework able to support and guide users in selecting and validating the data preparation tasks to perform in a data analysis pipeline. The pipeline suggestion mechanism of DIANA is driven by the content of a knowledge base (KB) in which we collected results of experiments performed on a large number of analysis contexts and related to (i) the impact of DQ errors and (ii) the effectiveness of several data preparation techniques for a variety of different data profile/ML model combinations (i.e., analysis contexts). Such impact and effectiveness are expressed in terms of ML model performance (e.g., F1 score, precision, and recall). Note that we focus only on classification tasks.

Figure 5.1 shows DIANA's general functioning, including the main steps of the data science pipeline. DIANA starts with the *Discovery and Audit* step, in which the input dataset is analyzed, and profiling metadata are provided. This phase allows users to check the dataset's content, providing basic statistics and interactive visualizations on value distributions, correlations, or missing values. Moreover, it assesses the levels of several DQ aspects (completeness, accuracy, consistency) and identifies DQ issues that could compromise data analysis results. The results of these phases should help users understand the datasets' content and their initial suitability for the task at hand. At this stage, the user also needs to enter the type of analysis to be performed in the *goal definition*.

The targeted analysis and the computed *data profile* (the new user's analysis context) are provided as input to the KB, which is responsible for generating an optimal sequence of data preparation actions (along with their suggested execution order) to maximize the quality of the results.

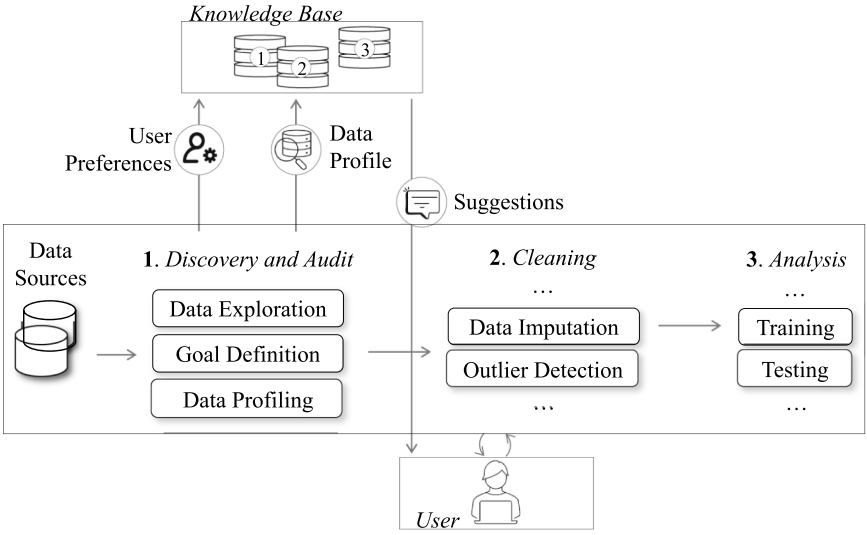


Fig. 5.1 DIANA's general architecture

As stated above, DIANA's KB includes results from previous pipelines, computed across several datasets, related to the impact of DQ errors and the effectiveness of data preparation techniques on ML models. The KB exploits the information on the collected data profiles that are most similar to the user's dataset and provides (i) the order in which the DQ dimensions should be improved and, for each DQ dimension, (ii) a suggestion about the best improvement technique to apply for each column.

The KB's recommendations are sent to the data *cleaning* phase, which executes the DQ improvement tasks. At the end of the cleaning phase, the user can proceed with the *analysis*, i.e., all the steps required to execute the ML model.

5.2.1 The Knowledge Base Design

As stated above above, the empirical content of KB includes: [KB_1] impact of DQ errors (related to a DQ dimension, e.g., missing values for completeness, outliers for accuracy) on ML performance; [KB_2] effectiveness of applying several data imputation techniques on ML performance; [KB_3] effectiveness of several outlier detection techniques in detecting outliers. The execution of the experiments for enriching the KB_1 was performed along the following phases:

1. *Feature Selection* We kept only the three most influential columns employing a univariate statistical test to identify features most correlated to the target column.

2. *Dataset Pollution* We created different versions of the original dataset, injecting a different percentage (from 10% to 50%, with an increasing step of 10%) of one type of DQ error. To inject the errors, we followed a uniform distribution.
3. *Profile Extraction* We computed the dataset and column profiles for the considered version of the dataset.
4. *Data Preparation* For each polluted version, we applied different data preparation techniques to improve the injected errors.
5. *Results Extraction* We executed the selected ML algorithms on the final dataset version and extracted the performance. The polluted/cleaned version is split into train and test sets (70% and 30%); the train set is used to fit the model, while the test set is used for prediction via 4-fold cross-validation. We tuned the hyperparameters on the original dataset, and the hyperparameters that achieved higher performance were used for all computations.
6. *Results Storage* We stored the experiment results as follows: [KB_1] ratio between ML performance achieved by the polluted version and the original dataset; [KB_2] ratio between ML performance achieved by the final cleaned version and the original dataset; [KB_3] percentage of detected outliers.

Setup. We performed the experiments considering two DQ dimensions, Completeness and Accuracy, thus injecting missing values and outliers as DQ errors and applying 12 data imputation and 6 outlier detection techniques as data preparation techniques (both standard and ML-based techniques were considered). We employ a total of 31 datasets with heterogeneous profiling characteristics taken from the UCI Machine Learning and Kaggle repositories.

Six classification algorithms were performed using the F1 score as a performance indicator: Decision Tree (DT), Logistic Regression (LR), K-Nearest Neighbors (KNN), Random Forest (RF), AdaBoost (AB), and Support Vector Classifier (SVC) implemented by the *scikit-learn* library.

Details on the datasets used to enrich the three KBs, the entire set of profiling metadata extracted for dataset and column profiles, and the complete list of data preparation tasks are recorded in our GitHub repository.[1]

The KB based its suggestions on three ML-based predictors, which are regression models (*KNNRegressor* models, with different hyperparameters depending on the prediction task) with the following characteristics:

P1 *Trained on*: [KB_1]. *Input*: a data profile and an ML model. *Output*: expected impact of a DQ dimension on the analysis performance.
P2 *Trained on*: [KB_2]. *Input*: a column profile and an ML model. *Output*: expected effectiveness (ML performance) of applying a data imputation task on that column.
P3 *Trained on*: [KB_3]. *Input*: a column profile. *Output*: expected effectiveness of applying an outlier detection task on that column.

Our predictors had an average root mean squared error of at most 13.44% (for the first predictor), while the other errors were even smaller. Moreover, we were able

[1] https://github.com/camillasancricca/diana-repo.git.

to predict the correct repair pipeline for 74,13% of the tested analysis contexts. We were always able to predict the most relevant methods to apply for each column.

By sorting the predictor results, we can suggest an optimal data preparation pipeline for a specific analysis context.

5.3 Preliminary Results on Lightweight Pipelines

Focusing on correcting only the most influential DQ dimensions in a given analysis context (while leaving less relevant errors unaddressed) can lead to shorter, more efficient data preparation pipelines. Our experiments show that this strategy preserves high levels of ML performance while significantly reducing computation time and resource consumption.

The goal of this evaluation is now to confirm that applying only the first recommended DQ improvement technique provides nearly the same result as executing the entire pipeline.

5.3.1 Experiment Pipeline and Metrics

We employed six new datasets from Kaggle[2] with different dimensionalities (number of tuples) and data types (numerical/categorical columns) to validate our approach: *weather*, *consumer*, *pet*, *character*, *galaxy*, and *heart*.

We kept only the three most influential columns for classification, based on the mutual information scores between each column and the target. More details are available in our public repository. For each dataset, we performed the following pipeline.

We polluted the dataset by creating different versions of the original dataset, each with a different error percentage (from 10% to 50%), with errors uniformly distributed. For each polluted version-ML model:

1. We computed the dataset/column profiles and extracted the suggested pipeline from our predictors.
2. We executed the suggested pipeline and all the other possible pipelines on the polluted dataset.
3. The ML model is executed on the resulting cleaned dataset. It is split into train and test sets (70% and 30%). The train set is used to fit the model, while the test set is used for prediction via 4-fold cross-validation. The model hyperparameters were tuned on the original dataset and used for all computations. The performance of the ML model is extracted *each time a DQ dimension is improved*.

[2] https://www.kaggle.com.

All the possible pipelines were applied to the polluted dataset. First, the DQ dimensions were improved in different orders: completeness → accuracy, and accuracy → completeness. Then, all possible combinations of data preparation tasks were applied to the dataset's columns in all possible orders.

For the completeness → accuracy pipeline: (a) a specific combination of imputation techniques is applied to the dataset's columns; (b) a combination of outlier detection techniques is applied to the numerical columns and the identified outliers are set to null; (c) the same imputation techniques applied in (a) are executed again to fill the null values. For the accuracy → completeness pipeline, only (b) and (c) are executed.

We performed a total of 176, 580 pipelines for datasets with three numerical columns, 89, 211 pipelines for datasets with two numerical columns, and 22, 440 pipelines for datasets with one numerical column.

To support this comparison, we define a performance ratio metric, denoted as ΔQ, which quantifies the relative change in model performance after one or more DQ improvements. Specifically, given a dataset version d_j, we compute the performance p_i of an ML model after applying the first-ranked DQ improvement (d_1) and after completing all planned improvements (e.g., d_1 and d_2). The metric is formalized as:

$$\Delta Q_i = \frac{p_i}{p_o}$$

where p_o denotes the model's baseline performance on the original, clean dataset d_o, and p_i is the performance after step i of the improvement process ($i = 1$ for the first dimension only, $i = 2$ for the complete sequence).

5.3.2 *Results and Discussion*

Preliminary results obtained by applying the validation pipeline for all tested dataset–ML model combinations are presented as follows.

Table 5.2 reports the median values of ΔQ_1 and ΔQ_2, computed across all error injection levels, for every dataset-model combination. Specifically, ΔQ_1 reflects the performance after applying only the first DQ improvement, while ΔQ_2 reflects the performance after applying the full data preparation pipeline. The results are grouped by pipeline type (either the *suggested* pipeline derived from our predictors, or *all* possible alternative pipelines). Aggregated medians across all datasets are included at the bottom of the table. In addition, Fig. 5.3 visualizes the distribution of ΔQ_1 and ΔQ_2 values for each model, again separating suggested pipelines from the rest.

The table reveals that the pipeline recommended by our approach consistently achieves higher ΔQ scores (at both stages) than any alternative combination. The only exception is ΔQ_1 for AdaBoost, although even in that case, the suggested pipeline yields a better ΔQ_2. This confirms the *effectiveness* of our prediction mechanism. The trend is clearly reinforced by Fig. 5.3, where the distribution of ΔQ

		DT		*LR*		*KNN*	
		ΔQ_1	ΔQ_2	ΔQ_1	ΔQ_2	ΔQ_1	ΔQ_2
weather	*suggested*	0.8388	0.836	0.8552	0.8698	0.8396	0.8405
	all	0.815	0.78	0.5592	0.657	0.7651	0.7819
galaxy	*suggested*	0.9802	0.9776	1.1107	1.1104	0.9696	0.9715
	all	0.9284	0.9011	0.7415	0.9142	0.8545	0.869
character	*suggested*	0.7926	0.813	0.9476	0.9456	0.8234	0.8283
	all	0.8039	0.7936	0.8714	0.8752	0.8195	0.8076
consumer	*suggested*	0.9441	0.9385	1.0049	0.9977	0.9361	0.9325
	all	0.9268	0.9113	0.9213	0.9171	0.9229	0.9161
pet	*suggested*	0.8223	0.8389	0.994	0.9924	0.8761	0.8853
	all	0.8178	0.8095	0.9761	0.9572	0.882	0.8567
heart	*suggested*	0.815	0.8089	0.9427	0.9269	0.8488	0.8107
	all	0.8187	0.8047	0.9363	0.9504	0.8494	0.83
MEDIAN (all data)	*suggested*	0.8306	0.8375	0.9708	0.969	0.8625	0.8629
	all	0.8183	0.8071	0.8964	0.9157	0.852	0.8434

		RF		*AB*		*SVC*	
		ΔQ_1	ΔQ_2	ΔQ_1	ΔQ_2	ΔQ_1	ΔQ_2
weather	*suggested*	0.8438	0.841	0.8561	0.8531	0.4609	0.5865
	all	0.8385	0.8124	0.8493	0.8316	0.3143	0.3549
galaxy	*suggested*	0.9735	0.9805	0.9627	0.9735	1.2328	1.2403
	all	0.9391	0.9207	0.9423	0.918	1.0045	1.0723
character	*suggested*	0.7984	0.7997	0.8514	0.8449	0.98	0.9805
	all	0.7962	0.792	0.8511	0.8305	0.9466	0.9588
consumer	*suggested*	0.921	0.9239	0.9348	0.9292	0.6352	0.6315
	all	0.908	0.8993	0.9427	0.9342	0.5804	0.5685
pet	*suggested*	0.8279	0.8258	0.8275	0.8279	1.0143	0.9912
	all	0.8172	0.8006	0.8768	0.862	1.0146	0.9902
heart	*suggested*	0.8224	0.7884	0.8215	0.7671	1.0343	1.0334
	all	0.8165	0.7894	0.8293	0.8349	1.0173	1.0188
MEDIAN (all data)	*suggested*	0.8359	0.8334	0.8538	0.849	0.9972	0.9859
	all	0.8279	0.8065	0.864	0.8485	0.9756	0.9745

Fig. 5.2 Median ΔQ_1 and ΔQ_2 results for different combinations of dataset and algorithms, such as Decision Tree (DT), Logistic Regression (LR), K-Nearest Neighbors (KNN), Random Forest (RF), AdaBoost (AB), and Support Vector Classifier (SVC)

values for suggested pipelines is consistently skewed toward the upper end of the violin plots.

Another important observation is that ΔQ_1 values are *always close to, and in many cases exceed,* the corresponding ΔQ_2 values. Thus, the performance obtained by correcting only the most influential DQ dimension is nearly equivalent (or even higher) to the one achieved by executing the full pipeline. This behavior is again reflected in the overlapping distributions shown in Fig. 5.3.

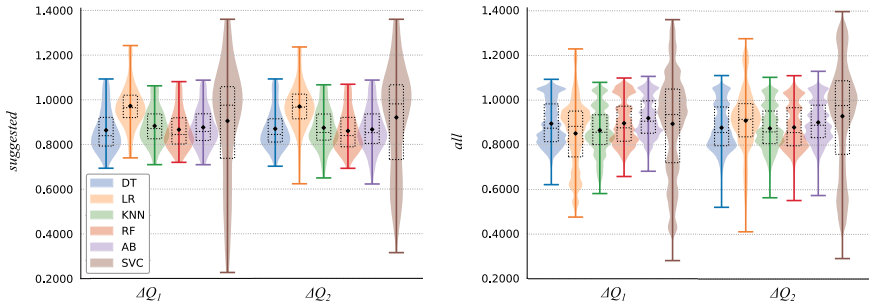


Fig. 5.3 ΔQ_1 and ΔQ_2 distributions for the following algorithms: Decision Tree (DT), Logistic Regression (LR), K-Nearest Neighbors (KNN), Random Forest (RF), AdaBoost (AB), and Support Vector Classifier (SVC)

These findings suggest that fixing the most important DQ issue provides a performance improvement that is almost the same as that achieved by computing the whole pipeline, while correcting additional, lower-impacting issues has a limited impact. In fact, in certain cases, addressing less influential dimensions can *degrade* performance, likely because these corrections inject approximate values that modify patterns the model had already learned to tolerate.

Ultimately, these results demonstrate the potential *sustainability* of DIANA approach: applying only the most impactful DQ improvement, as identified by our approach, yields performance comparable to (or better than) completing the full pipeline. This allows substantial savings in time, computation, and energy, making the data preparation process more efficient and sustainable while maintaining high quality of the results.

References

1. Salehi, S., Schmeink, A.: Data-centric green artificial intelligence: a survey. IEEE Trans. Artif. Intell. **5**(5), 1973–1989 (2023)
2. Sancricca, C.: DIANA: A knowledge-driven framework for Data-centric AI. In: EDBT/ICDT Workshops, (2024)
3. Schwartz, R., Dodge, J., Smith, N.A., Etzioni, O.: Green AI. Commun. ACM **63**(12), 54–63 (2020)
4. Verdecchia, R., Cruz, L., Sallou, J., Lin, M., Wickenden, J., Hotellier, E.: Data-centric green AI an exploratory empirical study. In: International Conference on ICT for Sustainability (ICT4S), pp. 35–45. IEEE, Piscataway, NJ (2022)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (http://creativecommons.org/licenses/by-nc-nd/4.0/), which permits any noncommercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if you modified the licensed material. You do not have permission under this license to share adapted material derived from this chapter or parts of it.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Chapter 6
Human Factors in Data Preparation



Marco Angelini, Matteo Filosa, and Tiziana Catarci

Abstract Data preparation is a cornerstone of the data science lifecycle, yet it remains one of the most time-consuming and human-dependent stages. Human factors are critical here, shaping how users interact with tools, perceive data quality, and make trade-offs under uncertainty. At the crossroads of Human–Computer Interaction (HCI), Visualization, and human-in/over-the-loop paradigms, recent research highlights new opportunities to make data preparation more usable, transparent, and trustworthy. Interactive interfaces and visual analytics systems allow users to iteratively refine transformations, explore provenance, and detect anomalies, while cognitive aspects such as workload, bias, and trust determine the reliability of human interventions. Automation increasingly complements these workflows: human-in-the-loop methods leverage user feedback to guide cleaning, labeling, or schema alignment, and over-the-loop frameworks provide oversight mechanisms to prevent over-reliance on automation. At the same time, modern generative AI introduces powerful new capabilities, such as automatically suggesting transformations, synthesizing metadata or documentation, and even generating realistic synthetic datasets. These advances promise efficiency and creativity but raise concerns about reliability, explainability, and provenance, reinforcing the need for careful human oversight. This chapter reviews these possibilities and the research challenges in developing a new class of hybrid, adaptive data-preparation solutions that blend human expertise with automated intelligence. Key challenges include balancing scalability with user effort, ensuring transparency in automated suggestions, preventing loss of situational awareness, and developing evaluation metrics that reflect both human expe-

M. Angelini
Link University of Rome, Via del Casale di San Pio V 44, Roma, Italy
e-mail: m.angelini@unilink.it

M. Filosa · T. Catarci (✉)
Sapienza Università di Roma DIAG, Roma, Italy
e-mail: catarci@diag.uniroma1.it; tiziana.catarci@istc.cnr.it

M. Filosa
e-mail: m.filosa@diag.uniroma1.it

T. Catarci
ISTC-CNR, Roma, Italy

© The Author(s) 2027

B. Pernici et al. (eds.), *Discount Quality for Responsible Data Science*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-032-29481-4_6

rience and technical accuracy. Addressing these challenges will enable more robust, human-aware data preparation practices across domains.

6.1 Introduction

When designing any kind of software, it is important to follow specific guidelines that account for human factors to deliver high-quality artifacts. The discipline leading this aspect is Human–Computer Interaction (HCI). It is a multidisciplinary field that studies how people design, use, and experience interactive computer systems, with the goal of making technology both effective and usable for its intended users [24, 65]. Research in HCI focuses on the design, implementation, and evaluation of interactive systems by drawing on computer science, psychology, cognitive science, design, ergonomics, and the social sciences [2, 24]. Foundational work framed HCI as the study of the human information processor interacting with computer interfaces, using models such as GOMS and keystroke-level modeling to predict user performance and inform interface design [15]. Over time, the field has expanded from low-level performance modeling to broader concerns, including user experience, collaboration, accessibility, and the societal impact of digital technologies [24, 65]. Within this broader perspective, HCI provides theories and methods to systematically account for human factors when designing interactive systems: for example, how perception, attention, memory, and motor capabilities shape the usability and learnability of interfaces [15, 24]. Classic HCI textbooks and design treatises synthesize these principles into concrete design rules, emphasizing consistency, feedback, affordances, and error prevention as core to creating interfaces that are predictable, controllable, and supportive of users' mental models [53, 65]. Canonical models and guidelines have been integrated into formal curricula, thereby establishing HCI as a discipline centered on understanding users, tasks, and contexts to design, prototype, and evaluate interactive systems [2].

In the context of data preparation, these HCI foundations are crucial for designing tools that respect human cognitive limits, support sensemaking under uncertainty, and foster appropriate levels of trust and control over automated transformations [27, 70]. Data preparation itself encompasses a structured pipeline of activities that transform raw, heterogeneous data into a format suitable for analysis and modeling. Core stages include data extraction, often a major bottleneck due to scale and source variety, requiring a mix of automated scraping and human validation [20, 28, 60], followed by integration to unify disparate sources [26]. Subsequent profiling examines data structure, content, and relationships to uncover issues like missing values or inconsistencies [1, 19], paving the way for intensive wrangling that cleans, imputes, transforms, and enriches data [31, 38]. Augmentation then expands datasets to enhance model robustness [59], while labeling and final quality checks ensure readiness for downstream tasks [37]. These steps are computationally intensive and error-prone, with data wrangling alone often consuming a large share of data science

effort, underscoring the need for human-aware designs that balance automation with targeted oversight.

At the same time, Visual Analytics (VA) has become a prominent discipline for supporting data analysis across vast scientific domains, often positioned at the final phase of the data preparation pipeline. Designing an optimized VA solution for the pipeline is no easy task, as many human factors must be considered. An example is latency, which can hinder the effective human analysis of critical data [9, 10], or granularity, which directly impacts server and DBMS usage (in terms of computation time, energy consumption, and CO_2 emissions) [58]. Several studies have conducted in-depth analyses of human interaction in big data visualization systems [27, 70]. Restricting the focus to data preparation, only a limited number of works address these issues through visual methods and explicit human involvement, with particular emphasis on data quality.

Recent advances in large language models (LLMs) have introduced specialized data preparation pipelines that amplify these challenges while highlighting opportunities for hybrid approaches. LLM datasets undergo extensive cleaning, normalization, deduplication, filtering, and strategic mixing, as seen in pipelines for BLOOM [11], LLaMA 3 [43], and Orion-14B, where human heuristics guide thresholds and domain selection [3, 46]. Generative AI further automates wrangling through natural-language prompts for transformations and imputation [82], yet introduces risks of hallucinated fixes, provenance tracking, and bias amplification that demand human review. Visualization plays a key role in LLM data curation, enabling analysts to inspect filtering outcomes and sample distributions interactively, much like in general pipelines, but at scale for massive corpora. This evolution reinforces the focus on human factors, as LLM data preparation trades raw scale for heightened demands for explainability, trust calibration, and situational awareness in quality steering. Addressing these pipelines' human dependency, recent work advocates human-in-the-loop (HITL) and visual analytics to make data preparation sustainable and effective. Rather than deferring human intervention to post-modeling feedback loops, HITL is inserted early, at extraction validation, wrangling decisions, and profiling interpretation, using think-aloud sessions and expert oversight to capture domain intent [18, 25]. Visual tools mediate this by rendering quality metrics, imputation previews, and pipeline flows, reducing cognitive load while enabling steering [7, 40, 69]. Patterns emerge: selective intervention at high-uncertainty stages (i.e., imputation choice, label verification), feedback loops that log corrections to refine automation, and visualizations that support fluency in iterative exploration [41]. Studying how users interact with visualization systems is crucial to minimizing disruption and errors in their workflows, leading to increasingly optimized visualization environments for data preparation experts. Despite these efforts, data preparation itself has received comparatively little attention in this line of research, leaving a clear gap in the literature that this chapter aims to address by systematically framing human factors in data preparation across HCI, VA, and human-/over-the-loop paradigms.

The remainder of this chapter is organized into two main parts. Section 6.2 and Sect. 6.3 develop a conceptual characterization of human factors in data prepara-

tion and outline open research challenges. Section 6.4 then illustrates these concepts through concrete systems and scenarios that tackle data quality issues in practice.

6.2 Characterization of Human Factors in the Data Preparation Pipeline

Evaluating human factors in data preparation requires adapting and combining metrics that originated in HCI, usability research, and visualization, and reinterpreting them for interactive data tools. In HCI, usability has traditionally been defined in terms of effectiveness, efficiency, and satisfaction, as formalized, for example, in ISO 9241-11 and related usability frameworks [35, 51]. These dimensions can be operationalized for data preparation interfaces by measuring how accurately users complete cleaning or transformation tasks (effectiveness), how quickly and with how many interactions they achieve their goals (efficiency), and how they rate their experience in terms of frustration, confidence, and perceived control (satisfaction) [14, 51]. In the context of complex data pipelines, additional usability components such as learnability, memorability, and error recoverability become crucial, as they determine how easily analysts can understand available operations, resume interrupted workflows, and backtrack from incorrect transformations [52].

6.2.1 Cognitive Load and Mental Workload

Beyond global usability, cognitive load and mental workload offer a finer-grained lens on how demanding data preparation tasks are for human operators. Cognitive load theory distinguishes among intrinsic, extraneous, and germane load, emphasizing that interface design should minimize unnecessary (extraneous) load so that users can focus on the task's conceptual difficulty [55]. In practice, mental workload can be assessed through subjective instruments such as the NASA–TLX and single-item mental effort scales, as well as behavioral and physiological indicators [33, 56, 81]. For data preparation tools, these measures can be tied to specific operations, such as schema alignment, rule specification, or conflict resolution, to identify interface patterns that overload working memory or require excessive context-switching. Eye-tracking–based measures, including fixation patterns and gaze-shift rate, have also been explored as proxies for mental workload in visualization evaluation, showing that increased cognitive load tends to induce more erratic gaze dynamics and slower, less focused visual processing [80, 81].

Table 6.1 Revised latency levels adapted from Waloszek and Kreichgauer [75]

Level	Latency threshold	Description
Level 0	0.1 (0–0.2) s	Perceptual level: feedback after UI input involving direct manipulation/hand-eye coordination (e.g., mouse click, mouse movement, keypress)
Level 1	1.0 (0.2–2) s	Dialog level: completing simple tasks (e.g., opening a window or dialog box, confirming a simple operation)
Level 2	3.0 (2–5) s	Cognitive level: completing common tasks, such as logging in or applying a moderate transformation
Level 3	10 (5–15) s	Cognitive level: completing complex tasks, such as expensive searches, complex calculations, or full-pipeline recomputations
Level 4	>15 s	

6.2.2 Latency in Data Preparation Workflows

Latency is a central human factor in data preparation because it directly shapes how users perceive responsiveness, maintain attention, and sustain sensemaking as they iteratively transform and inspect data. In interactive tools, operations such as filtering, profiling, cleaning, or re-running quality checks often trigger complex queries on large datasets; when system responses exceed certain temporal thresholds, users may lose focus, distrust results, or abandon exploratory workflows altogether [9, 77]. Unlike purely transactional workloads, acceptable latency in human-facing data preparation is governed by cognitive constraints rather than by query semantics alone.

A large body of HCI work has investigated latency thresholds, the maximum response times that can be tolerated before user cognition and task performance start to degrade. Early guidelines by Miller, Shneiderman, and Nielsen proposed rules-of-thumb (e.g., 0.1 s, 1 s, 10 s), but these values were only loosely grounded in empirical evidence [45, 52, 63]. Later, Dabrowski and Munson conducted controlled experiments to derive new thresholds tied to entire UI widgets (buttons, menus, dialogs) rather than abstract operations [21]. Nah's review refined these insights and argued that delays of around 2 s for simple interactions are already perceived as unacceptable in many contexts [50]. Building on this, Shneiderman et al. introduced an additional 3 s threshold for common tasks, showing that beyond this point users tend to perceive systems as slow and begin to lose task focus [66]. Waloszek and Kreichgauer revisited Nielsen's thresholds, recasting them as ranges, adding the 3 s category, and extending the upper bound to 15 s to better capture complex tasks [75].

These latency levels are particularly informative for data preparation. Level 0 (perceptual) captures immediate feedback for direct manipulation actions (e.g., brushing a histogram, dragging a slider); Level 1 (dialog) covers short operations such as opening a quality panel or applying a simple filter; Levels 2–4 describe progressively longer delays associated with login-like interactions, complex searches, or

expensive recomputations. Data preparation systems ideally keep frequent, iterative operations (e.g., editing transformation parameters, toggling quality rules, previewing the effect of a cleaning step) within Level 0 or Level 1. More expensive operations (e.g., full re-profiling of a large dataset or recomputing global quality metrics) may fall into Level 2 or Level 3, but should then be clearly signaled, made interruptible, or supported via progressive results.

Recent empirical studies in visualization further refine how these latency bands impact exploratory behavior. Liu and Heer demonstrate that increasing latency from sub-200ms (Level 0) to around 1 s (Level 1) significantly reduces interaction frequency, dataset coverage, and hypothesis generation in exploratory visual analysis [41]. Stolper et al. investigate longer delays (6 s vs. 12 s, between Level 2 and Level 3) in progressive visual interfaces and show that progressive computation can partially compensate for high latency by providing early, coarse results that keep users engaged and support incremental refinement [67]. These findings suggest that data preparation tools should prioritize instant or near-instant responses for highly iterative actions (e.g., local previews, incremental filtering), and adopt progressive or approximate strategies for heavier operations, clearly communicating that results will refine over time.

In big data settings, guaranteeing that data preparation operations always remain below fixed latency thresholds is often infeasible [9]. Nevertheless, system design can be tuned to human latency expectations by carefully mapping interface elements to underlying queries and by scheduling work according to perceived priority. For example, dashboards that expose multiple quality views may generate many concurrent queries; blindly executing all of them as fast as possible can lead to contention and starve the views the user is currently attending to [77]. Instead, data preparation systems *should: (i)* identify which views are in focus and prioritize their updates; *(ii)* delay or downgrade updates for peripheral views; and *(iii)* exploit prefetching, incremental computation, and caching for likely next interactions. From a human-factors perspective, this calls for latency-oriented design patterns, such as optimistic updates, skeleton screens, and explicit progress indicators, that help maintain trust and situational awareness even when operations cannot complete within the strictest thresholds.

6.2.3 Modeling Interaction and Exploration from User Traces

Exploratory tasks in visual analytics blend interactive exploration with human cognition to derive insights from complex datasets within visualization systems. In data preparation, these tasks enable end users to navigate transformations, quality assessments, and provenance trails comprehensively, identifying latency issues [9], usability gaps, and interaction bottlenecks. Interactive systems typically operate through event loops, in which input events from devices are processed by applications, trig-

gering actions such as rendering. To manage event complexity, the HCI literature offers diverse models, ranging from low-level descriptions [4] to high-level user intents [23, 68]. High-level tasks like selection can be realized concretely through mouse clicks, lasso tools, or search boxes, while widgets encapsulate both visual representations and behaviors, connected via callbacks [5]. However, widgets tightly couple appearance and logic, hindering independent modifications [12, 34, 54]. More structured approaches separate visual components from behavior. Meyers' Interactors [49] employ state machines to specify interactions, complemented by earlier HCI works modeling interactions as state machines [29, 36, 57]. Popular toolkits like D^3 [13] use callbacks for stereotyped interactions (e.g., drags), while Vega [61] leverages reactive functional programming. Yet these implementations often treat state machines as black boxes, obscuring the links between low-level events and high-level operations such as filtering and aggregation. For transparent modeling amenable to optimization across query, rendering, and perceptual/cognitive constraints, statecharts [32] provide a declarative, standardized foundation. As high-level specifications that compile to simple state machines, statecharts represent directed graphs with states as nodes and transitions triggered by events (optionally guarded), which execute actions upon data model updates. This navigable structure supports concurrent reasoning during execution and explicitly manages interaction states and data models.

In Sect. 6.4, we will revisit these concepts in the context of concrete systems that record and analyze interaction traces to optimize data preparation workflows.

6.2.4 User Control, Situational Awareness, and Trust

Other important dimensions relate to *user control*, *situational awareness*, and *trust*. User control reflects the degree to which analysts can steer the pipeline, such as parameterizing transformations, overriding automated suggestions, or undoing operations, and can be assessed through measures such as the number of manual overrides, the use of advanced controls, and subjective ratings of perceived controllability [40, 65]. Situational awareness, in turn, concerns how well users understand the current state of the data, the provenance of changes, and the status of automated processes; evaluations can examine whether users can correctly describe recent transformations, identify the root cause of anomalies, or anticipate the effects of a proposed operation [7, 70]. Finally, trust and calibration of trust in semi-automated preparation tools can be measured through questionnaires and behavioral indicators (i.e., reliance on suggestions, frequency of verification), capturing whether users appropriately balance automation benefits against the risk of errors or opaque behaviour [16, 27].

6.2.5 *Visual Analytics Dimensions*

Beyond latency and interaction patterns, several visual analytics dimensions are particularly relevant when supporting data preparation tasks through visualization. These dimensions help assess whether visual representations of intermediate or final prepared data actually support accurate, efficient, and trustworthy human judgment.

Granularity captures the level of detail at which data is presented and manipulated. If granularity is too coarse, important anomalies, outliers, or small subgroups may be hidden; if it is too fine, users can be overwhelmed by clutter and noise. In data preparation, appropriate granularity determines whether an analyst can both diagnose local data quality issues (e.g., record-level errors) and maintain an understanding of global patterns (e.g., overall completeness or distribution shifts). Effective tools, therefore, allow users to fluidly navigate between overview and detail, often by providing "overview first, zoom and filter, then details-on-demand" interactions [64, 76]. Evaluating granularity involves checking whether the resolution of views matches the current task (e.g., inspecting a small subset versus auditing entire columns) and whether navigation between levels of detail is cognitively and interactively smooth (as seen in Fig. 6.1).

Sparsity refers to the extent of missing, incomplete, or structurally sparse data and how that sparsity is communicated visually. High sparsity can hinder readability and interpretation, for example, when large portions of a matrix or time series are left blank or aggregated away. In data preparation, sparsity is closely tied to data quality: missing values, absent categories, or undersampled regions in the data space often signal where imputation, augmentation, or targeted collection is required. Visual analytics systems can represent sparsity explicitly (e.g., by encoding missingness in

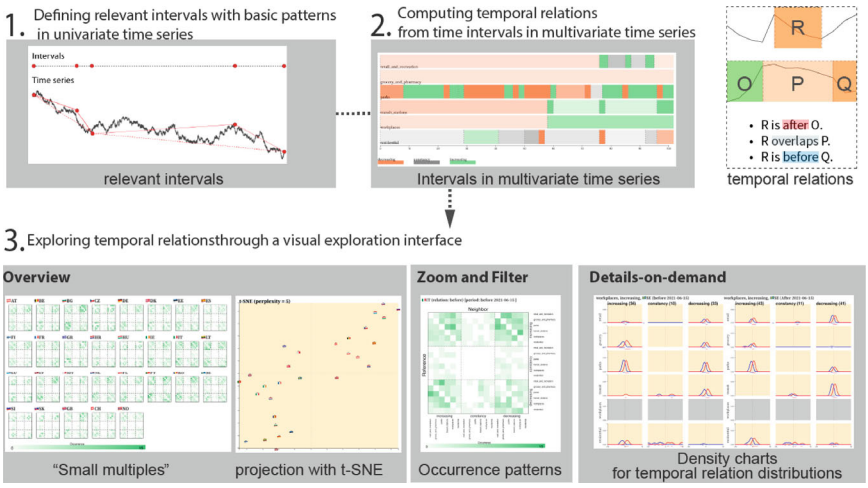


Fig. 6.1 Shneiderman's task taxonomy illustrating granularity levels in data visualization: overview first, zoom and filter, then details-on-demand [64]

dedicated layers or glyphs) and support interaction techniques to explore where and why data is absent [22, 42]. Quantitatively, sparsity can be assessed by measuring the proportion and distribution of missing or zero entries and by analyzing how these patterns affect the stability and interpretability of visual summaries.

Visual integrity concerns whether visualizations faithfully encode the underlying prepared data without introducing distortions that mislead users. Violations include inconsistent scales, truncated axes, inappropriate aggregation, or excessive embellishment that obscures data variation. For data preparation, maintaining visual integrity is critical: if quality dashboards or provenance views distort magnitudes or trends, users may make incorrect decisions about which cleaning rules to apply or which subsets to trust. Principles from the visualization literature, such as avoiding "chartjunk", maximizing the data–ink ratio, and aligning visual encodings with the semantics and scales of the data, all contribute to preserving integrity [39, 71]. Evaluating this dimension involves assessing whether the chosen encodings, scales, and annotations support honest comparison and do not exaggerate or obscure data quality issues.

Volatility captures how quickly data, quality indicators, or model outputs change over time, and how promptly these changes are reflected in the visual interface. In a data preparation context, volatility arises when new data arrives continuously, upstream sources are unstable, or cleaning and transformation pipelines are frequently rerun. Highly volatile environments demand that visualizations remain timely and coherent: if quality metrics lag too far behind underlying changes, users may act on stale information; if views update too aggressively, they may disrupt ongoing analysis. Timeliness can be assessed by measuring the latency between data updates and visual refreshes, and by characterizing how different refresh strategies (e.g., periodic batching, event-driven updates, progressive refinement) affect comprehension and trust [9, 41].

Finally, **transparency** refers to how clearly data preparation processes and visualization algorithms expose their assumptions, transformations, and limitations. In many pipelines, complex sequences of extraction, cleaning, integration, and aggregation are hidden behind opaque interfaces, making it difficult for users to understand why a given visual result appears as it does. Provenance-driven visualization approaches show that systematically documenting and visually exposing data lineage and transformation steps can promote transparency, deeper interpretation, and more ethical use of data [73]. In the context of data preparation, transparency can be evaluated by examining whether users can access explanations of the applied operations, see how quality metrics were computed, and trace visual elements back to raw or intermediate data. High transparency supports calibrated trust and helps users detect when visual analytics may be masking unresolved data quality issues.

Table 6.2 summarizes the main human factors and visual analytics dimensions discussed above, highlighting their meaning, typical metrics, and the stages of the data preparation pipeline where they are most relevant. This summary provides a conceptual scaffold that we will use in Sect. 6.4 to structure the discussion of concrete systems and use cases.

Table 6.2 Human factors and visual analytics dimensions in data preparation

Dimension	Typical metrics/observations	Data preparation stage(s)
Usability (efficiency, effectiveness, satisfaction)	Task completion time, error rate in transformations, SUS scores, subjective satisfaction [14, 35, 51]	All (extraction, wrangling, profiling, quality checks)
Cognitive load	NASA–TLX, mental effort ratings, fixation patterns, gaze-shift rate [33, 55, 81]	Complex wrangling, schema alignment, rule specification
Latency	Time-to-first-render, response time to interactions, progressive update cadence [9, 41, 75]	Interactive filtering, profiling, quality dashboards
User control	Number of manual overrides, use of advanced controls, perceived controllability [40, 65]	Rule editing, parameter tuning, conflict resolution
Situational awareness	Correct recall of transformations, ability to identify error sources, provenance comprehension [7, 70]	Provenance exploration, quality diagnosis
Trust	Reliance vs. verification rates, trust questionnaires [16, 27]	Semi-automatic cleaning, LLM-based transformations
Granularity	Ability to move between overview and detail, clutter vs. missing anomalies [64, 76]	Record-level inspection vs. global quality monitoring
Sparsity	Proportion and distribution of missing values, visual encoding of missingness [22, 42]	Profiling, imputation, augmentation
Visual integrity	Use of appropriate scales, absence of distortive encodings, adherence to visualization guidelines [39, 71]	Quality dashboards, reporting
Volatility	Update latency between data and view, stability vs. thrashing of displays [9, 41]	Streaming sources, frequently updated pipelines
Transparency	Availability of operation logs, provenance visualizations, explanation facilities [73]	All stages, especially integration, wrangling, LLM-based preparation

6.3 Research Challenges

Despite growing interest in human-centered data systems, data preparation remains understudied compared to analysis or modeling, and several research challenges emerge from the perspectives outlined above.

A first challenge concerns *bridging system-centric and human-centric performance metrics*. Traditional data management focuses on metrics such as throughput, latency, and resource utilization, yet Human–Data Interaction work shows that these indicators do not necessarily correlate with perceived responsiveness or fluid interaction [77]. Small interface actions (for example, adjusting a slider) may trigger bursts of queries, many of which update off-screen or low-priority views, so naively minimizing query latency can still yield poor user experience [77]. For data preparation tools, this gap calls for evaluation frameworks that couple system-level metrics with human-centered measures such as perceived latency, interaction frequency, exploration coverage, cognitive load, and error rates during cleaning and transformation tasks. Case studies on interaction-driven optimization and progressive visualization already point in this direction, but more systematic methodologies and shared benchmarks are needed to enable comparisons of designs across tools and domains [10, 58].

A second challenge is the *co-design of interfaces and systems under design dependence*. Human–Data Interaction (HDI) work argues that visualization interfaces and data systems are tightly coupled: interface designs impose latency, interaction semantics, and consistency demands on the system, while physical design and execution strategies constrain which interactions are feasible at interactive timescales [77]. In data preparation, this dependence is particularly strong because interfaces both inspect and manipulate the underlying data. Emerging approaches such as Physical Visualization Design (PVD) suggest that formal interface representations, including data flow and latency constraints, can guide the automatic synthesis of execution plans and caching strategies [17]. Extending such ideas to data preparation raises open questions about how to represent cleaning, profiling, and quality-steering interactions in a way that allows optimizers to choose human-aware physical designs while preserving transparency and debuggability for end users. The Translation layer and TraVIS case studies illustrate first steps in this direction by using augmented statecharts as a shared abstraction across interaction, data, and rendering layers [10, 30], but further work is needed to integrate such models into mainstream data preparation platforms.

A third challenge involves *calibrating human involvement along the human-in/on/over-the-loop spectrum*. Surveys on human-in-the-loop machine learning highlight a rich design space of when and how humans should intervene, during data collection, labeling, feature engineering, model training, or monitoring [78]. Data preparation pipelines similarly span phases where human expertise is essential (for example, resolving schema conflicts, specifying quality rules, adjudicating ambiguous labels) and phases that can be safely automated. However, there is little guidance on how to allocate human effort across these stages, how to escalate from automation

to human review under uncertainty, and how to avoid over-reliance on automation in high-stakes quality decisions. The systems discussed in Sect. 6.4 offer partial patterns, such as interaction-driven indexing or progressive previews that keep users in the loop, but a more principled account of human roles, escalation policies, and accountability in data preparation remains an open problem [27, 79].

A fourth challenge relates to *modeling and leveraging interaction traces*. As discussed in Sect. 6.2, interaction models and state-based specifications provide a foundation for capturing how users explore and manipulate data. For data preparation, interaction logs capturing sequences of transformations, quality inspections, and manual corrections could be mined to identify common workflows, expert strategies, and pain points. TraVIS shows how augmented statecharts and trace analysis can reveal hierarchical interaction structures and latency bottlenecks in complex visualization systems [30]. However, robust frameworks are still missing for representing these traces across tools, preserving privacy, and turning them into actionable adaptations (for example, macro suggestions, adaptive defaults, or personalized quality views) without introducing new cognitive overhead or bias. Connecting such interaction analytics with sustainability-aware models, as in recent work on progressive and on-demand data preparation [58], is an especially promising direction.

A fifth challenge concerns *communicating uncertainty, approximation, and data quality* in ways that support trustworthy decisions. Approximate query processing, progressive visualization, and sampling-based profiling are increasingly used to meet latency and sustainability constraints, but HDI research shows that users struggle to interpret statistical uncertainty and that naïve error bounds can distort perceived patterns [72, 77]. In data preparation, approximate quality scores, partial cleaning results, and heuristic LLM-generated transformations must be presented so that users understand their limitations. Recent work on human-centered AI and AI-supported human-centered design suggests that LLMs should be treated as components in human–AI systems, with explicit attention to roles, boundaries, and accountability [62, 74, 79]. However, designing transparent interactions and addressing human factors such as trust calibration and situational awareness remain open challenges [16, 73].

Finally, *AI- and LLM-augmented data preparation* introduces a new set of research questions that cut across all the above challenges. Large language models are increasingly used to suggest transformations, write data-wrangling code, infer schemas, and even generate synthetic data, but current practice offers limited transparency about how these suggestions are derived or how reliable they are for a given context [3, 82]. Open problems include: how to expose the provenance and rationale of AI-generated operations; how to design feedback mechanisms that allow systems to learn from human corrections without eroding user control; how to avoid amplifying existing data biases and to assess sustainability impacts of LLM-based workflows; and how to evaluate such hybrid solutions with metrics that capture both technical quality (for example, error rates, coverage) and human experience (for example, trust, workload, sense of agency). Recent work on human-centred AI and AI-supported human-centred design suggests that LLMs should be treated as components in human–AI systems, with explicit attention to roles, boundaries, and accountability [62, 74, 79].

Bringing these insights into the design and assessment of data preparation tools is an important agenda for future research.

6.4 Case Studies

The previous subsections introduced a conceptual characterization of human factors in data preparation and outlined key research challenges. We now turn to concrete systems and scenarios that exemplify how these issues arise in practice and how they can be addressed. We focus on a set of representative case studies that span different stages of the data preparation pipeline and illustrate distinct combinations of human factors. Section 6.4.1 discusses the modification of the classical visualization pipeline by leveraging statecharts for interaction modeling, while Sect. 6.4.2 provides visualization system designers a diagnostic tool to analyze user behavior within a visualization system, analyzing user traces and the statechart representation to reveal latent structure in interaction spaces. Section 6.4.3 presents systems that use progressive computation to reconcile latency constraints with rich visual exploration. Finally, Sect. 6.4.4 reviews emerging AI- and LLM-based data preparation workflows, highlighting how trust, transparency, and control are negotiated in these hybrid settings.

6.4.1 Optimizing Exploratory Data Analysis in Big Data Visualization Systems

From the perspective of data preparation, exploratory visual analysis typically sits at the end of the pipeline, where analysts interactively inspect cleaned, integrated, and profiled data to validate quality and derive insights. In big data settings, however, this last step can become the new bottleneck: even when upstream preparation is well engineered, interactive dashboards may still suffer from high latency because each small visual action (for example, brushing or filtering) translates into dozens of heavy queries on large tables [9]. This undermines several human factors highlighted in Sect. 6.2, including cognitive load, perceived latency, and trust in quality views, and can even distort user feedback during user-centered design (UCD) evaluations. A recent framework by Benvenuti et al. [10] tackles these issues by revisiting the classical visualization pipeline and explicitly accounting for user interaction traces, building on Moreland's three-layer model [47], which distinguishes a data layer, a rendering layer, and an interaction layer. Each layer, taken in isolation, can introduce latency and usability problems. Large datasets and high query rates can push DBMS response times beyond human-centered thresholds (technical issue 1). Low-level event forwarding, such as directly sending raw mousemove or brush events to the data layer, can generate excessive query loads for simple interactions (technical issue

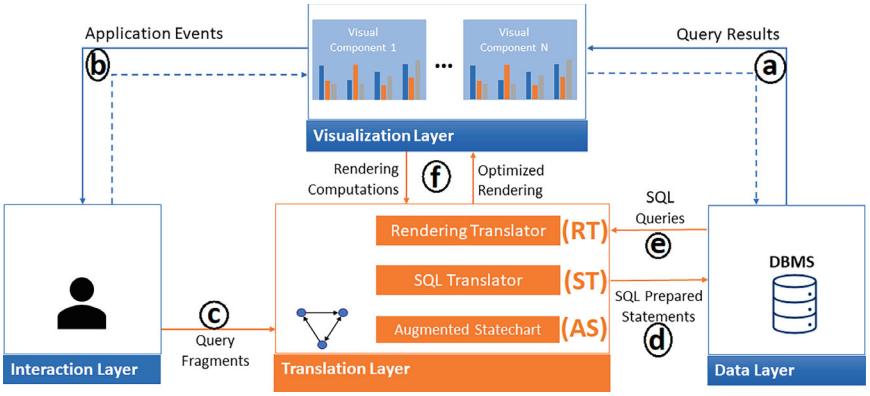


Fig. 6.2 How the classical visualization pipeline can be enhanced by introducing a Translation layer between the data, interaction, and visualization layers [10]

2). The exploratory nature of use makes it difficult to reason about the computational cost and latency sensitivity of all possible interactions (technical issue 3). Existing optimizations are rarely coordinated across layers (technical issue 4), sometimes leading to visually redundant updates that do not improve situational awareness (technical issue 5).

To address these limitations, the framework introduces a fourth tier, the Translation layer, between the interaction, data, and rendering layers (Fig. 6.2). The core idea is to model the entire interaction space as an augmented statechart, where nodes represent interface contexts and edges represent user-triggered transitions (for example, mousemove, click, or brush), each annotated with a desired latency threshold derived from empirical HCI studies (Level 0 to 4 as in Table 6.1). On top of this, an SQL Translator associates each transition with a parameterized SQL template, while a Rendering Translator links query results back to appropriate rendering strategies. In a data preparation context, this means that frequent quality-steering operations (such as brushing on delay or completeness histograms, or toggling cleaning rules) can be tied to concrete query patterns and latency requirements, turning end-of-pipeline interaction logs into actionable signals for reshaping the prepared data and its access paths. One tangible optimization that can be guided in this way is indexing at the data layer. Indexes create auxiliary structures, such as B-trees or hash indexes, over frequently queried columns so that lookups, range scans, and joins can be executed in sublinear time [44].

The next example relies on indexing optimization, demonstrating how interaction-driven optimization can improve the user experience in an exploratory visualization built on a large, prepared flights dataset. 50 user traces were used, collected within a user study involving exploratory tasks performed on Falcon, a Crossfilter-style interface displaying 2006 US flights data [48], connected to a 10M-row DuckDB instance (Fig. 6.3).

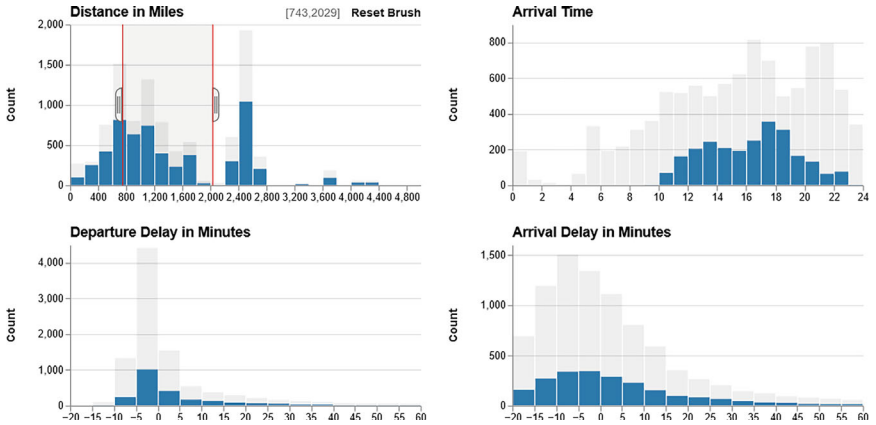


Fig. 6.3 The Falcon crossfilter visualization system [48]. Brushing one visual component updates the others in real-time

Table 6.3 Total execution time of interaction-induced queries for a representative user trace, with and without indexing on frequently used attributes

Condition	Total time (s)
No indexing	155.72
With indexing	146.62
Time saved	**9.10 (5.85%)**

The traces were replayed, mimicking all the interactions both on Falcon and over the augmented statechart, to identify where latency violations occur and which attributes dominate the WHERE clauses of interaction-induced queries. For a representative trace with 21 brush interactions and 8 latency violations, the analysis shows that attributes such as `ARR_TIME`, `FL_DATE`, and `AIR_TIME` are queried far more often than other attributes. Guided by this evidence, let's create indexes on these three attributes and replay the same trace on two versions of the system: one without indexing and one with indexing enabled. Table 6.3 reports the total execution time of the interaction-induced queries in the two conditions. Overall, indexing reduces the total query time from 155.72 s to 146.62 s, yielding a saving of 9.10 s, which is about 5.85% of the computation time for that trace. The largest gains occur precisely on the brushed components that were most heavily used (Airtime in Minutes and Departure Time), confirming that interaction-driven indexing can directly target latency-critical parts of the exploration workflow. From a human-factors standpoint, this directly reduces perceived latency for the most cognitively demanding operations, supporting more fluent exploration and more reliable assessment of data-quality patterns.

This example also illustrates important limitations and design trade-offs. Building and maintaining indexes is costly for a DBMS and can slow down insertions and updates, so it is not feasible to index everything by default. Instead, the frame-

work advocates focusing on columns involved in interactions that actually violate latency thresholds, as revealed by the augmented statechart and user traces. Moreover, the reported improvement is a baseline obtained on relatively simple queries without joins; indexing typically shows even greater benefits for complex, multi-table queries, which often underpin more advanced data preparation dashboards [44]. Nonetheless, the case study demonstrates how the Translation layer and statechart modeling can systematically connect interaction behavior at the visualization end of the data preparation pipeline to concrete optimizations in the data layer, closing the loop between human-centered evaluation and system-centric performance tuning.

6.4.2 Analyzing Interaction and Exploration via TraVIS

The previous sections highlighted the importance of modeling interaction and exploration to understand and optimize human behavior in data preparation. TraVIS [30] is a system that demonstrates how to apply these principles in practice by automatically discovering and visualizing interaction patterns in complex visualization systems. TraVIS targets the analysis of user interactions within visualization systems, modeling them as statecharts to reveal hierarchical structures, providing the designer with the complete interaction space of the visualization system. The statechart is a directed graph in which nodes represent interaction states and edges capture user-triggered transitions, optionally guarded by conditions and executing actions on the visualization system (e.g., a mousemove, a click, or a brush interaction). This declarative format supports hierarchical nesting, concurrency, and navigable structure, making it ideal for reasoning about complex interaction spaces. The system, leveraging user traces, applies a modified Depth-First-Search (DFS) to identify rarely traversed paths, dominant interaction types, and structural inefficiencies that may indicate parts of the visualization system that are rarely explored or suffer from major latency issues. Figure 6.4 shows TraVIS analyzing the W4SP visualization system [6], a complex environment for big data analytics. The left panel displays the W4SP interface in active use, while the right panel shows the automatically generated statechart that maps its complete interaction space. The color-coded legend distinguishes interaction types (e.g., orange for dominant mouse movements), and the minimap facilitates navigation of the deeply nested hierarchy. This visualization reveals that W4SP's interaction structure is highly hierarchical, with most actions relying on a few prevalent interaction types: a pattern that suggests opportunities for simplification, such as consolidating redundant mouse actions into higher-level macros or surfacing expert shortcuts. In the context of data preparation, TraVIS could be applied to systems where analysts spend significant time exploring data quality profiles, testing cleaning rules, or validating schema alignments. For example, interaction traces from a profiling dashboard might reveal that users frequently cycle through filtering → inspection → rule adjustment → re-profiling loops, but get stuck in low-level mouse manipulations. By surfacing these patterns as statecharts, TraVIS enables developers to identify and mitigate bottlenecks, such as

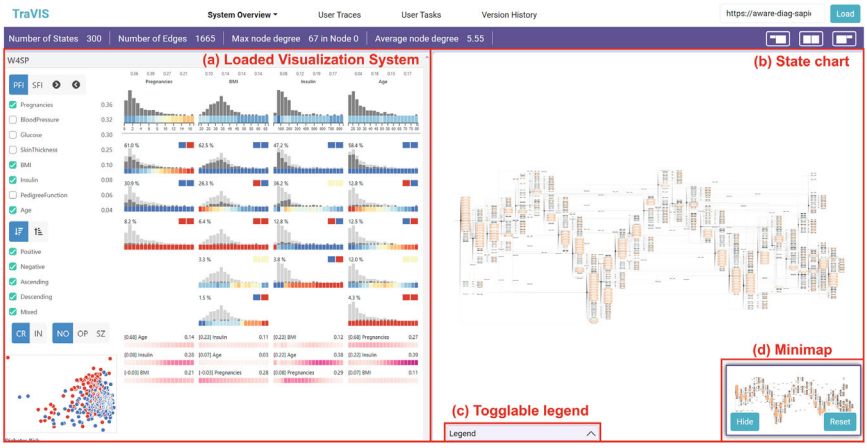


Fig. 6.4 TraVIS' main interface displays the W4SP visualization system [6] (left, **a**) alongside an automatically generated statechart (right, **b**) that maps its complete interaction space as a graph of states and user-triggered transitions (the low-level interactions, such as mousemove, click, etc., happening within the visualization systems). A color-coded legend (**c**) identifies interaction types, while a minimap (**d**) aids navigation. TraVIS allows us to see W4SP's deeply hierarchical interaction structure, evident in the nested statechart, where most actions rely on a few dominant interaction types (prevalent orange)

excessive context-switching between quality views or inefficient navigation of provenance trees. This directly addresses several human factors from Table 6.2: it improves *cognitive load* by reducing redundant interactions, enhances *user control* by exposing latent structure, and supports *situational awareness* by making the system's interaction model transparent and inspectable. Beyond analysis, TraVIS's statechart models provide a foundation for runtime optimizations. For instance, rarely used branches could trigger on-demand computation rather than prefetching, while dominant paths could be accelerated with specialized caching. In data preparation scenarios with high latency sensitivity (Sect. 6.2.2), such optimizations would help maintain "conversation speed" during iterative quality steering. Preliminary evaluations in the original paper show that TraVIS successfully uncovers interaction hierarchies in real systems, suggesting its potential for broader adoption in human-aware data preparation tools. Overall, TraVIS allows the visualization system designer to automatically discover the most frequent interaction patterns and analyze which of them are most affected by latency. This enables the application of tailored optimization strategies, leveraging the interactive nature of the visualization system through the statechart representation and user trace analysis. The next subsection explains how such analysis can be used to hypothesize a specific optimization technique in a visualization system whose potential latency is deeply tied to a certain interactive behavior.

6.4.3 Progressive Visual Analytics for Data Preparation

Progressive visual analytics refers to approaches in which data and visualizations are updated incrementally rather than waiting for a complete, final result before showing anything to the user [8, 72]. Instead of executing long-running queries or cleaning operations in a single batch, the system delivers early, coarse results that are then refined over time. This paradigm is particularly relevant in data preparation, where datasets can be large, pipelines can be complex, and users need timely feedback to guide decisions on cleaning, enrichment, and integration [58]. From a human-factors viewpoint, progressive visual analytics aims to keep latency within acceptable thresholds (Sect. 6.2.2) and to reduce cognitive load by giving the user something to inspect and reason about early in the process, even if it is approximate. In the context of data preparation, progressive visualization supports both the inspection and the improvement phases [58]. During inspection, progressive views expose distributions, patterns, and outliers without requiring full data loading or complete quality computation. For instance, a progressive histogram of completeness can first display approximate counts for a sample or aggregated tiles, then refine bin heights as more data are processed. This allows analysts to identify problematic columns or regions of the data space and decide where to invest cleaning effort, directly supporting reduce strategies in sustainable pipelines [58]. During improvement, progressive previews of cleaning or imputation results let users assess the impact of a transformation on a subset of records before committing it globally, thereby limiting unnecessary work and avoiding over-cleaning that yields negligible quality gains.

An example drawn from digital marketing data illustrates how progressive visual analytics can yield concrete benefits [58]. Consider a scenario in which an analyst must geocode tens of thousands of city-level records and inspect their quality through an interactive map and bar charts. In a bulk workflow, all records are geocoded, and quality flags are computed before the visualization is updated. This can require tens of minutes of computation, high API usage, and substantial assessment cost, since every record is processed at fine granularity. A progressive alternative first aggregates at state level to identify a smaller set of "dirty" rows, for example, around five thousand cities with missing or inconsistent information, and only then performs detailed geocoding and quality checks on that subset. The interface updates after each mini-batch, so the analyst can immediately see which states and cities are problematic and can start correcting them while the system continues to refine the remaining results. In such a setting, the progressive approach can reduce the number of rows processed at high granularity to about 10% of the total, cut assessment cost by roughly an order of magnitude, and significantly lower energy consumption, while still achieving similar final data quality [58]. Progressive visual analytics also interacts with several human factors discussed in Sect. 6.2. Providing early partial results improves perceived responsiveness and helps keep users within Level 0 or Level 1 latency bands for many interactions [41, 58, 72]. Immediate but approximate feedback supports better situational awareness, since users can see how quality metrics evolve as more data are processed and where remaining gaps are located [58]. At the same time, the

design must carefully manage cognitive load and visual integrity: early views should communicate their provisional nature, and uncertainty should be made explicit so that users do not over-trust incomplete results. When used thoughtfully, progressive techniques help balance granularity, latency, and transparency, which are key dimensions of visual analytics for data preparation (Sect. 6.2.5). Finally, progressive methods can be combined with interaction-driven frameworks such as the Translation layer and with tools like TraVIS. Interaction logs and augmented statecharts can identify which parts of the data and which operations are most critical for users, so that progressive computation focuses on those subsets first [10, 30, 58]. For example, if traces show that analysts repeatedly brush on a small set of quality histograms, progressive pipelines can prioritize early updates for those views and defer work on rarely used components. In this way, progressive visual analytics not only supports sustainable execution but also aligns computational effort with actual human behavior in data-preparation workflows, closing the loop among quality steering, user experience, and resource-aware system design [58].

6.4.4 AI and LLM-Based Data Preparation in Practice

Large language models (LLMs) are increasingly used to support data preparation, for example, by suggesting transformations, generating data-wrangling code, imputing missing values, synthesizing metadata, and even proposing quality rules or constraints [3, 82]. In principle, this can reduce manual effort and lower the barrier to entry for non-expert users, but it also introduces new human-factor challenges around trust, transparency, and control that are especially critical when LLMs operate over large, heterogeneous datasets. In line with the human factors outlined in Sect. 6.2, LLM-based data preparation needs to be framed as a form of human–AI collaboration rather than full automation, with analysts retaining the role of ultimate decision makers over how data are cleaned and transformed [79]. A first class of scenarios involves LLMs providing *assistive suggestions* within interactive data preparation tools. Given natural language prompts such as "split this column into city and state" or "standardize date formats," an LLM can propose code snippets, declarative transformation specifications, or even multi-step pipelines that users can inspect, edit, and apply [82]. This can lower cognitive load for complex operations and improve perceived usability, particularly for analysts who are not expert in data-wrangling languages. At the same time, it shifts part of the sensemaking burden: users must evaluate whether suggested operations are appropriate, anticipate side effects on data quality dimensions, and understand how to revert or adapt them. Human-centered AI work emphasizes that such hybrid intelligence arrangements require explicit support for explainability, oversight, and shared control, enabling users to calibrate their reliance on AI across tasks and scenarios [79]. A complementary use of LLMs is as *world models* that encode large volumes of human-written experience and can be queried to simulate stakeholders or users involved in data-preparation workflows [62]. For example, LLMs can generate personas of data

stewards with different expertise, suggest likely pain points in a wrangling interface, or enumerate domain-specific quality checks that an expert might expect in a profiling dashboard. Such synthetic input can help designers explore the space of requirements and interaction patterns before running full user studies, potentially reducing the number of costly evaluation cycles. However, Schmidt et al. caution that LLMs reproduce biases present in their training data and may over-represent some perspectives while under-representing others, so their use in design and evaluation must remain transparent and be complemented with real user involvement wherever possible [62]. In data preparation contexts, this implies that AI-generated "expert advice" on quality rules or cleaning strategies should be clearly identified as such and validated against domain knowledge and empirical evidence. LLMs also raise questions about sustainability and agency in data preparation workflows. On the one hand, they can help implement data-centric AI strategies that focus on improving data quality rather than endlessly scaling model size, for instance, by prioritizing the most impactful cleaning operations or by identifying redundant or low-value data that can be safely discarded [58]. On the other hand, if incorporated uncritically as black-box decision makers, they risk reinforcing the very patterns that critical HCD work has highlighted in other domains: over-automation, loss of human agency, and optimization for short-term efficiency at the expense of long-term sustainability and control [74]. For example, fully automatic LLM-based cleaning that silently rewrites values or imputes missing data without clear provenance may improve some technical metrics while making it harder for analysts to understand what was changed, why, and with what confidence, thereby harming situational awareness and trust. Therefore, designing LLM-augmented data preparation tools that respect human factors requires a careful combination of interaction and system patterns. From an interaction perspective, interfaces should expose AI contributions explicitly (for instance by marking AI-suggested transformations or imputed values), provide justifications or examples on request, and offer lightweight mechanisms for correction and override so that users can steer and audit the AI's behaviour [62, 79]. From a systems perspective, LLM-based components need to integrate with provenance tracking and quality metrics, so that their effects on the data can be traced and evaluated alongside traditional operations [73]. Finally, in line with recent calls to rethink human-centered design around sustainability and agency, LLM use in data preparation should be tuned to support human judgment and long-term maintainability of pipelines, rather than simply maximizing automation or throughput [74]. Within such a human-aware and sustainability-aware frame, LLMs become one component of a broader ecosystem that includes interaction-driven optimization, progressive visual analytics, and human-in-the-loop governance over data quality.

References

1. Abedian, Z., Golab, L., Naumann, F.: Data profiling: a tutorial. In: Proceedings of the 2017 ACM International Conference on Management of Data, SIGMOD Conference 2017, pp. 1747–1751 (2017). https://doi.org/10.1145/3035918.3054772
2. ACM Sigchi Curriculum Development Group: ACM SIGCHI curricula for human-computer interaction. Tech. rep, Association for Computing Machinery, New York, NY (1992)
3. Albalak, A., Elazar, Y., Xie, S.M., Longpre, S., Lambert, N., Wang, X., Muennighoff, N., Hou, B., Pan, L., Jeong, H., Raffel, C., Chang, S., Hashimoto, T., Wang, W.Y.: A survey on data selection for language models. Trans. Mach. Learn. Res. (2024). https://openreview.net/forum?id=XfHWcNTSHp
4. Amar, R.A., Eagan, J., Stasko, J.T.: Low-level components of analytic activity in information visualization. In: Stasko, J.T., Ward, M.O. (eds.) IEEE Symposium on Information Visualization (InfoVis 2005), pp. 111–117. IEEE Computer Society (2005). https://doi.org/10.1109/INFVIS.2005.1532136
5. Angelini, M., Blasilli, G., Lenti, S., Palleschi, A., Santucci, G.: Crosswidgets: Enhancing complex data selections through modular multi attribute selectors. In: Proceedings of the 2020 International Conference on Advanced Visual Interfaces (AVI '20), pp. 12:1–12:9. ACM, New York, NY, USA (2020). https://doi.org/10.1145/3399715.3399918
6. Angelini, M., Blasilli, G., Lenti, S., Santucci, G.: A visual analytics conceptual framework for explorable and steerable partial dependence analysis. IEEE Trans. Visual Comput. Graphics **30**(8), 3895–3910 (2024). https://doi.org/10.1109/TVCG.2023.3263739
7. Angelini, M., Daraio, C., Lenzerini, M., Leotta, F., Santucci, G.: Performance model's development: a novel approach encompassing ontology-based data access and visual analytics. Scientometrics **125**(2), 865–892 (2020)
8. Angelini, M., Santucci, G., Schumann, H., Schulz, H.: A review and characterization of progressive visual analytics. Informatics **5**(3), 31 (2018) https://doi.org/10.3390/informatics5030031.
9. Battle, L., Chang, R., Stonebraker, M.: The role of latency and task complexity in predicting visual exploration behavior over large datasets. IEEE Trans. Visual Comput. Graphics **26**(1), 1101–1111 (2020)
10. Benvenuti, D., Filosa, M., Catarci, T., Angelini, M.: Modeling and assessing user interaction in big data visualization systems. In: Abdelnour Nocera, J., Kristín Lárusdóttir, M., Petrie, H., Piccinno, A., Winckler, M. (eds.) Human-Computer Interaction - INTERACT 2023, pp. 86–109. Springer Nature Switzerland, Cham (2023)
11. BigScience Workshop: BLOOM: A 176B-parameter open-access multilingual language model (2023). https://arxiv.org/abs/2211.05100
12. Blanch, R., Beaudouin-Lafon, M.: Programming rich interactions using the hierarchical state machine toolkit. In: Proceedings of the Working Conference on Advanced Visual Interfaces (AVI '06), pp. 51–58. ACM, New York, NY, USA (2006). https://doi.org/10.1145/1133265.1133275
13. Bostock, M., Ogievetsky, V., Heer, J.: D^3: data-driven documents. IEEE Trans. Visual Comput. Graphics **17**(12), 2301–2309 (2011). https://doi.org/10.1109/TVCG.2011.185
14. Brooke, J.: SUS: a quick and dirty usability scale. In: Usability Evaluation in Industry, pp. 189–194. CRC Press, London (1996). https://doi.org/10.1201/9781498710411-35
15. Card, S.K., Moran, T.P., Newell, A.: The Psychology of Human-Computer Interaction. Lawrence Erlbaum Associates, Hillsdale, NJ (1983)
16. Chai, C., Li, G.: Human-in-the-loop techniques in machine learning. IEEE Data Eng. Bull. **43**(3), 37–52 (2020)
17. Chen, Y., Li, X., Tao, J., Ramjit, L., Mitra, S., Ghaderi, J., Netravali, R., Parameswaran, A., Rubenstein, D., Wu, E.: Physical visualization design: decoupling interface and system design. Proc. ACM Manag. Data **3**(3), 197:1–197:27 (2025). https://doi.org/10.1145/3725334
18. Clow, D.: Data wranglers: Human interpreters to help close the feedback loop. In: Proceedings of the 4th International Conference on Learning Analytics and Knowledge (LAK), pp. 49–53. (2014)

19. Couto, J.C., Damásio, J.: New trends in big data profiling. J. Big Data **9**(1), 1–31 (2022)
20. Crescenzi, V., Mecca, G., Merialdo, P.: RoadRunner: towards automatic data extraction from large web sites. In: Proceedings of the 27th International Conference on Very Large Data Bases (VLDB), pp. 109–118. (2001)
21. Dabrowski, J., Munson, E.V.: Is 100 milliseconds too fast? In: CHI '01 Extended Abstracts on Human Factors in Computing Systems, pp. 280–281. ACM, New York, NY, USA (2001). https://doi.org/10.1145/634067.634255. https://doi.org/10.1145/634067.634255
22. Demirtas, H.: Flexible imputation of missing data. J. Stat. Softw., Book Rev. **85**(4), 1–5 (2018). https://doi.org/10.18637/jss.v085.b04. https://www.jstatsoft.org/index.php/jss/article/view/v085b04
23. Desolda, G., Esposito, A., Lanzilotti, R., Costabile, M.F.: Detecting emotions through machine learning for automatic ux evaluation. In: Human-Computer Interaction—INTERACT 2021, pp. 270–279. Springer International Publishing, Cham (2021)
24. Dix, A., Finlay, J., Abowd, G.D., Beale, R.: Human-Computer Interaction, 3rd edn. Pearson Education, Harlow, UK (2004)
25. Doan, A.: Human-in-the-loop data analysis: a personal perspective. In: Proceedings of the Workshop on Human-In-the-Loop Data Analytics, HILDA'18. Association for Computing Machinery, New York, NY, USA (2018). https://doi.org/10.1145/3209900.3209913
26. Dong, X.L., Srivastava, D.: Big data integration. Proc. VLDB Endow. **6**(11), 1188–1189 (2013)
27. Endert, A., Hossain, M.S., Ramakrishnan, N., North, C., Fiaux, P., Andrews, C.: The human is the loop: new directions for visual analytics. J. Intell. Inf. Syst. **43**(3), 411–435 (2014)
28. Ferrara, E., De Meo, P., Fiumara, G., Baumgartner, R.: Web data extraction, applications, and techniques: a survey. Knowl.-Based Syst. **70**, 301–323 (2014)
29. Feyock, S.: Transition diagram-based CAI/HELP systems. Int. J. Man Mach. Stud. **9**(4), 399–413 (1977)
30. Filosa, M., Plexousaki, A., Di Stadio, M., Bovi, F., Benvenuti, D., Catarci, T., Angelini, M.: TraVIS: a user trace analyzer to support user-centered design of visual analytics solutions. IEEE Trans. Visual Comput. Graphics **31**(10), 7614–7631 (2025). https://doi.org/10.1109/TVCG.2025.3546863
31. Furche, T., Gottlob, G., Libkin, L., Orsi, G., Paton, N.W.: Data wrangling for big data: challenges and opportunities. In: Proceedings of the 2016 International Conference on Extending Database Technology (EDBT), pp. 473–478. (2016)
32. Harel, D.: Statecharts: a visual formalism for complex systems. Sci. Comput. Program. **8**(3), 231–274 (1987). https://doi.org/10.1016/0167-6423(87)90035-9
33. Hart, S.G., Staveland, L.E.: Development of NASA-TLX (Task Load Index): results of empirical and theoretical research. Adv. Psychol. **52**, 139–183 (1988)
34. Huot, S., Dumas, C., Dragicevic, P., Fekete, J.D., Hégron, G.: The MaggLite post-WIMP toolkit: draw it, connect it and run it. In: Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology (UIST'04), pp. 257–266. ACM, New York, NY, USA (2004). https://doi.org/10.1145/1029633.1029675
35. International Organization for Standardization: Ergonomics of human-system interaction—part 11: Usability: Definitions and concepts. Tech. Rep. ISO 9241-11:2018, ISO, Geneva, Switzerland (2018)
36. Jacob, R.J.: Using formal specifications in the design of a human-computer interface. Commun. ACM **26**(4), 259–264 (1983). https://doi.org/10.1145/358061.358067
37. Jadhav, A., Pramod, D., Ramanathan, R.: Comparison of performance of data imputation methods for numeric dataset. Appl. Artif. Intell. **33**(10), 913–933 (2019)
38. Jäger, S., Allhorn, A., Bießmann, F.: A benchmark for data imputation methods. Front. Big Data **4** (2021). https://doi.org/10.3389/fdata.2021.693674. https://www.frontiersin.org/journals/big-data/articles/10.3389/fdata.2021.693674
39. Kelleher, C., Wagener, T.: Ten guidelines for effective data visualization in scientific publications. Environ. Model. & Softw. **26**(6), 822–827 (2011). https://doi.org/10.1016/j.envsoft.2010.12.006

40. Liu, S., Andrienko, G.L., Wu, Y., Cao, N., Jiang, L., Shi, C., Wang, Y., Hong, S.: Steering data quality with visual analytics: the complexity challenge. Vis. Inform. **2**(4), 191–197 (2018)
41. Liu, Z., Heer, J.: The effects of interactive latency on exploratory visual analysis. IEEE Trans. Visual Comput. Graphics **20**(12), 2122–2131 (2014). https://doi.org/10.1109/TVCG.2014.2346452
42. Liu, Z., Zhang, A.: Sampling for big data profiling: a survey. IEEE Access **8**, 72713–72726 (2020). https://doi.org/10.1109/ACCESS.2020.2988120
43. Llama Team: The LLaMA 3 herd of models. CoRR **abs/2407.21783** (2024). https://doi.org/10.48550/arXiv.2407.21783
44. Manolopoulos, Y., Theodoridis, Y., Tsotras, V.J.: Advanced Database Indexing. In: Advances in Database Systems, vol. 17. Kluwer Academic Publishers (1999). https://doi.org/10.1007/978-1-4419-8590-3
45. Miller, R.B.: Response time in man-computer conversational transactions. In: Proceedings of the AFIPS '68 Fall Joint Computer Conference, pp. 267–277. AFIPS/ACM/Thomson Book Company, Washington D.C. (1968)
46. Minaee, S., Mikolov, T., Nikzad, N., Chenaghlu, M., Socher, R., Amatriain, X., Gao, J.: Large language models: a survey. CoRR abs/2402.06196 (2024). https://doi.org/10.48550/arXiv.2402.06196. https://doi.org/10.48550/arXiv.2402.06196
47. Moreland, K.: A survey of visualization pipelines. IEEE Trans. Vis. Comput. Graph. **19**(3), 367–378 (2013). https://doi.org/10.1109/TVCG.2012.133
48. Moritz, D., Howe, B., Heer, J.: Falcon: balancing interactive latency and resolution sensitivity for scalable linked visualizations. In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, pp. 1–11. Association for Computing Machinery (2019). https://doi.org/10.1145/3290605.3300924
49. Myers, B.A.: Separating application code from toolkits: eliminating the spaghetti of call-backs. In: Proceedings of the 4th Annual ACM Symposium on User Interface Software and Technology (UIST '91), pp. 211–220. ACM, New York, NY, USA (1991). https://doi.org/10.1145/122444.122467
50. Nah, F.F.H.: A study on tolerable waiting time: how long are web users willing to wait? Behav. & Inf. Technol. **23**(3), 153–163 (2004)
51. Nielsen, J.: Usability metrics: tracking interface improvements. IEEE Softw. **13**(6), 1–2 (1996). https://doi.org/10.1109/MS.1996.8740869
52. Nielsen, J., Mack, R.L.: Usability inspection methods. In: ACM SIGCHI Conference Companion on Human Factors in Computing Systems, pp. 413–414 (1994)
53. Norman, D.A.: The Design of Everyday Things, revised and, expanded The MIT Press, Cambridge, MA (2014)
54. Oney, S., Myers, B.A., Brandt, J.: InterState: a language and environment for expressing interface behavior. In: Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14), pp. 263–272. ACM, New York, NY, USA (2014). https://doi.org/10.1145/2642918.2647353
55. Paas, F., Renkl, A., Sweller, J.: Cognitive load theory and instructional design: recent developments. Educ. Psychol. **38**(1), 1–4 (2003)
56. Paas, F.G.W.C.: Training strategies for attaining transfer of problem-solving skill in statistics: a cognitive-load approach. J. Educ. Psychol. **84**(4), 429–434 (1992)
57. Parnas, D.L.: On the use of transition diagrams in the design of a user interface for an interactive computer system. In: Proceedings of the 1969 24th ACM national conference, pp. 379–385 (1969)
58. Pernici, B., Cappiello, C., Bono, C., Sancricca, C., Catarci, T., Angelini, M., Filosa, M., Palmonari, M., Paoli, F.D., Bergamaschi, S., Simonini, G., Mozzillo, A., Zecchini, L.: Sustainable quality in data preparation. ACM J. Data Inf. Qual. **17**(4:23), 1–33 (2025)
59. Rebuffi, S.A., Gowal, S., Calian, D.A., Stimberg, F., Wiles, O., Mann, T.: Data augmentation can improve robustness. Adv. Neural Inf. Process. Syst. (NeurIPS) **34**, 29935–29948 (2021)
60. Roh, Y., Heo, G., Whang, S.E.: A survey on data collection for machine learning: a big data-AI integration perspective. IEEE Trans. Knowl. Data Eng. **33**(4), 1328–1347 (2021)

61. Satyanarayan, A., Wongsuphasawat, K., Heer, J.: Declarative interaction design for data visualization. In: Proceedings of the 27th annual ACM symposium on User interface software and technology, UIST '14, pp. 669–678. Association for Computing Machinery, New York, NY, USA (2014). https://doi.org/10.1145/2642918.2647360. https://doi.org/10.1145/2642918.2647360
62. Schmidt, A., Elagroudy, P., Draxler, F., Kreuter, F., Welsch, R.: Simulating the human in HCD with ChatGPT: redesigning interaction design with AI. ACM Interact. **31**(1), 24–31 (2024). https://doi.org/10.1145/3637436
63. Shneiderman, B.: Response time and display rate in human performance with computers. ACM Comput. Surv. (CSUR) **16**(3), 265–285 (1984)
64. Shneiderman, B.: The eyes have it: a task by data type taxonomy for information visualizations. In: Proceedings of the IEEE Symposium on Visual Languages, pp. 336–343. (1996)
65. Shneiderman, B., Plaisant, C.: Designing the User Interface: Strategies for Effective Human-Computer Interaction, 5th edn. Addison-Wesley, Boston, MA (2010)
66. Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., Diakopoulos, N.: Designing the User Interface: Strategies for Effective Human-Computer Interaction, 6 edn, Pearson (2016)
67. Stolper, C.D., Perer, A., Gotz, D.: Progressive visual analytics: user-driven visual exploration of in-progress analytics. IEEE Trans. Visual Comput. Graphics **20**(12), 1653–1662 (2014). https://doi.org/10.1109/TVCG.2014.2346574
68. Strahl, J., Peltonen, J., Floréen, P.: Directing and combining multiple queries for exploratory search by visual interactive intent modeling. In: Human-Computer Interaction—INTERACT 2021, pp. 514–535. Springer International Publishing, Cham (2021)
69. Sulo, R., Eick, S., Grossman, R.: Davis: a tool for visualizing data quality. Posters Compend. InfoVis **2005**, 45–46 (2005)
70. Tory, M., Möller, T.: Human factors in visualization research. IEEE Trans. Visual Comput. Graphics **10**(1), 72–84 (2004)
71. Tufte, E.R.: The Visual Display of Quantitative Information, 2 edn, Graphics Press (2001)
72. Ulmer, A., Angelini, M., Fekete, J.D., Kohlhammer, J., May, T.: A survey on progressive visualization. IEEE Trans. Visual Comput. Graphics **30**(8), 6447–6467 (2024). https://doi.org/10.1109/TVCG.2023.3346641
73. Vancisin, T., Clarke, L., Orr, M., Hinrichs, U.: Provenance visualization: tracing people, processes, and transformations in historical records. Digit. Sch. Hum. **38**(3), 1322–1346 (2023). https://doi.org/10.1093/llc/fqad028
74. Visciòla, M.: Rethinking human-centered design: from automation to sustainable innovation. ACM Interact. **32**(4), 22–26 (2025). https://doi.org/10.1145/3733153
75. Waloszek, G., Kreichgauer, U.: User-centered evaluation of the responsiveness of applications. In: Human-Computer Interaction—INTERACT 2009, pp. 239–242. Springer, Berlin, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03658-3_26
76. Ware, C.: Information Visualization: Perception for Design, 4 edn, Morgan Kaufmann (2021)
77. Wu, E., Chen, Y., Mohammed, H., Huang, Z.: A decade of systems for human data interaction. Inf. Syst. **138**, 102689 (2026). https://doi.org/10.1016/j.is.2026.102689
78. Wu, X., Xiao, L., Sun, Y., Zhang, J., Ma, T., He, L.: A survey of human-in-the-loop for machine learning. Futur. Gener. Comput. Syst. **135**, 364–381 (2022). https://doi.org/10.1016/j.future.2022.05.014
79. Xu, W., Dainoff, M.: Enabling human-centered AI: a new junction and shared journey between AI and HCI communities. ACM Interact. **30**(1), 43–47 (2023). https://doi.org/10.1145/3571883
80. Yim, S., Yoon, C., Yoo, S., Jang, Y.: A mental workload estimation for visualization evaluation using EEG and NASA-TLX. In: Eurographics Conference on Visualization (EuroVis) Posters, (2022)
81. Zagermann, J., Pfeil, U., Reiterer, H.: Measuring cognitive load using eye tracking technology in visual computing. In: Proceedings of the Sixth Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization, BELIV '16, p. 78–85. Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2993901.2993908. https://doi.org/10.1145/2993901.2993908

82. Zhang, Z., Groth, P., Calixto, I., Schelter, S.: Directions towards efficient and automated data wrangling with large language models. In: 2024 IEEE 40th International Conference on Data Engineering Workshops (ICDEW), pp. 301–304 (2024). https://doi.org/10.1109/ICDEW61823.2024.00044

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (http://creativecommons.org/licenses/by-nc-nd/4.0/), which permits any noncommercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if you modified the licensed material. You do not have permission under this license to share adapted material derived from this chapter or parts of it.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Chapter 7
Key Insights and Future Opportunities for Sustainable Data Preparation



Barbara Pernici and Sonia Bergamaschi

Abstract The always-increasing amount of available data and the opportunity to use it in a large variety of systems to analyze trends, for decision making, for predictions, but also to support day-to-day work and human activities, call for larger and larger efforts in data preparation. Starting from the experience gained in the project "Discount quality for responsible data science: Human-in-the-Loop for quality data", we analyze its main outcomes and open challenges for future research.

7.1 Sustainability Challenges

The research developed in the "Discount quality" project has contributed to the advancement of data preparation methodologies by establishing the central role of sustainability in balancing data quality, efficiency, and environmental impact. In particular, the project has focused on two main aspects. On the one hand, a general framework for assessing sustainability was developed, considering not only energy efficiency as a sustainability parameter, but also human efforts and the purpose of the data preparation activities, considering their specific context, which can have a big impact on the choice of preparation methods. The project has shown that the adoption of strategies inspired by the circular economy model used in production contexts [3] can also support data preparation, introducing more effective data management practices, while at the same time reducing waste and computational costs. The integration of these strategies with the FAIR (Findability, Accessibility, Interoperability, and Reuse of digital assets) principles further reinforces a responsible, sustainable, and long-term approach to the use of digital resources. The "Discount

B. Pernici (✉)
Politecnico di Milano - DEIB, Milano, Italy
e-mail: barbara.pernici@polimi.it

S. Bergamaschi
Dipartimento di Ingegneria "Enzo Ferrari", Università di Modena e Reggio Emilia, Modena, Italy
e-mail: bergamaschi@unimore.it

© The Author(s) 2027

B. Pernici et al. (eds.), *Discount Quality for Responsible Data Science*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-032-29481-4_7

quality for data preparation" framework [2] is articulated through a set of complementary strategies: Rethink, to promote the creation of reusable datasets and shared tools; Reduce, to minimize resource consumption by eliminating redundant or unnecessary data; Reuse, to support the exploitation of existing datasets and data preparation pipelines; Repair, to enable the continuous cleaning and validation of data; and Refurbish, to update and enrich datasets to preserve their relevance over time. Taken together, these strategies provide a structured foundation for the adoption of more sustainable data preparation processes. An additional contribution of the project lies in the framework's ability to quantify sustainability dimensions associated with circular economy strategies. This capability provides a basis for the future development of data-driven decision support tools able to recommend whether data should be cleaned, transformed, reused, or discarded according to context-specific trade-offs. In more dynamic scenarios, such as streaming analytics, the framework may also support the real-time optimization of pipeline configurations, with the aim of balancing timeliness and resource consumption, for example, through the prioritization of on-demand cleaning operations. Another contribution of the project was the development of specific data preparation methods and techniques that were included within the developed frameworks. The researches that have been presented in the previous chapters have illustrated the focus of the project on preparing tabular data, and in particular developing a new taxonomy for quality dimensions relevant in this activity (Chaps. 2 and 6), a general on-demand approach, that has been studied both to select tables in the collection phase, to merge tables, and to identify on which entities preparation activities should be prioritized (Chap. 3) and visualization focused (Chap. 6). The design of data preparation pipelines has also been studied to select the most sustainable strategies for enriching information about entities, and for selecting which activities should be included in the data preparation pipeline (Chap. 4) and in which order, to reduce the computational and human efforts (Chap. 5).

The project started at a time when large language models had a large impact on research. In particular, in the project, we studied both how data preparation can benefit from using LLMs, developing different approaches to use them, and the characteristics of LLM models in relation to the tasks to be performed. On the other hand, data-centric AI and data-centric green AI have emerged as new paradigms in recent years, focusing on the importance of the quality of data when developing new AI models. While coming from different research areas and starting from different research methods, there is a convergence on similar goals between AI focusing on model development and the information systems and database research communities focusing on the quality of data, with an interplay of development steps of AI models and high-quality data preparation, typical of data-centric AI. As remarked in the introductory chapter, the rapid increase of AI approaches is resulting in a significant exploitation of natural resources, both in terms of energy consumption and also of water consumption for cooling and of rare-earth elements, which results in larger needs to develop sustainable approaches in this domain.

7.2 Future Opportunities

Future research in this area is needed to address several important aspects. Here, we identify several research directions that we consider key to future developments: investigation of data preparation in critical domains, the interplay of data quality improvement and model development in data-centric AI, the reuse of previous experience in data preparation pipeline development, and, finally, further developments in the theoretical efforts towards the conceptualization of sustainability assessment techniques in this area.

Further investigation of the sustainability framework for data preparation is needed in domains where sustainability requirements are particularly significant, analyzing the suitability of the different strategies. In IoT and edge computing environments, the adoption of Reuse and Reduce strategies in resource-constrained settings could contribute to lowering energy consumption and processing latency; in particular, new architectures, on-demand approaches, and ad-hoc data preparation pipelines can be exploited to improve sustainability. The framework proposed in the project can also serve as a reference model and be extended for sustainable data preparation practices in cloud environments to support dynamic resource allocation.

Data-centric artificial intelligence emphasizes engineering data when building AI systems. The implications of this approach are being discussed in the literature to analyze open challenges (see, for instance, [4]). Among the open issues, the main one is to design approaches that enable taming the complexity of data management, focusing on the most relevant aspects for improving the AI models. Data preparation is still considered one of the most demanding activities for developing AI models. As we have analyzed in this volume, critical issues concern the ability to understand data through visualization techniques, preserving the usability of the system, selecting data for training and input data for inference tasks with an appropriate level of quality, which may depend on the task itself. Given that many data preparation tasks are needed for transforming and cleaning the needed data, there is also a need to develop data preparation pipelines, while limiting the complexity of exploring alternative solutions with search methods. For all the above-mentioned tasks, it is clear that while automation can reduce development time, human involvement is needed to understand the implications and possible biases that could be introduced by the automation process. Further research on a clear definition of the tasks to be performed and assessment criteria for the implications of data management on the models is still at an initial phase.

The sustainability metrics introduced by the framework may provide the basis for recommending reusable pipeline fragments from shared repositories, in which actors exchange metadata on pipelines while preserving data sovereignty, reducing bias, and limiting the curation effort required to ensure privacy-aware data preparation.

In recent times, the sustainability aspects of IT have been discussed and analyzed [1]. In particular, while it is clear that there is an increasing impact of the use of IT and, in particular, of AI and generative AI on energy and natural resources consumption, the research on sustainability assessment techniques is still at an initial

phase. There is an inherent difficulty in measuring the impact of both the training and the inference phases, considering the complete lifecycle of the AI products and of the underlying infrastructure. In addition, the evaluations should consider not only efficiency aspects, but also economic and societal perspectives, to provide a global assessment of the sustainability of the development of data, software, and computing components of an IT system.

References

1. Chien, A.A., Schmeck, H.: Sustainability and computing - edited Special Section. Commun. ACM **68**(7), 43–107 (2025)
2. Pernici, B., Cappiello, C., Bono, C., Sancricca, C., Catarci, T., Angelini, M., Filosa, M., Palmonari, M., Paoli, F.D., Bergamaschi, S., Simonini, G., Mozzillo, A., Zecchini, L.: Sustainable quality in data preparation. J. Data Inf. Qual. **17**(4:23), 1–33 (2025)
3. Potting, J., Hekkert, M.P., Worrell, E., Hanemaaijer, A.: Circular economy: measuring innovation in the product chain. PBL Netherlands Environmental Assessment Agency, 2544 (2017)
4. Zha, D., Bhat, Z.P., Lai, K., Yang, F., Jiang, Z., Zhong, S., Hu, X.B.: Data-centric artificial intelligence: a survey. ACM Comput. Surv. **57**(5), 129:1–129:42 (2025). https://doi.org/10.1145/3711118

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (http://creativecommons.org/licenses/by-nc-nd/4.0/), which permits any noncommercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if you modified the licensed material. You do not have permission under this license to share adapted material derived from this chapter or parts of it.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

