

Robust and transparent Privacy-Preserving Record Linkage

Von der Fakultät für Mathematik und Informatik
der Universität Leipzig
angenommene

D I S S E R T A T I O N

zur Erlangung des akademischen Grades

DOCTOR RERUM NATURALIUM
(Dr. rer. nat.)

im Fachgebiet

Informatik

vorgelegt

von M. Sc. Elektrotechnik und Informationstechnik Florens Rohde
geboren am 28. Oktober 1990 in Berlin

Die Annahme der Dissertation wurde empfohlen von:

1. Prof. Dr. Erhard Rahm, Universität Leipzig
2. Prof. Dr. Thorsten Papenbrock, Universität Marburg

Die Verleihung des akademischen Grades erfolgt mit Bestehen der
Verteidigung am 15. Juni 2026 mit dem Gesamtprädikat *magna cum laude*.

Abstract

Personal data is collected and stored by many organizations, and this data is highly valuable for analyses in domains such as health research or fraud detection. In many cases, the data of multiple organizations has to be combined because no single data provider holds a sufficiently large number of samples to support statistically reliable conclusions or because the relevant pieces of information are under distinct data ownership.

Linking records from different sources requires a common global identifier, but such identifiers are rarely available. Therefore, records are compared and classified pairwise based on quasi-identifying attributes, such as first name, last name or date of birth, to assign common global IDs to records corresponding to the same real-world entity.

Data linkages may potentially harm the privacy of individuals by combining information that can be used against their interests. As a consequence, they are subject to many legal and organizational constraints. In most cases, the identifying information is not needed for the actual data analysis but only for the matching. Privacy-preserving record linkage (PPRL) methods are a promising approach to conduct such linkages without sharing identifying plaintext information between the data owners or to a third party, thereby following the data minimization principle and reducing the privacy risk. The data owners encode the identifying data and the matching is conducted on the encoded data only.

A variety of encoding techniques have been proposed, mostly employing similarity-preserving but irreversible transformations, thereby enabling fuzzy matching to achieve a higher linkage quality in the presence of minor discrepancies between duplicates. An attribute-level application of such encodings preserves the frequency distribution of values, which allows an adversary to recover plaintext values, e.g., by matching the most common encoded item to the most common real-world attribute. Record-level encodings are a recommended approach to reduce the risk of reidentification attacks but impose considerable limitations on the linkage algorithm. In this thesis, we focus on this encoding scheme and study its impact on the linkage outcome. We propose methods for improving the robustness and quality of the PPRL results in order to foster its consideration as a privacy-by-default alternative to conventional record linkage.

PPRL algorithms are typically applied to linkage problems where personal identifying attributes are used for comparison. Naturally, such datasets from real-world databases are rarely publicly available, which is why research results in this domain are often reported for a few datasets only. Therefore, we investigate the robustness of PPRL methods

with record-level encodings. To this end, we designed a modular evaluation framework that includes components for dataset creation, linkage execution, and subsequent result analysis. The approach supports comprehensive studies of different linkage strategies across a broad range of datasets including both curated subsets of existing real-world data and synthetically constructed linkage scenarios that reflect varying levels of data quality. For the majority of the evaluated scenarios, baseline PPRL methods lag behind a state-of-the-art deep learning entity resolution system that depends on the existence of labeled training data though.

Assigning weights to attributes is a fundamental component of most record linkage algorithms. Several encoding schemes have been introduced to embed these weights into record-level representations. However, a detailed investigation on how these weights influence the results in the PPRL context is missing. We propose and evaluate adaptive weighting strategies for record-level encodings with the goal of enhancing the robustness and overall quality of PPRL outcomes. Our findings indicate that this rather small modification in the encoding process leads to an improved and more robust linkage quality despite the potentially adverse effects arising from using inconsistent encoding parameters at the data owners.

Fine-tuning record linkage methods for high-quality outcomes requires training data that contains information about confirmed matches and non-matches. To address this, we propose a linkage protocol that allows labeling of record pairs in a multi-layer clerical review procedure, in which more information is gradually disclosed if needed. Only a fraction of the uncertain pairs has to be reviewed manually by a (human) oracle using masked display of selected attributes. The majority of labeling is done automatically using hardened attribute-level encodings based on a model trained in active learning cycles built upon the manual reviews. While adding complexity to the otherwise straightforward linkage protocol, the approach not only leads to improved linkage quality but also allows to control the linkage outcome with respect to the balance of false and missed matches and to increase the reliability of the otherwise unverified results.

Acknowledgments

First and foremost, I would like to express my special gratitude to my supervisor Prof. Dr. Erhard Rahm for his guidance and support over the years. The discussions on possible directions of research as well as his ability to pinpoint weaknesses in manuscripts at a glance and to anticipate reviewer comments were very helpful for improving the work. Without his encouragement to continue on this path, the completion of this thesis would not have been possible.

I would also like to thank all my colleagues in the database group and at ScaDS.AI for the friendly and funny coffee breaks that brightened many days. In particular, the discussions with and the support from my PPRL co-researchers, Dr. Martin Franke, Dr. Victor Christen, and Ziad Sehili, were of great help to me. I also thank Prof. Dr. Peter Christen and the colleagues at the Australian National University for the warm hospitality and interesting discussions. Managing the organizational challenges of contracts, business and seminar trips, and much more was only possible with the patient support by Andrea Hesse and the ScaDS.AI management assistant team, in particular Nicole Spieß and Madlen Bartholmeß. Thank you!

I would also like to thank Dr. Matthias Schmidt and Prof. Dr. Ulf-Dietrich Braumann for sharing their passion for science and guiding me onto the path of academic research. After finishing my master thesis, I was welcomed at ScaDS by Dr. Eric Peukert with all his enthusiasm. His support, together with that of Prof. Dr. Erhard Rahm, enabled me to work on the topic within several interesting projects.

I am grateful to my family and friends for their encouragement and understanding for my absence in numerous occasions over the years, and especially during the most demanding times. Last but not least, I would like to thank Bianca for her love, understanding and constant support. Her belief in me made this work possible.

Leipzig, 6. Februar 2026

Florens Rohde

Dissertation-related Publications

The following list is ordered ascending by publication year.

- Martin Franke, Marcel Gladbach, Ziad Sehili, **Florens Rohde**, and Erhard Rahm. ‘ScaDS research on scalable privacy-preserving record linkage’. In: *Datenbank-Spektrum* 19 (2019), pp. 31–40. DOI: [10.1007/s13222-019-00305-y](https://doi.org/10.1007/s13222-019-00305-y)
- **Florens Rohde**, Martin Franke, Ziad Sehili, Martin Lablans, and Erhard Rahm. ‘Optimization of the Mainzelliste software for fast privacy-preserving record linkage’. In: *Journal of Translational Medicine* 19.33 (2021). DOI: [10.1186/s12967-020-02678-1](https://doi.org/10.1186/s12967-020-02678-1)
- Ziad Sehili, **Florens Rohde**, Martin Franke, and Erhard Rahm. *Multi-Party Privacy Preserving Record Linkage in Dynamic Metric Space*. BTW 2021. Streaming, Dresden, Germany, 2021. DOI: [10.18420/btw2021-13](https://doi.org/10.18420/btw2021-13)
- Martin Franke, Ziad Sehili, **Florens Rohde**, and Erhard Rahm. ‘Evaluation of Hardening Techniques for Privacy-Preserving Record Linkage’. In: *Proceedings of the 24th International Conference on Extending Database Technology (EDBT)*. 2021, pp. 289–300. DOI: [10.5441/002/edbt.2021.26](https://doi.org/10.5441/002/edbt.2021.26)
- **Florens Rohde**, Martin Franke, Victor Christen, and Erhard Rahm. ‘Value-specific Weighting for Record-level Encodings in Privacy-Preserving Record Linkage’. In: *BTW 2023*. Dresden, Germany: Gesellschaft für Informatik e.V., 2023. DOI: [10.18420/BTW2023-21](https://doi.org/10.18420/BTW2023-21)
- Martin Franke, Victor Christen, Peter Christen, **Florens Rohde**, and Erhard Rahm. ‘(Privately) Estimating Linkage Quality for Record Linkage’. In: *Proceedings of the 27th International Conference on Extending Database Technology (EDBT)*. 2024. DOI: [10.48786/EDBT.2024.26](https://doi.org/10.48786/EDBT.2024.26)
- **Florens Rohde**, Victor Christen, Martin Franke, and Erhard Rahm. ‘Multi-Layer Privacy-Preserving Record Linkage with Clerical Review based on gradual information disclosure’. In: *BTW 2025*. Bamberg, Germany: Gesellschaft für Informatik e.V., 2025. DOI: [10.18420/BTW2025-18](https://doi.org/10.18420/BTW2025-18)
- **Florens Rohde**, Victor Christen, and Erhard Rahm. ‘SecUREmatch: Integrating Clerical Review in Privacy-Preserving Record Linkage’. In: *SIGMOD-Companion* ’25. Berlin, Germany, 2025. DOI: [10.1145/3722212.3725131](https://doi.org/10.1145/3722212.3725131)
- **Florens Rohde**, Victor Christen, and Erhard Rahm. ‘Exploring Privacy-Preserving Record Linkage: A Holistic Framework for Dataset Generation and Detailed Result Analysis’. In: *VLDB 2025 Workshop: 14th International Workshop on Quality in Databases (QDB’25)*. London, 2025

Contents

Abstract	III
Acknowledgments	V
Dissertation-related Publications	VII
1 Introduction	1
1.1 Motivation	1
1.2 Scientific Contributions	5
1.3 Structure	7
2 Background and Related Work	9
2.1 Record linkage	9
2.1.1 Data quality issues	10
2.1.2 Linkage process	11
2.2 Quality assessment	17
2.2.1 Binary classification evaluation	17
2.2.2 Blocking evaluation	19
2.3 Privacy-preserving record linkage	21
2.3.1 Protocol designs	21
2.3.2 Attack scenarios	23
2.3.3 Encoding methods	24
2.3.4 Linkage components	30
2.3.5 Privacy assessment	32
2.4 Application context	33
3 Robustness evaluation framework	37
3.1 Motivation	37
3.2 Service-oriented framework	40
3.2.1 Tag concept	41
3.2.2 Linkage problem creation	42
3.2.3 Linkage execution	45
3.2.4 Linkage result analyzer	47
3.2.5 Workflow example	48

3.3	Robustness evaluation	51
3.3.1	Linkage problem creation	51
3.3.2	Dataset comparison	53
3.3.3	Linkage methods	58
3.3.4	Results	61
3.4	Discussion	65
3.5	Conclusion	67
4	Value-specific attribute weight adaptation	69
4.1	Motivation	69
4.2	Related Work	71
4.3	Methods	71
4.3.1	Frequency-dependent weight adaptation	71
4.3.2	Weight estimation	75
4.3.3	Limited frequency information	76
4.3.4	Effects of attribute differences in duplicates	76
4.3.5	Handling missing values	78
4.4	Evaluation	78
4.4.1	Setup	79
4.4.2	Results	80
4.4.3	Robustness	83
4.5	Conclusion	88
5	Multi-layer protocol with clerical review	89
5.1	Motivation	89
5.2	Background and related work	90
5.3	Methods	93
5.3.1	Components	95
5.3.2	Privacy analysis	97
5.4	Evaluation	99
5.4.1	Setup	100
5.4.2	Results	102
5.5	Discussion	106
5.6	Demonstration	108
5.7	Conclusion	112
6	Conclusion and Outlook	113
6.1	Conclusion	113
6.2	Outlook	115

List of Figures	119
List of Tables	123
Bibliography	125
Bibliografische Angaben	143
Selbständigkeitserklärung	145

1

Introduction

1.1 MOTIVATION

In today's mostly digital world, nearly everybody is part of data collections by different organizations. Governments and their administrations need to manage tax and social benefit payments of their citizens. Companies process data of their customers to provide their services. Health care providers store and process medical information of patients to enable proper treatment. While many use cases are possible based on the data within a single organization, this does not generally hold true. State administrations need to gather data from multiple federal institutions, e.g., to prohibit multiple claims of financial aids. Companies want to verify the financial credibility of potential customers by querying a credit score database. Health care providers exchange medical information of patients with each other to avoid unnecessary examinations and with health insurances to ensure payment. Researchers often rely on datasets that comprise information from several domains, such as medical records along with data on living and working situation. Some organizations, in particular Big Tech companies, may have such a large user base and integration in multiple areas of peoples lives that they can conduct such analysis within their realm. However, they are not obliged to do so or share the results as it may not be in their strategic and financial interest.

In general though, such use cases require the integration of data originating from multiple organizations [17]. Health researchers in particular need data from multiple health care providers because the individual sources do not provide enough samples (patients) for statistically reliable analytical conclusions. They also need data from multiple sources when the required data items of patients are available only at distinct institutions, e.g., a general health care provider and a specific disease registry.

This thesis focuses on such use cases where data of individuals from multiple sources (or *data owners*) is combined. For that, it is essential to assign a globally unique identifier to source-specific records that correspond to the same real-world entity. Consequently, this task is referred to as *entity resolution* or *record linkage*. In a straightforward approach, the data owners share their respective datasets with a trusted institution, called *linkage unit*, which determines pairs or groups of records considered a *match* that are assigned a common global ID. Using these identifiers the respective data of matching entities can be combined or duplicates can be excluded. This linkage is rather trivial if the data sources already contain uniquely identifying attributes for each record such as a social security number (SSN). In most cases, such information is either not (or only for some records) available, or it is not usable due to regulatory restrictions. Therefore, the classification of record pairs as *match* or *non-match* typically relies on comparing suitable quasi-identifying attributes, e.g., names, date of birth, and address.

Data integration can potentially harm the privacy of individuals by processing information that can be used against their interest, like revealing a stigmatized disease. Thus, the conduction of such a linkage is subject of many legal and organizational constraints which aim to reduce the privacy risk [20, 91, 101]. *Privacy-preserving record linkage (PPRL)* addresses this challenge and allows to compare and classify records pairs without sharing sensitive plaintext information between the data owners or a third party. It has been an active research subject for the last decades [50, 148]. To protect the identities of the records, the quasi-identifying attributes are encoded before being sent to the linkage unit which performs the linkage on the encoded data only. A variety of encoding techniques have been proposed [20]. They mostly employ a similarity-preserving but irreversible transformation allowing fuzzy matching to achieve a higher *linkage quality* in the presence of minor discrepancies between duplicates. Beyond these perturbation-based PPRL techniques, secure multiparty computation (SMPC) methods have been employed for this purpose, offering a higher degree of data protection at the cost of substantially increased computational overhead.

In many cases a consent of the affected persons is required [65]. This consent can also comprise allowance of linkage on plaintext. Studies on the attitude towards secondary research on patient data show that privacy concerns are among the most influential factors [1, 96, 118]. It is therefore reasonable to assume that the application of privacy-preserving computation techniques can enhance acceptance, even in situations where they are not strictly necessary. Especially when sensitive (health-related) information is involved, individuals are more likely to approve of its use if their privacy is protected during the linkage process.

Furthermore, there are data protection regulations in place that manifest individuals' fundamental right to privacy. Article 5(1)(c) of the General Data Protection Regulation (GDPR) of the European Union states [35]:

Article 5 – Principles relating to processing of personal data

1. Personal data shall be:

- (c) adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed ('data minimisation');

In the context of data linkage, this data minimization principle can be interpreted as follows: If identifying plaintext attributes are not necessary for the purpose of the processing (in this case, record linkage), they should not be used. Therefore, privacy-preserving and privacy-enhancing techniques should always be prioritized. There is a caveat though: if PPRL is unable to reach the level of linkage quality necessary for the intended use of the data or considered a disproportionate effort, it can be argued that it does not fall under this regulation.

The most commonly employed perturbation-based PPRL method using record-level Bloom filter encodings is very simple when compared to the plethora of methods available in conventional record linkage [20, 99]. All attributes of a record are hashed into a single joint Bloom filter, that is compared to another Bloom filter representing a match candidate. Pairs are classified based on the similarity score using a threshold. This approach is sensitive in particular to the chosen threshold, the attributes included with their respective weights, and the quality of the underlying data.

Data custodians may have concerns against PPRL due to privacy attacks and the potentially lower linkage quality. The former are possible in particular when similarity-preserving encodings are used that share exploitable frequency patterns with the underlying plaintext [154]. Hence, there is an inherent possibility of successfully aligning input and output values for reidentification attacks on perturbation-based methods, i.e., utilizing Bloom filter encodings. These PPRL approaches do not provide formal security guarantees, but increase the effort required to carry out attacks and lower the likelihood of success, making it less appealing for malicious adversaries or curious data custodians to peek into the plain data. SMPC techniques reduce this risk even further, but they generally require efforts that may be seen as disproportionate to the benefits.

In addition to demonstrating resistance against attacks, PPRL methods need to achieve high linkage quality as errors can lead to distortions in the integrated data and consequently to incorrect analytical conclusions. The result of a record linkage process – conventional or privacy-preserving – contains errors in all non-trivial applications, because certain potential duplicates are difficult to assess. Some true matches are missed, for example when attributes differ greatly or have been replaced, and some false links between non-matching records occur, for example due to agreement on frequent attributes.

The focus of this work is to enhance the linkage quality of PPRL while still providing a high level of privacy protection. In the following, we outline the primary objectives addressed by this thesis.

Robustness To ensure high linkage quality in real-world applications with unknown and varying dataset characteristics, the linkage method should be capable of delivering such results across a wide range of (benchmark) datasets. Two possible approaches come to mind: The linkage method can be designed to recognize and appropriately handle various kinds of discrepancies between duplicates, for instance transposed attributes and typographical mistakes. However, in the PPRL context that typically implies a lower level of privacy protection, i.e., by using attribute-level encodings, to support such flexible matching strategies.

Alternatively, one could employ an adaptive linkage protocol in which the parameters of the encoding are determined by the actual composition of the given dataset. For example, if the records are automatically integrated at the data owners from a reliable source, such as an electronic health card, matching records may be from different points in time. In this scenario, typographical errors are not to be expected, but changes in address and possibly the last name. The encoding technique could then be tailored to tolerate these types of differences.

Transparency Typically, record linkage is merely one link in the processing chain of a data integration task. Any errors and uncertainties at this stage should be made visible to the downstream components, for example, so that less certain links receive a lower weight in subsequent analyses. If confidence scores are included in the linkage results, the users can determine whether they would rather risk accepting false links and missing true matches, or the other way around. Including a description of the datasets in the linkage report is regarded as good practice, as it enables readers to evaluate how difficult the linkage task is and, consequently, what level of linkage quality can be expected [87].

Transparency is also important in the evaluation of linkage algorithms. Typically, quality measures are reported for benchmark datasets that comprise a variety of errors. Data custodians cannot assess whether can expect comparable quality for their own linkage problem when it differs in terms of available attributes, patterns of missing data, and other characteristics. The evaluation should provide fine-grained insights into the performance of a linkage method, allowing to examine strengths and weaknesses related to specific error types or other dataset characteristics.

Reliability Instead of enforcing a binary decision as match or non-match, one can also introduce a third class for uncertain links. A manual human-based clerical review of these links can help to resolve the ambiguities. For example, a record pair with a different address, but same name and date of birth is likely a match whereas a record pair with several small but meaningful differences across multiple attributes would be labeled as a non-match by a human. However, when the classification model cannot determine whether the observed differences stem from one or several attributes – as it is the case with record-level encodings – this can lead to misclassifications. In the PPRL context, conducting a clerical review that requires displaying plaintext attributes to

human oracles is generally not feasible [20]. In consequence, there is a need for research on privacy-preserving techniques for classifying uncertain record pairs, enabling reliable and high-quality linkage results.

Control The privacy-utility tradeoff principle [81] applies to PPRL as well. A higher level of protection against cryptanalysis usually leads to lower linkage quality, because the methods used to obfuscate exploitable patterns in the encoded data also make it more difficult to reliably distinguish patterns that characterize a match from those that denote a non-match. The linkage method cannot be optimal with regard to both aspects. It depends on the intended use case which linkage quality is considered *good* or *sufficient*. Therefore, it is desirable to employ adaptive linkage protocols that provide a high level of protection in general, but can be adjusted as needed to reach the desired level of confidence in the outcome.

1.2 SCIENTIFIC CONTRIBUTIONS

In this section, we outline the scientific contributions made in this thesis.

EVALUATION FRAMEWORK FOR PPRL

To support the objectives of robustness and transparency, a modular evaluation framework for PPRL was developed. It includes components for dataset creation, linkage execution, and subsequent result analysis. The approach enables comprehensive assessment of PPRL strategies across a broad range of datasets including both curated subsets of existing real-world data and synthetically constructed linkage scenarios that reflect varying levels of data quality. Natural or intentionally induced data characteristics are traced throughout the linkage process to enable analysis of influencing factors on the linkage quality. The proposed framework is employed to study the robustness of record-level encodings, in particular with regard to attribute weights. A suite of benchmark linkage problems with variable composition is created and described, focusing on data characteristics with particular influence on the parametrization of the linkage algorithm such as value frequency distributions. The results indicate that the weights have a strong impact on the linkage quality in most scenarios. This highlights the need for further research on their application in the PPRL context. The comparison with a state-of-the-art deep-learning-based plaintext entity resolution system reveals that the PPRL method has considerable shortcomings. The framework was presented at the *14th International Workshop on Quality in Databases at the VLDB 2025* and published in the workshop proceedings [120].

VALUE-SPECIFIC WEIGHTING FOR RECORD-LEVEL ENCODINGS

Assigning weights to attributes is a fundamental component of most record linkage algorithms. Although encoding schemata were previously proposed for PPRL to incorporate such weights in record-level Bloom filters, the estimation of the weight values and their impact were not studied in detail. These methods are examined and weight adaptation approaches based on individual value frequencies were integrated. Key challenges are that this linkage approach does not support dynamic assignment of weights during classification, as well as the restricted availability of the frequency distribution of the datasets. The evaluation on real-world datasets in general shows that frequency-dependent weighting schemes improve the linkage quality as well as the robustness with regard to threshold selection. While the CLK-RBF approach [147] is rather tolerant to weight variations when comparing record pairs, this is not true for RBF encodings [32] that are composed of attribute-level encodings. Overall, the findings indicate that the approach leads to an increased linkage quality despite drawbacks due to the inconsistent encoding parameters. This work was presented at the *BTW 2023* and published in the conference proceedings [122].

MULTI-LAYER LINKAGE PROTOCOL WITH CLERICAL REVIEW USING GRADUAL INFORMATION DISCLOSURE

The parametrization of record linkage methods often requires training data with information on known matches and non-matches for high quality results. This is particular true for PPRL approaches that (are restricted to) use a simple threshold for classification. A linkage protocol which includes acquisition of labeled pairs was proposed using a multi-layer clerical review process with gradual information disclosure. Only a fraction of the uncertain pairs is reviewed manually by a (human) oracle using a masked display of selected attributes. The majority of labeling is done automatically using hardened attribute-level encodings by utilizing a model trained in active learning cycles based on the manual reviews. While adding complexity to the otherwise straightforward linkage protocol, it allows to resolve uncertain match candidates and control the linkage outcome with respect to the balance of false and missed matches and to gain confidence in the otherwise unverified results. The approach was presented at the *BTW 2025* and published in the conference proceedings [123]. Furthermore, the work was demonstrated at the *SIGMOD 2025* and published in the companion proceedings [121].

The concepts, implementations, experiments and original manuscripts of these publications were realized by Florens Rohde. The manuscripts were revised by Florens Rohde and the respective co-authors. The research process was supervised by Prof. Rahm.

1.3 STRUCTURE

This doctoral thesis is organized as follows:

Chapter 2 provides the required background knowledge on conventional and privacy-preserving record linkage, including quality and privacy measures. To further clarify the objectives of this thesis, it also includes descriptions of PPRL use case scenarios along with their respective requirements.

Chapter 3 introduces the evaluation framework and implementation for studying PPRL methods which is used throughout this thesis. Furthermore, existing PPRL approaches are benchmarked with this framework to assess their robustness in particular with regard to input dataset variations.

Chapter 4 studies methods for adapting important parameters of the encoding phase of PPRL protocols, namely attribute weights, based on value frequencies. The proposed scheme is applied to attribute-level and multiple record-level encoding methods, and it is also tested in constrained settings where only limited information is available.

Chapter 5 proposes a multi-layer PPRL protocol using masked clerical review with minimal information disclosure. The labels are incorporated in an active learning process to reduce the number of reviews while training a classification model for the majority of uncertain links.

Chapter 6 gives a summary of the contributions of this thesis and outlines directions for future work.

2

Background and Related Work

In this chapter, the required background knowledge for the contributions of this dissertation is provided. The general record linkage workflow and its privacy-preserving counterpart as well as approaches for assessing linkage quality and privacy are described. Finally, application scenarios for PPRL are outlined.

2.1 RECORD LINKAGE

The notation $\mathbf{R}$ is used for datasets containing all records of a specific linkage problem. In a scenario involving S sources, $\mathbf{R}_s$ denotes a database owned by s so that $\mathbf{R} = \bigcup_{s \in S} \mathbf{R}_s$. Each $\mathbf{R}_s$ consists of records r which comprise a set of attributes $\mathbf{A} = \{a_0, a_1, \dots, a_{i-1}\}$, that contain granular information, e.g., first name, last name or date of birth.

Record linkage refers to the task of identifying records that correspond to the same real-world entity, such as a specific individual. Unlike *deduplication*, which aims to detect duplicates *within* a single database $\mathbf{R}_s$, record linkage is concerned with finding matching records *across* multiple data sources $\mathbf{R}_s$. The term *entity resolution* is frequently used interchangeably with record linkage, though occasionally, it serves as an overarching term encompassing both record linkage and deduplication.

Unique record identifiers would enable a join-like operation. For example, social security numbers (SSN) and tax identification numbers are very robust identifiers for individuals. Nevertheless, these attributes are not universally available but specific to particular domains. As such, their use is legally restricted to specific applications and processes, at least in certain countries like Germany. The health insurance number for example is utilized for accounting between health providers and the insurance and typically cannot be used for other purposes such as secondary research [91]. Even if such

an unique identifier can be used, it is likely not present across all data sources or in every record within those sources [97]. For instance, when establishing links across domains like between a hospital and a corporation or across different companies, these lack both an SSN and a tax identification number.

Due to these obstacles, records are compared pairwise based on their (quasi-)identifying attributes, such as first name, last name and date of birth, and classified as match or non-match. Databases often face data quality issues such as missing or outdated values, and errors from data collection [17, 107, 110]. The main challenge in record linkage is detecting duplicates having these issues, which makes exact matching impossible, while avoiding wrong classifications of similar but distinct records. Thus, in the next section we explore the data quality challenges to record linkage.

2.1.1 DATA QUALITY ISSUES

Various approaches exist that propose terminology and frameworks to describe data quality dimensions [3, 68, 155]. Common categories include completeness, correctness, concordance, plausibility, and timeliness. However, a thorough discussion of these dimensions is out of scope for this thesis. Thus, we focus on data quality issues that arise when linking records between heterogeneous systems [17, 110]. Table 2.1 illustrates some examples of such problems.

First name	Last name	Date of birth	Address	Quality issue
Johannes	Schulze	12/02/1972	28755 Bremen	
Joha m es	Schul t ze	12/02/1972	28755 Bremen	Minor error (OCR/typo/ spelling variation)
Johann a	Schulze	12/02/1972	28755 Bremen	
Johannes	Schulze	02.12. 1972	28755 Bremen	Inconsistent format
Schulze	Johannes	12/02/1972	28755 Bremen	Exchanged attributes
Johannes	Schulze	12/02/1972		Missing attributes
Johannes	Schulze	12/02/1972	10117 Berlin	Attribute replacement
Johannes	Theuerkauf	12/02/1972	28755 Bremen	
Johannes Paul	Schulze-Theuerkauf	12/02/1972	28755 Bremen	Extended attributes

Table 2.1: Common data quality issues in record pairs: A given record (first row) may suffer from a variety of possible differences when compared to corresponding records making it challenging to identify these matches while detecting meaningful differences indicating a non-match.

Minor errors, such as typographical (typo) and optical character recognition (OCR) mistakes are often introduced during manual entry or automated text recognition. Small spelling variations may occur in corresponding records, e.g., due to multiple possible transcriptions from other alphabets like Cyrillic. When attempting to accommodate match tolerance for these variations, it could also result in incorrect matches with similar values, such as male and female name variants like *Johannes* and *Johanna*.

Inconsistent formatting frequently arises in systems with differing or missing data entry standards and database constraints. These problems may also occur within records of a single system, due to erroneous schema modifications or human mistakes. A common example is the date format which may differ due to distinct language standards of the databases to be linked.

Exchanged attributes occur when field values are swapped. A typical example for exchanged attributes are first and last name which are also known as forename or given name and surname or family name. While the initial terms implicitly refer to the positions of the components, the latter better describe the meaning of the components. In some non-western countries such as Vietnam the family name is written first which can easily result in such errors.

Missing attributes reflect incomplete records, where some fields are blank. Possible reasons could be that the value does not exist, e.g., because a person has no middle name, or that an attribute such as the place of birth is not known to a data owner. This data completeness problem can lead to missed matches when agreement on a given attribute would strongly indicate a duplicate, and it can also cause incorrect links when the absent attribute would, in fact, differ between the records.

Attribute replacements involve one value being substituted with another due to real-world changes, for example, a change of address when an individual moved or last name in case of marriage/divorce. This would not be problematic for record linkage if the sources are updated in a time-synchronous way but this is usually not the case. Data owners often have record entries for individuals from different points in time, resulting in such discrepancies.

Extended attributes represent cases where additional information is provided, such as multiple first names, a full middle name instead of the initial or compound last names. As one attribute is a substring of the other, it is a rather minor problem for record linkage methods. However, these data characteristics are again culture-dependent and have to be treated accordingly. For instance, middle name(s) usage and abbreviation is more common in US American culture than in Germany.

2.1.2 LINKAGE PROCESS

In this section, we describe the common steps involved in record linkage processes [17], as illustrated in Figure 2.1 for a scenario with two databases ($S = 2$).

PREPROCESSING

First, the databases are preprocessed to facilitate further use in the linkage algorithm, including schema matching, normalization and deduplication [17].

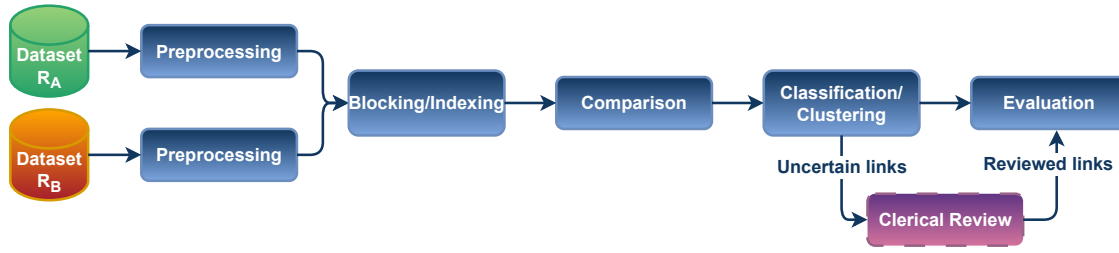


Figure 2.1: Record linkage workflow for matching two databases with the core steps preprocessing, blocking/indexing, comparison, classification, evaluation, and potentially clerical review (adapted from [20]).

Each data source R_s may have its own database schema. Thus, schema matching is a precondition in many record linkage processes to align the attributes of different sources for comparison later in the process [110]. As illustrated in Figure 2.2, the databases may exhibit a single or multiple attribute(s) for the name components as well as different column names for the same semantic content (*date of birth*). The database specific schemata are mapped to common schema for subsequent unified processing.

Data cleaning often comprises imputation of missing values, normalization, e.g., by replacing special characters such as German umlauts, or by enforcing a common date format. It should be noted that the selection of data cleaning methods should be carefully considered as it may also have detrimental effects on the linkage quality by eliminating information which is meaningful for distinguishing similar but non-matching records [116].

Each source database may contain multiple records for an entity. This could be due to insufficient data entry constraints which requires internal deduplication. Furthermore, a data owner may deliberately store multiple representations for known entities from different subsystems or from record history. Having *clean* sources without duplicates enables application of certain post processing methods as discussed below. Nevertheless, incorporating several distinct representations of a given entity from the same source can likewise enhance the quality of the linkage. For instance, if R_A includes two representa-

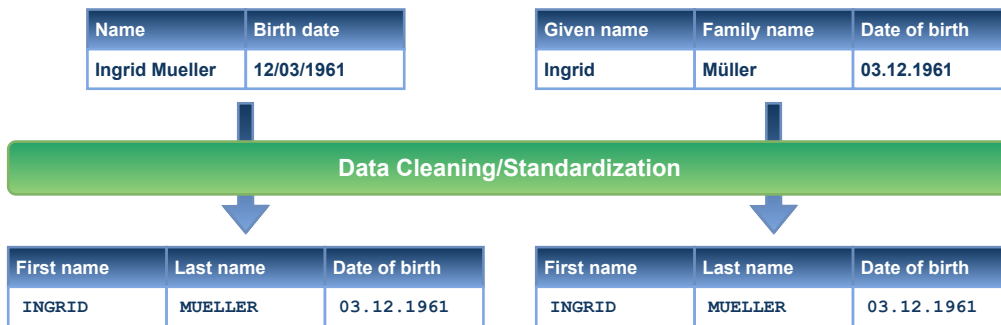


Figure 2.2: Example of data normalization for record preprocessing: A combined name field is split into its components and the source schema is aligned with a standardized schema. The date format is unified and umlaut characters are replaced.

tions of the same entity from different years, such as 2020 and 2004, and the data in $\mathbf{R}_B$ was collected in 2003, the older record in source A is likely to exhibit greater similarity to the records in B, thereby simplifying the linkage process.

BLOCKING AND FILTERING

A performance issue in record linkage is that the pairwise comparison of all records in the datasets scales poorly. For $S = 2$, the number of comparisons grows quadratically with the database sizes. Blocking is a common technique to reduce the number of match comparisons [16]. The standard blocking approach partitions the records according to a function on the values of selected fields, returning blocking keys. The similarity computation is then restricted to pairs of records from the same partition, i.e., records sharing a common blocking key.

A frequently used approach for plaintext matching is phonetic blocking, e.g., based on the Soundex encoding [95]. Phonetic encoding functions are typically applied on name attributes and aim to produce the same output for input values with a similar pronunciation (even with typographical variations or errors). For instance, the Soundex value for both names 'Sara' and 'Sarah' is S600. Since the first letter of the attribute value is preserved in the Soundex code, variations at the beginning of a name, e.g., 'Zara' (Z600) cannot be compensated. Thus, a single blocking key strategy may lead to a large number of missed matches if they happen to disagree on the selected blocking criteria. Therefore, it is common to combine multiple blocking strategies, allowing in-detail comparison if at least one of the blocking keys of records agree. However, this approach can result in the same pair being compared multiple times, making it necessary to deduplicate the pair candidates generated by the different blocking strategies.

Figure 2.3 shows an example of standard blocking where two blocking key functions are used. For pairs to be selected as match candidates, they have to agree on the initials of first and last names, the year of birth or both.

	PHILIPP WOLF 18.03.1981	INGRID MUELLER 03.12.1961	PHILINE WOLFF 18.03.1981
PHILIP WOLF 18.03.1982	PW		PW
HERIBERT KAISER 03.01.1961		1961	
INGRID MUELLER 03.12.1961		IM 1961	

Figure 2.3: Example of standard blocking using two strategies for key generation: Initials of first and last name (yellow, orange) and full year of birth (green). This reduces the number of comparisons from 9 to 4.

Filtering uses a similarity or distance measure and removes pairs that cannot reach a predefined threshold, e.g., by employing record features like length or prefix [135]. We explain metric space filtering briefly in Section 2.3.4. Both approaches, blocking and filtering, result in a set of match candidate record pairs which are compared in detail for subsequent classification.

COMPARISON

The selected record pairs are compared, often separately for each of the corresponding attributes. A variety of similarity measures for strings and other data types have been proposed [17, 140], as outlined in Figure 2.4. Common measures to compare names are edit distance, also known as Levenshtein edit distance, or Jaro-Winkler similarity. For numeric attributes, it is advisable to rely on numeric comparison methods that measure numerical distance, rather than on string-based similarity measures.

Besides straightforward attribute-by-attribute comparison, some linkage methods also make use of exchange groups. These groups contain attributes that may be mistakenly exchanged, e.g., first and last name, which are compared interchangeably [123].

The comparison phase produces an output similarity vector comprising one or more features describing the closeness of the record pairs. Typically, the similarity functions produce normalized values within the interval $[0, 1]$.

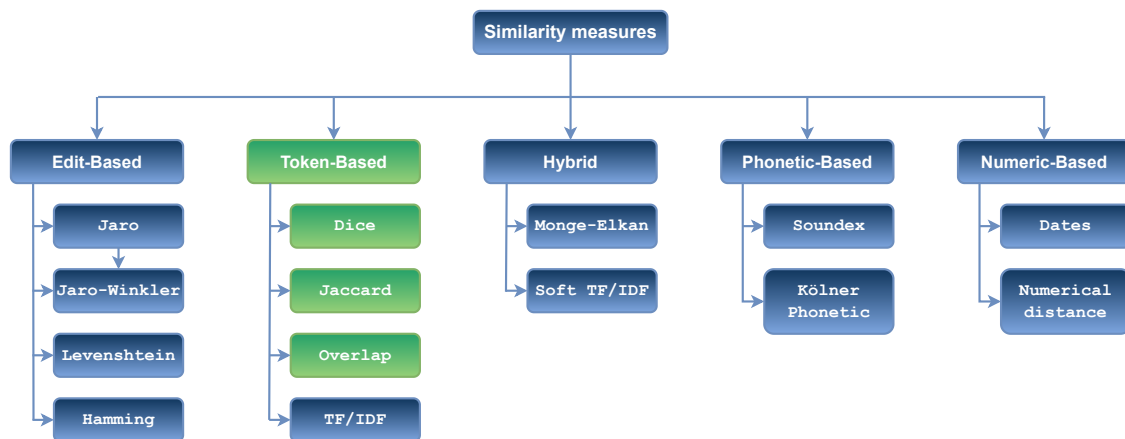


Figure 2.4: Overview of common similarity measures for record linkage: Edit-based (compare characters), Token-based (compare words or tokens), Hybrid (combine token and character similarity), Phonetic-based (compare sounds), and Numeric-based (compare numbers).

CLASSIFICATION

For each record pair, a classification as a match or non-match is done based on its feature vector. Different approaches for such classification exist, as outlined in this section.

Rule-based classification Data custodians may implement a set of rules that describe what are considered match criteria [17]. For example, records may need to fully agree on at least one of the attribute sets {Social Security Number}, {First name, last name, date of birth}, {First name, date of birth, full address} to be classified as a match.

Probabilistic classification Classification of two records r_a and r_b can also be based on their overall similarity $\text{sim}(r_a, r_b)$ that has to exceed a certain threshold t . Alternatively, a threshold range at the decision boundary may be implemented to select uncertain links which are resolved by clerical review.

The record similarity can be calculated as the weighted sum of the normalized similarities of all attributes pairs (a_i, b_i) [25].

$$\text{sim}(a, b) = \frac{\sum w_i \cdot \text{sim}(a_i, b_i)}{\sum w_i}, \quad (2.1)$$

A key challenge is the determination of these attribute weights. Fellegi and Sunter proposed a probabilistic record linkage approach based on dynamic weightings of attributes when comparing records [36]. It is the basis for many record linkage approaches and is still frequently used and adapted [17, 63, 123]. The method grounds on two aspects which influence the importance of each attribute pair for the overall similarity: reliability and discriminatory power. Consequently, the m - and u -probabilities are defined as

$$m_i = P(a_i = b_i | r_a \equiv r_b) \quad (2.2)$$

$$u_i = P(a_i = b_i | r_a \not\equiv r_b) \quad (2.3)$$

where r_a and r_b are records from datasets $\mathbf{R}_a$ and $\mathbf{R}_b$, and a_i and b_i are the values of corresponding attributes i of these records. $\equiv$ indicates that two records are a match, whereby $\not\equiv$ that they are a non-match.

The m -probability defines the likelihood that two *matching* records have the same value for attribute i . Ideally, $m_i = 1$ if there is full agreement among all matches for attribute i which is the case only when the attribute pairs i do not contain any errors. In practice, the value is lower due to the aforementioned data quality issues. For instance, if 20 % of the duplicates have a non-equal value due to any reason, then $m = 0.8$.

In contrast, the u -probability specifies the chance that two *non-matching* records agree on values for attribute i . The probability score decreases when an attribute has a wide range of possible values that occur with roughly the same likelihood. When an attribute has only two possible and equally likely values, then $u = 0.5 \cdot 0.5 + 0.5 \cdot 0.5 = 0.5$. However, the likelihood u of a random agreement for values is influenced by their frequency, with common values having higher scores than rare ones.

Using the m - and u -probabilities, the weight w_i for attribute i is determined as

$$w_i = \begin{cases} w_m = \log_2 \left(\frac{m_i}{u_i} \right) & \text{if } a_i = b_i \\ w_u = \log_2 \left(\frac{1-m_i}{1-u_i} \right) & \text{if } a_i \neq b_i \end{cases} \quad (2.4)$$

Originally, an overall similarity weight $w = \sum w_i$ is computed based on the positive w_m and negative w_u of all attributes and has to reach a certain minimal score for the record pair to account as a match. In order to get a positive weight for the normalized weighted aggregation in Equation (2.1) one can use either only w_m [123] or the range $w_m - w_u$ [32].

The model relies on the assumption of conditional independence of the attributes which is typically not true and results in biased parameter estimations. Nevertheless, research has shown that the effect on the record linkage quality is rather low [160]. The study further derives the conclusion that incorporation of conditional dependencies is important when the true overlap is low and the conditional dependence within the non-matches is dominant. If the discriminating power of the matching fields is high, consideration of conditional dependencies is not required in general.

Learner-based classification Supervised machine learning takes a different approach for classification by fitting the model parameters using labeled samples [99]. Thus, it relies on training data consisting of tuples of (similarity) feature vectors and corresponding target predictions (*match* and *non-match*). The typical imbalance of these classes in the record linkage domain requires its consideration when selecting the training dataset [38]. Ensemble learners combine multiple weak learners for improved linkage quality. A well-known example is Random Forest (RF) [12] based on decision trees which has been widely used in record linkage.

Advancements in deep learning techniques, which pioneered in computer vision tasks, have also yielded encouraging outcomes in the field of record linkage [99]. These approaches avoid the need for manual feature engineering (in particular selecting similarity functions in the record linkage domain) by transforming records into multidimensional vectors, or embeddings. Transformer models are commonly pretrained on extensive external datasets like Wikipedia, allowing them to incorporate broad world knowledge into the process, which often results in higher linkage quality [100].

Traditional supervised methods necessitate training datasets that are expensive to produce. Active learning strategies mitigate this issue by choosing an appropriate subset of samples for labeling, thereby avoiding the need to annotate the entire dataset [99].

CLUSTERING/POSTPROCESSING

The outcome of the previous step are *tuples* (r_i, r_j, sim) . In multi-party linkage scenarios with $S > 2$, matching pairs are clustered to group the records, e.g., based on transitive closure, for assignment to a joined unique entity [125]. Under the assumption of *clean*, or duplicate free, sources, a postprocessing step can be employed that removes links when more likely match candidates exists [43]. Figure 2.5 shows an example of such a postprocessing step based on record-level similarity scores.

	PHILIPP WOLF 18.03.1981	INGRID MÜLLER 03.12.1961	PHILINE WOLFF 18.03.1981
PHILIP WOLF 18.03.1982	0.93		~~0.81~~ (Postprocessing)
HERIBERT KAISER 03.01.1961		0.24 ($< \text{threshold}$)	
INGRID MUELLER 03.12.1961		1.00	

Figure 2.5: Example of linkage result postprocessing using a similarity threshold of 0.7 and under the assumption of duplicate-free/clean data sources which allows to discard links when a more likely match candidate exists.

2.2 QUALITY ASSESSMENT

The term *ground truth* refers to the verified labels used as a benchmark to evaluate the performance of a classifier. It represents the (real-world) truth against which the predictions of a model are compared. Ground truth may be specified either in terms of expected matching record pairs or by defining cluster of records from different sources.

2.2.1 BINARY CLASSIFICATION EVALUATION

Binary classifiers for record linkage tasks produce one of two labels for each pair, match or non-match. Consequently, we refer to ground truth pairs as True Matches (TM) and True Non-Matches (TNM).

Following [30], we use the term *match* to denote the true relationship between a pair of records according to ground truth, *link* to indicate their predicted/assumed relationship, and *agreement* to express their similarity with respect to matching variables.

The confusion matrix as illustrated in Figure 2.6 summarizes the results of classifications by showing the counts of all four possible outcomes: True Positives (TP, correctly predicted TM), True Negatives (TN, correctly predicted TNM), False Positives (FP, incorrectly predicted TNM), and False Negatives (FN, incorrectly predicted TM).

		Ground Truth	
		True Match (TM)	True Non-Match (TNM)
Classification Results	Predicted Link	True Positive (TP)	False Positive (FP)
	Predicted Non-Link	False Negative (FN)	True Negative (TN)

Figure 2.6: Confusion matrix for evaluating binary classification results.

Based on these values a variety of quality measures can be computed, of which the following are commonly used in record linkage tasks.

In the context of record linkage, the number of non-matches in general exceeds that of matches. Consequently, any measure that relies on the count of true non-matches, like accuracy, becomes disproportionately sensitive to this class. In record linkage tasks, the primary interest typically lies with the matches. Moreover, the count of non-matches is influenced by the chosen blocking strategy, leading to the possibility of artificially inflating accuracy by employing an ineffective strategy that results in a large number of highly improbable matches. Therefore, we focus on the following measures for our binary classification task: Recall measures the proportion of found true matches from all true matches. Precision measures the proportion of found true matches from all found matches. These two measures are largely influenced by the choice of the classification threshold whereby a lower threshold usually results in increased recall and lower precision and vice versa. It is desirable to have a single score which can be used by optimizers when tuning the parameters. The F1 score is a measure which combines both scores using their harmonic mean.

$$\text{Recall: } R = \frac{TP}{TP + FN} = \frac{TP}{TM} \quad (2.5)$$

$$\text{Precision: } P = \frac{TP}{TP + FP} \quad (2.6)$$

$$\text{F1 score: } F = \frac{2 \cdot R \cdot P}{R + P} \quad (2.7)$$

While being a popular quality measure for record linkage tasks, it has been noted that the F1 score actually is not well suited for comparing results of different datasets or different methods [19, 57]. The authors show that the weight of recall and precision implicitly depends on the linkage outcome, namely the values of the confusion matrix. Stable weights are achieved when each method classifies the same number of pairs as matches which means that $TP + FP = \text{const}$. This can be realized by adapting the threshold for each method accordingly. A suitable weight of recall and precision depends on the use

case. A general approach is the equal weighting which is achieved when the number of predicted matches is on par with the number of actual matches ($TP + FP = TP + FN$) [57].

Furthermore, the F1 score is criticized for being a pragmatic measure without a plausible justification for its use of the harmonic mean besides the desire to combine recall and precision in a single measure. An interpretable transformation named F^* has been proposed to mitigate this issue [58], see Equation (2.8). As the F^* metric is a monotonic transformation of the F1 score, an optimization with regard to one of them also maximizes the other.

$$F^* = \frac{F}{2 - F} = \frac{R \cdot P}{R + P - R \cdot P} \quad (2.8)$$

It should be mentioned that those pairwise measures may fall short when applied to multi-source linkages. For example, when evaluating a linkage result with four data sources, a missing record in a true entity cluster that should contain four records would count as three FN pairs. It could also lead to multiple FP pairs if the missed record was incorrectly assigned to another unrelated entity. However, as this work focuses on two-party linkage scenarios we do not dive into this topic further.

The quality measures described so far in this section depend on the choice of the classification threshold. For assessing the robustness of the classification outcome, the area under the curve (AUC) of recall and precision (AUC-PR) is beneficial. In the following, the shorter term AUC is used and refers to AUC-PR.

Figure 2.7 illustrates the binary evaluation measures for two scenarios where the similarity distributions of TM and TNM have a higher (left) or lower (right) overlap. The quality measures are computed for various thresholds (middle) and the optimal value with regard to the F1 score is determined, which is different for both scenarios. While the F1 score for the optimal threshold of each scenario is higher in the low-overlap setting, it is nearly equal when using the threshold of 0.53 which is optimal for the high-overlap scenario but not for the low-overlap setting. When using the AUC, it becomes evident that the classification results in the low overlap case are overall better independent of the chosen threshold.

2.2.2 BLOCKING EVALUATION

Blocking reduces the number of comparisons but may lead to early loss of matches. While the overall linkage quality measures also reflect these effects it is often beneficial to determine the effectiveness and quality of the blocking phase itself.

The Pairs Completeness (PC) measures the proportion of remaining true matches after the blocking step from all true matches. This is equivalent to the recall measure.

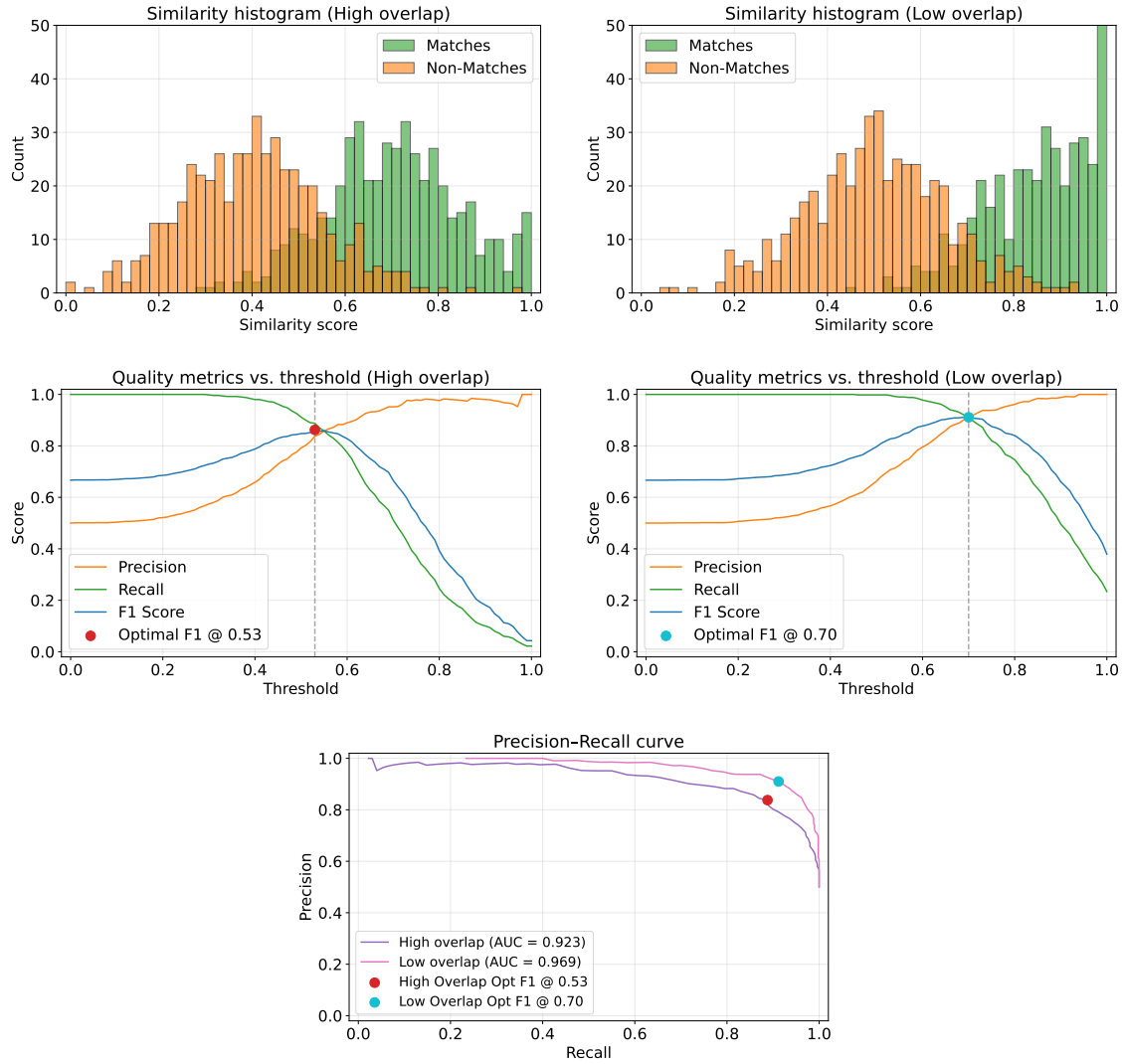


Figure 2.7: Illustration of how the extent of overlap between the similarity distributions of matches and non-matches (top) affects the resulting outcomes. The low overlap example on the right achieves better quality results than the high overlap example on the left. The precision-recall curve at the bottom enables comparisons of the overall quality for various thresholds.

Consequently, there is another measure analogous to the precision: Pairs Quality (PQ) is the ratio of true matches to the total number of match candidates.

The reduction ratio (RR) is defined as the proportion of comparisons that is avoided by the use of blocking. For record linkage between two files of equal size N , the total number of possible comparisons is $N \cdot N$.

$$RR = 1 - \frac{\text{\#candidate pairs after blocking}}{\text{\#all possible pairs}} \quad (2.9)$$

For example, a value $RR = 0.999 = 99.9\%$ refers to a reduction of the number of comparisons by a factor of 1,000.

2.3 PRIVACY-PRESERVING RECORD LINKAGE

In the last decades, a variety of methods for privacy-preserving record linkage has been proposed [20, 40, 50, 150]. In this section, we first describe the fundamental protocol designs and possible attack scenarios before explaining existing perturbation-based techniques, in particular Bloom filter encodings.

2.3.1 PROTOCOL DESIGNS

SEPARATION PRINCIPLE

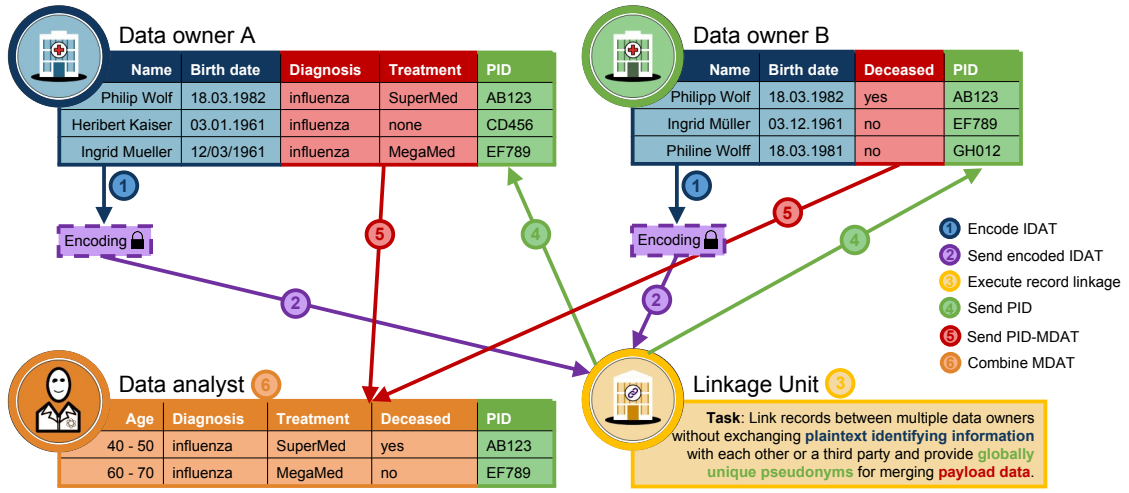


Figure 2.8: Separation principle: Identifying information (IDAT) is only used for linkage while the payload/microdata (MDAT) is only provided to the analysts. The linkage unit assigns globally unique pseudonyms (PID) that are used to combine corresponding MDAT. Perturbation-based PPRL adds an encoding step for the IDAT before sending them to the linkage unit.

At the data owners, we distinguish between two types of information: Identifying data (IDAT), needed for record linkage, such as name, date of birth and address, and medical/micro data (MDAT), needed for data analysis, such as type of disease, blood pressure, or medication. A principle commonly applied in medical research is the separation of MDAT and IDAT early in the data integration process, both, on the technical as well as on the organizational level [6, 49, 101]. The linkage unit only receives the IDAT values from the data holders but not the medical data to expose only minimal information for record linkage. The linkage unit determines matching records and returns unique pseudonyms, also called patient identifier codes (PID), which serve as global identifiers.

After the linkage process, the data holders can send the MDAT with the corresponding PID to the analysts which enables them to combine the information. To prevent unauthorized combination of such pseudonymized data from different research projects, use cases specific pseudonyms are used.

PERTURBATION-BASED METHODS

Even with this separation principle, the linkage unit can infer certain information from the plaintext records based on their context, i.e., the data source. For instance, a record from a mortality register implies that the corresponding individual is diseased. In presence of the plaintext IDAT, the linkage unit learns the identity of these individuals.

Some PPRL methods are based on perturbation techniques where the data owners encode the IDAT into a binary representation or bit vector (BV) before sending it to a semi-trusted third party for linkage [20]. The central component of these PPRL approaches is the transformation of plaintext data into an encoded form using keyed cryptographic one-way hash functions. One simple method is to map the (concatenated) plaintext data directly to a hash value, from which it is not possible to deduce the original data without knowing the hash function. As collisions of hash values are very unlikely, a matching hash value means that the plaintext data is also identical. The drawback of this straightforward method is that even small deviations, for example caused by typing errors, result in different hash values and therefore no match is recognized. Thus, encoding methods employ different techniques to achieve similarity-preserving, error-tolerant transformations, and thereby enable fuzzy comparison (see Section 2.3.3).

It is essential that the data owners keep the encoding parameters confidential and do not share it with the linkage unit. Otherwise, this reidentification attacks are possible where the original plaintext values are obtained, as outlined in Section 2.3.2.

Such three-party PPRL methods using a semi-trusted third party have been used in multiple real-world linkage projects for health research, such as [26, 105]. They are very efficient as they have low communication costs as the data owners typically transfer the encoded dataset only once to the linkage unit. The linkage unit can employ blocking techniques on the encoded data to reduce the number of match candidates.

Further approaches have been proposed that allow for two-party linkage without a third party, e.g., by using similarities to a reference dataset [149] or iterative disclosure of selected bits of the encodings [146].

SECURE MULTIPARTY COMPUTATION

Other PPRL protocols facilitate Secure Multi-Party Computation (SMPC) methods which provide formal security guarantees when evaluating a certain function, here for computing the record similarity, over private input data where the participating parties learn only about the result [16]. However, these approaches require significantly more communication and computational resources, which makes them practically unsuitable for linking large datasets. In principle, any function can be computed with SMPC, but many protocols impose certain restrictions in order to obtain reasonably efficient performance. Lazrig et al. [79] proposed a protocol without a third party using Yao protocol to evaluate

approximate similarity functions for Bloom filter encodings. In [78], a comparably fast linkage using the SMPC framework Sharemind [8] is presented which uses record-level hashes and thus has low error-tolerance. Extensions of the popular pseudonymization and record linkage tool Mainzliste using SMPC for linkage as well as set intersection cardinality computation have been proposed [76, 141].

Overall, these techniques are appealing, but their constraints hamper their use in practice. As illustrated in the preceding examples, SMPC linkage protocols frequently convert plaintext data into binary representations which are also used in perturbation-based PPRL to enable more efficient comparisons. Encoding techniques are therefore an important subject of study and can be integrated with SMPC to enhance security. Thus, this thesis focuses on perturbation approaches for PPRL.

2.3.2 ATTACK SCENARIOS

In this section, we outline potential attacks on perturbation-based PPRL protocols. For a more comprehensive overview, we refer to [150, 152, 154]. A common assumption is an honest-but-curious or semi-trusted adversary model which means that all participating institutions follow the protocol but may try to learn as much as possible about the sensitive data of others.

Dictionary attacks An attacker who a) can access an arbitrary number of bit vectors and b) knows the parameters used for the encoding, could encode any chosen combination of plaintext attributes, compare the result to the available bit vectors, and thereby reidentify individuals. This attack has a high damage potential, as it can also be applied to a single bit vector. Rare attributes can also be revealed in this way. The encoding parameters, in particular the secret key, must therefore only be known to the data owners. If a data owner collaborates with the linkage unit, they can jointly carry out a dictionary attack on the bit vectors of the remaining data sources, because the linkage unit is able to satisfy condition a) while the data owner fulfills condition b).

Brute force attacks An attacker has a) access to any number of bit vectors and b) the plaintext data for one or more bit vectors which is also referred to as a *known plaintext attack*. The plaintext data can then be encoded using every possible combination of parameters until the resulting output data matches the given bit vectors. These parameters can then be used to perform a dictionary attack on all bit vectors. This risk can be mitigated by choosing a sufficiently complex secret for the keyed hash functions.

Frequency attacks An attacker has a) access to a large number of bit vectors and b) a plaintext dataset with personal attributes (e.g., a telephone book) that has similar properties to the dataset on which the bit vectors are based. If the bit vectors and the plaintext data are sorted according to the frequency of attribute values, it is possible to identify correspondences that lead to the reidentification of certain individuals. This is

Encoded value	Frequency		Plaintext value	Frequency
AQIgcMYgIAAMKgoggyEkQA	812	↔	MUELLER	2076
SFQIATkATAEIEQJgCAQAAQ	694	↔	SCHMIDT	1432
U8cAIDIgJgEIOgJEDA0AAQ	521		SCHNEIDER	874
wEQgJKJAJAaIqIJECAQBAQ	503		FISCHER	734
...	...		...	...
			WALTER-BORJANS	1

Figure 2.9: Example of a reidentification attack by aligning encoded and plaintext values by their respective frequencies, thereby allowing an adversary to infer the original sensitive data without knowledge of the encoding function.

particularly relevant for individuals with common attribute values, like names, since only in their case it will be feasible to find reliable matches. In addition to the frequency of attributes, their length poses another vulnerability, given that some encoding schemes, such as Bloom filters, exhibit a correlation between the length of the input data and the cardinality of the resulting bit vector, that can be exploited [152]. These analyses can also be extended to the level of tokens, e.g., substrings, whereby rarer values could also be recovered if they consist of frequent tokens. Attribute values vary from region to region. Hence, it is challenging for an attacker to find a suitable plaintext dataset that is sufficiently similar to the origin of the bit vectors and large enough to compute a meaningful frequency distribution.

Graph-based attacks The results of the comparison step in a record linkage procedure can be represented as a similarity graph, where records serve as nodes and the similarity scores between them form the edges. If an attacker possesses a) such a similarity graph constructed from encoded records and b) a sufficiently similar reference dataset together with its own similarity graph, then plaintext records and encoded records can be matched by comparing their neighboring nodes and edges [151]. The underlying attack scenario assumes access to an identical or at least highly similar plaintext dataset, which restricts its applicability in practice. Further work has demonstrated that such attacks only succeed when there is a substantial overlap between plaintext and encoded records [61].

2.3.3 ENCODING METHODS

This section describes approaches for encoding plaintext in perturbation-based PPRL protocols. Given a record r with a set of attributes $\mathbf{A} = \{a_0, a_1, \dots, a_{i-1}\}$, the encoding function $\mathbf{E} = \text{encode}(r, \mathbf{P})$ is defined, where $\mathbf{P}$ denotes the set of encoding parameters and the output $\mathbf{E}$ is the encoded record consisting of one or more encoded parts. The encoding parameters $\mathbf{P}$ are known only to the data owners.

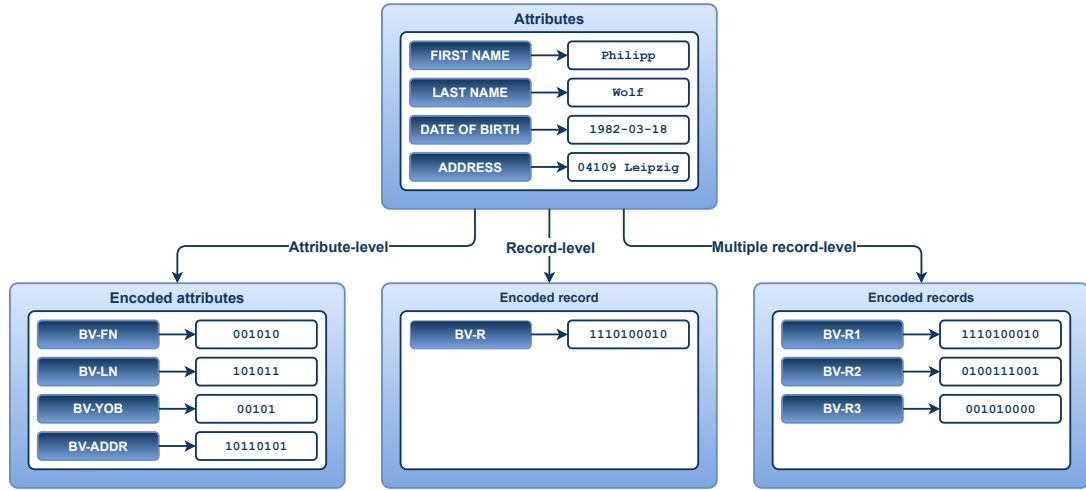


Figure 2.10: Comparison of different schemata for encoding an example record: Attribute-level encodings expose fine granular information and thus are generally more flexible in the linkage process. Record-level encodings conceal much of the details but suffer from potential quality issues that can be mitigated by using multiple of such encodings with varying composition.

ENCODING SCHEMATA

Most encoding functions can be applied either on individual attributes or on entire records. Figure 2.10 shows an example of these approaches.

Attribute-level encodings The encoding function produces $|\mathbf{E}| \geq i$ encoded parts, each corresponding to a specific attribute a_i . Attributes may have multiple encoded representations, e.g., for the value and its frequency. Each attribute encoding may use different parameters, which depend on the characteristics of its values, such as their average length. As a result, the encoded representations can differ, e.g., in their binary lengths. This schema enables the computation of attribute-level similarities and a dynamic weighting of attributes, as well as the detection and handling of missing values during matching.

Record-level encodings Record-level encodings result in $|\mathbf{E}| = 1$. Thus, all attributes are encoded in a joint bit vector. Different parameters (including weights) can be used for individual attributes. Frequency analysis becomes more challenging since the bits in the record-level encodings can no longer be readily assigned to their original individual attributes. Such record-level encodings impose limitations that can affect the linkage quality. The encoding parameters are chosen based on assumptions about dataset properties that might be inaccurate. In particular, attribute weights are typically determined by the attributes' value frequencies and error rates as described in Section 2.1.2. The latter are not known to the data owners prior to the linkage and must be estimated.

Multiple record-level encodings This method is similar to the previous, however, here distinct subsets of attributes are created, and each subset is represented by its own bit

vector. Consequently, the encoding function produces $|\mathbf{E}| > 1$ encoded parts, which do *not* correspond to individual attributes. For instance, one might construct combinations such as $\{\textit{First name}, \textit{last name}, \textit{date of birth}\}$ and $\{\textit{First name}, \textit{date of birth}, \textit{full address}\}$. If someone has moved, they could be matched using the first combination; if they have changed their last name due to marriage or divorce, they could be matched using the second. Different classification approaches are possible, for instance, using a majority decision or assuming a match when at least one encoding agrees.

When selecting an encoding method, it is necessary to balance the potential linkage quality and the associated privacy risk. Attribute-level encodings enable the linkage unit to adapt flexibly to the data at hand, like dealing with missing values. Exchanged attributes (such as first and last name) can be tolerated in both approaches as long as they are encoded in the same way, meaning no attribute-specific parameters are applied.

BLOOM FILTER ENCODING

Bloom filters were originally introduced as a data structure designed to efficiently determine whether an element belongs to a set [7]. Their use in the PPRL context was proposed by Schnell et al. [130]. They became the de-facto standard for practical PPRL on large datasets due to their straightforward implementation as well as their fast and error-tolerant comparison [50].

A Bloom filter is a bit vector of fixed size l where initially all bit positions are set to 0. The input data is split into tokens, typically (overlapping) substrings of length q (q-grams). Next, a set of h cryptographic hash functions $\mathbf{H} = \{H_0, H_1, \dots, H_{h-1}\}$ is applied to each token. The hash values obtained are assigned to bit positions in $[0, l - 1]$, e.g., using the modulo operator, and these positions are then set to '1'. Given that identical q-grams are mapped to the same bit positions, a high overlap of q-grams leads to similar Bloom filters making them suitable for fuzzy comparison. The hash functions should be independent, e.g., by using Random Hashing [92]. This transformation is not reversible due to collisions where multiple features are hashed to the same position. Figure 2.11 shows an example of two similar plaintext values that are tokenized and encoded in Bloom filters.

Bloom filter encodings were shown to be susceptible to certain types of attacks (see Section 2.3.2). Various encoding techniques and recommendations to hamper reidentification attacks have been proposed [46, 154]. In particular, attribute-level encodings should be avoided, as they make it relatively straightforward to match commonly occurring encoded attribute values to frequent plaintext values [21]. As a result, record-level variants of Bloom filter encodings have been introduced, most prominently the Cryptographic Long-term Key (CLK) scheme where tokens from all attributes are hashed into a joint Bloom filter [128].

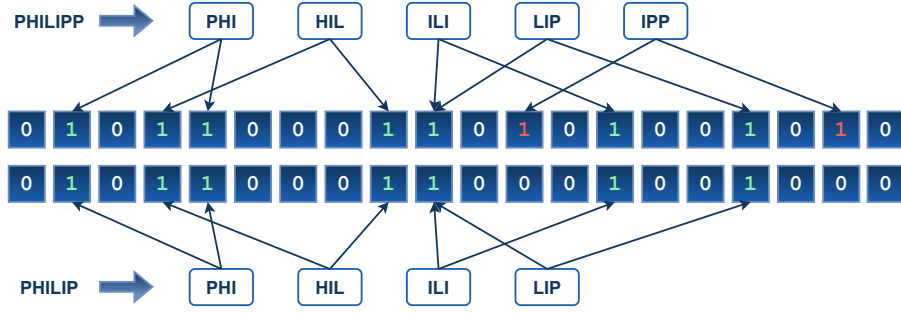


Figure 2.11: Example for encoding two similar values using Bloom filters with overlapping trigrams ($q = 3$), $l = 20$ and $h = 2$. Please note that in practical encodings more hash functions h are used, leading to more collisions; here, h is reduced for illustration.

WEIGHT APPLICATION

Since attribute weights play an important role in record linkage algorithms, it is essential to integrate them into such encodings in PPRL. Attribute weights can be applied in different ways in the PPRL process depending on the encoding strategy. If *attribute-level Bloom filter* (ABF) are used, the linkage unit can compare record pairs attribute-wise.

In the probabilistic record linkage theory of Fellegi and Sunter [36] (positive) agreement and (negative) disagreement weights are assigned to each attribute depending on whether they are equal or not (Equation (2.4)). The total weight is calculated by adding up the respective weights of all attribute pairs, but this approach does not support approximate similarity functions. Therefore, a common approach is the combination of normalized attribute similarity scores in the range $[0, 1]$ to an overall record similarity using a weighted average (Equation (2.1)). This is equivalent to the application of weights on plaintext data in conventional record linkage as they can make use of attribute-level comparisons. Weights can be determined and applied at the linkage unit during the matching phase.

When using record-level encodings, however, attribute weights must be incorporated in the encoding phase at the data owners. For Bloom filter encodings, the following approaches have been proposed and are illustrated with an example in Figure 2.12.

Record-level Bloom filter (RBF) encodings, proposed by Durham et al. [32], use a sampling based approach. Initially, separate attribute-level Bloom filters are generated. Based on the respective weights a proportional number of bits is sampled from each attribute-level Bloom filter to construct the joint record-level Bloom filter. Finally, the bits in the RBF are permuted to ensure that an attacker cannot easily reassign bits of the Bloom filters to specific attributes.

Following the *CLK-RBF* approach by Vatsalan et al. [147], weights can be reflected in the number of hash functions h_i used for each attribute i . The more hash functions are used for an attribute, the more bits in the final Bloom filter are set based on that attribute. Consequently, the influence of that attribute on the Bloom filter similarity is stronger.

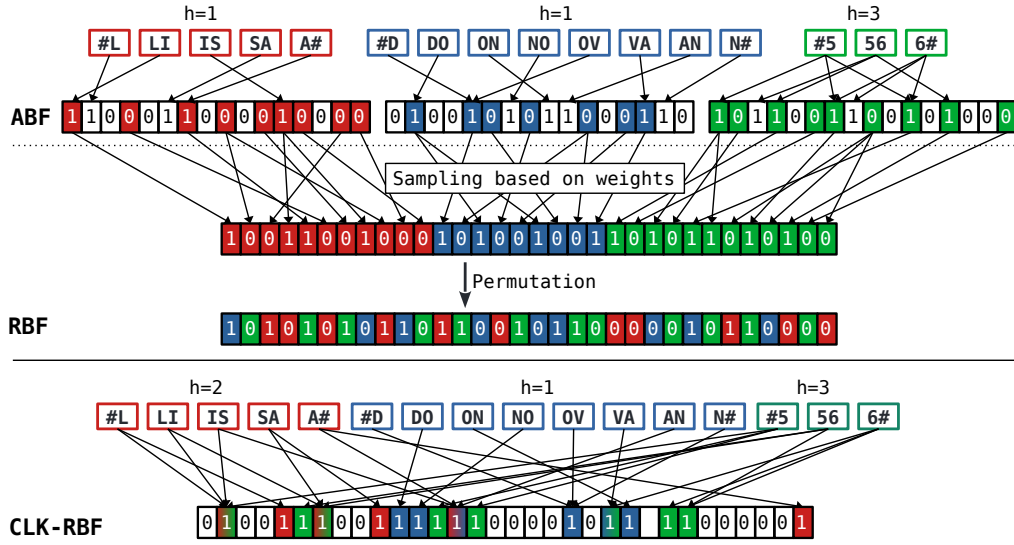


Figure 2.12: Encoding of an example record using two record-level techniques which incorporate attribute weights: RBF [32] sample bits from attribute-level encodings (ABF) whereas CLK-RBF [147] adapt the number of hash functions based on relative attribute weights.

BLOOM FILTER VARIATIONS AND HARDENING TECHNIQUES

To prevent or hamper frequency analyses, a variety of hardening techniques has been proposed in the literature. These aim to avoid vulnerability patterns in the bit vectors [152], but are partially ineffective or may have a strong negative impact on the linkage quality [46]. The usage of salting for individual attributes or groups of attributes can improve privacy as well as the linkage quality. This attribute salting uses different seeds for constructing hash functions of different attributes, which obfuscates q-gram frequencies in the encodings. Record-level salting, on the other hand relies on the presence of a stable attribute to derive a seed from the specific record being encoded. Without such an attribute, the resulting Bloom filter encodings become highly dissimilar, which in turn causes low recall.

XOR-folding is a straightforward yet effective method for balancing utility and privacy: the Bloom filter is divided into two equal parts, that are then merged by applying a bitwise XOR operation to corresponding bits [133]. Other hardenings aim to achieve a balanced cardinality of the resulting Bloom filters to impede attacks exploiting attribute lengths [132] or integrate random bit flipping within the differential privacy framework [2, 34]. A different approach based on autoencoders was proposed by [24]. Autoencoders are a type of neural networks designed for transforming input data to an (compressed) encoded representation which carries the essential information.

Apart from their susceptibility to reidentification attacks, Bloom filter encodings also come with some limitations that potentially affect the linkage quality. Typically, they do not incorporate any positional information of the tokens [20]. Thus, q-grams from

different parts of an attribute or in record-level encodings potentially even from different attributes are encoded likewise, thereby losing this information for the comparison. This stands in contrast to plaintext string similarity measures such as Jaro-Winkler [161] which are explicitly designed to assign greater importance to agreeing characters that occur at the beginning of strings as it has been shown to be favorable for comparing names [17].

OTHER ENCODINGS

Token-based encodings such as Bloom filters are not well suited for numerical data as they would fail, for example, to detect the close distance of '99' and '100'. Specialized encoding techniques have been proposed for numerical data [28, 69] and hierarchical classification codes [131]. Unfortunately, these methods are not suitable for mixed data types for application in record-level encoding schemata.

Due to the attack possibilities on Bloom filters, further encoding techniques have been proposed to mitigate this issue, such as tabulation min-hashes [139], two-step hash encoding [113] and reference set encodings [164]. They follow the same basic principle of sending the encoded data to the linkage unit for comparison which requires a similarity preserving transformation. Thereby, they are still naturally prone to some kind of frequency analysis.

A very different technique are fingerprint or signature based linkage approaches [49]. They adopt the *minimum linkage information* approach, aiming to create a record representation that includes only the information necessary to uniquely identify it, while still tolerating slight differences. For instance, the Swiss Anonymous Linkage Code [10] concatenates the Soundex codes of first and last names with the full date of birth and gender. Essentially, they are variants of rule-based linkage approaches. Since they combine plaintext information bits that are often commonly known, such as data about well-known individuals, they do not provide sufficient protection against reidentification. Self-generated identifying codes aim to include information which is not publicly known such as an persons favorite sports team during high school [85, 129]. Since this information is not available in datasets for general purpose linkage, these methods are restricted to longitudinal research studies.

To address the possible linkage quality issues associated with using a single encoded representation, some applications employ multiple record-level encodings. The US American clinical research network PCORnet employs a hashing-based PPRL system [71, 88]. It generates multiple hashed tokens from combinations of personal identifying information (PII) (or IDAT) and uses a majority approach of matching tokens to classify record pairs as a match. Randall et al. [117] proposed a protocol using multiple dynamic match keys that is essentially an extension of the anonymous linkage code concept. Possible attribute

combinations are determined based on the characteristics of the datasets to be linked using the probabilistic model (see Section 2.1.2). Following a privacy attack [153], a recommendation for enhancing the privacy of this method was developed: removing non-unique match keys at each data owner. The study showed that this countermeasure led to only a minor reduction in recall. A similar cryptanalysis approach on the linkage protocol employed in German cancer registries showed similar flaws and recommendations [62].

2.3.4 LINKAGE COMPONENTS

The procedure within the linkage unit in PPRL protocols generally resembles matching based on plaintext as described in Section 2.1. However, due to the encoded representation of the records, certain restrictions apply for the available methods.

PREPROCESSING

The preprocessing must be carried out by the data owners themselves, since it operates on plaintext data. Schema matching is usually not required in PPRL, because its privacy-preserving setup generally limits the use to linkage based on personal quasi-identifiers, which tend to have similar structures or can at least be transformed into a comparable structure with relatively little effort. Differences in the granularity of attributes may require preprocessing, e.g., to split a single attribute with the full name into multiple attributes for first, middle and last name. Recently, privacy-preserving schema matching methods using functional dependencies [60] and k-anonymized data [4] have been proposed. In this work, a schema matching step is not included, because the preprocessing is conducted on the plaintext at each data source to align with a common target schema for the linkage input.

BLOCKING AND FILTERING

Both, blocking and filtering techniques are employed in PPRL similarly to plaintext linkage. Nevertheless, PPRL specific research has been conducted on blocking and filtering methods for the multi-party setting with $S > 2$ [111, 136].

Locality-sensitive hashing (LSH) was proposed for solving the nearest neighborhood problem in high-dimensional data spaces [48]. The basic idea of LSH is to apply a set of hash functions on the bit vectors. These hash functions are designed to be sensitive to a particular distance measure. The probability of a collision is much higher for objects with a small distance (high similarity) than for objects with greater distance (low similarity). The hash functions employed in LSH are separate and unrelated to the ones used in Bloom filter construction.

LSH in the PPRL context is used to derive blocking keys from bit vectors, including Bloom filter [44, 70]. For this, one can employ hash functions that are sensitive to Hamming distance (HLSH). The functions f_i yield the bit located at position i in the bit vector [33]. For example, when the function f_4 is applied to the bit vector 11011001, it returns the bit value on position 4 and therefore 1. To cluster similar records, a blocking key is formed by applying Ψ hash functions that are chosen at random. The resulting outputs of these Ψ hash functions are then concatenated to generate the blocking key. Thus, the parameter Ψ represents the length of the blocking key.

To enhance error tolerance, Λ blocking keys are generated to raise the likelihood that two similar bit vectors coincide in at least one blocking key, ensuring that the encoded records are then compared with each other to determine whether they represent a match.

A higher value of Ψ increases the chance that only highly similar bit vectors end up in the same block, what consequently results in smaller blocks and fewer comparisons. A smaller value of Ψ yields larger blocks, but it also reduces the chance that two similar bit vectors are overlooked because of erroneous data. Higher values of Λ increase the likelihood that similar bit vectors are assigned the same blocking key. At the same time, however, this also raises the number of blocks and, consequently, the number of candidates to be processed, which leads to longer execution times.

Metric space filtering is another approach which leverages the mathematical properties of metric spaces to select potential record pairs before applying more computationally intensive comparison methods. The filtering mechanism works by excluding match candidates that cannot reach a predefined similarity threshold due to the triangle inequality property of metric spaces, thereby eliminating unnecessary comparisons without sacrificing accuracy [51, 135].

COMPARISON

For token-based encoding techniques such as Bloom filters, set similarity measures are used, usually either Jaccard or Dice [29]. Given their close relationship, both are suitable and applied in literature [20]. Dice similarity assigns relatively more weight to shared elements by normalizing the intersection by the sum of set sizes rather than their union, what generally results in higher similarity values than Jaccard for the same degree of overlap. Consequently, when selecting a classification threshold, the optimal threshold is typically higher for Dice than for Jaccard similarity. The similarity measures are defined as

$$\text{dice}(a, b) = \frac{2|a \wedge b|}{|a| + |b|} \quad (2.10)$$

$$\text{jaccard}(a, b) = \frac{|a \wedge b|}{|a \vee b|} = \frac{|a \wedge b|}{|a| + |b| - |a \wedge b|} \quad (2.11)$$

where $|x|$ is the cardinality of a Bloom filter x , i.e., the number of bits set to 1. $\wedge$ denotes the bitwise AND operation.

In contrast to plaintext linkage, the same token-based similarity function is typically used for all encoded attributes or even the record-level encodings whereas in plaintext linkage distinct functions can be chosen for attributes representing numerical data or strings, as outlined in Figure 2.4.

CLASSIFICATION

The classification models used in PPRL are, in principle, the same as the ones applied in traditional record linkage (see Section 2.1.2), but they are usually simpler because only a limited number of features are available. For record-level encodings typically only a single record-level similarity score is used, leading to the selection of a threshold-based classifier. Multiple similarity measures could be applied on the same record-level encodings to derive additional features for classification. Nonetheless, as they are applied on the same masked data and similarity measures such as the Dice or Jaccard coefficients are transformations of each other, the benefit is limited. Deep learning classifiers were applied in PPRL where the data owners utilized federated learning to collaboratively train such a classification model on multiple record-level similarities [114]. However, the method depends on the existence of labeled training data which is commonly not available in the PPRL context.

2.3.5 PRIVACY ASSESSMENT

There is no universal privacy measure for perturbation-based PPRL approaches or more specifically, for associated encoding techniques. It is mostly about showing that previously published attack scenarios are not (practically) feasible on certain encoding techniques.

In [152], a vulnerability assessment framework is proposed, consisting of the following attack vectors: frequency, length, co-occurrence, similarity, and similarity neighborhood.

Multiple privacy measures for Bloom filters have been proposed based on the uniformity of the bit vectors [46]. The general notion is that encodings are less vulnerable to attacks when the encoded data follows a uniform distribution, i.e., does not contain exploitable frequency patterns.

A straightforward approach uses the well-known Gini coefficient [15] which is primarily known in economics for assessing the unequal distribution of wealth or income among the population. Nevertheless, it can also be applied to other variables, in this case frequencies of bit position being set to 1. The counts are normalized to the total sum of the 1-bits and sorted according to their size. They are then summed up, resulting in a cumulative distribution.

$$G(x) = \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{m-1} |c_i - c_j|}{2m \cdot b} \quad (2.12)$$

The Gini coefficient varies between 0, indicating perfect equality where all values are identical, and 1, representing maximum inequality where a single position contains all the 1-bits while the others consist solely of 0-bits.

2.4 APPLICATION CONTEXT

(Privacy-preserving) record linkage methods are often applied as a part of rather complex data integration processes. In order to facilitate such projects, various countries such as Australia [11, 115], USA [88, 157] or the United Kingdom [86] established population data centers, in particular for health data. In Germany, the Medical Informatics Initiative [137] – among other goals – aims to develop a unified infrastructure for secondary health research. Linkage is conducted on-demand per data use project (DUP) and thus on different patient groups with varying data characteristics, depending on the requirements of the analysis. The following application scenarios utilizing cross-site record linkage were identified by the data protection working group of this initiative [55, 89].

Data provision to analysts Data from multiple data owners is combined based on the record linkage result. It is provided as a one-time dataset or as a initial dataset with subsequent updates when additional subjects turn up or additional data for previous subjects is collected. This use case is straightforward and illustrated above in Figure 2.8. The linkage is typically not tightly coupled to the analysis but a distinct previous step in the workflow. Even when quasi-identifiers, including names, addresses, or phone numbers, are removed from patient records, additional methods such as generalization or k-anonymity [143] are necessary to de-identify the MDAT. This is because distinctive or infrequent details in the records, like uncommon medical conditions or rare clinical incidents, can still enable reidentification when they are linked with publicly accessible information or other datasets [102, 106].

Distributed analytics As anonymized or pseudonymized data still poses a reidentification risk, distributed analytic methods aim to *bring the algorithm to the data* and not vice versa [156, 158]. Rather than (centrally) aggregating the data, intermediate analysis results are securely exchanged to update the final outcome iteratively. Bloom filter based PPRL has been integrated in such a setting to enable the detection of patients that are registered in multiple institutions [67]. After determining global record identifiers, the data owners can include their unique records in the analysis to prevent duplicate bias.

Feasibility query Feasibility queries allow researchers to get an estimate of the number of patient records they can expect for certain selection criteria. These queries are applied before the actual data use agreement is finalized and intended to enable

researchers to formulate their request in such a way that the data provided will meet their requirements. In most cases, matching can be omitted and the hit counts reported by the individual data owners can be summed up, given that no exact values are returned anyway. If the intended analysis requires the consolidation of data variables from different sources, linkage is essential in order to assess the expected result. Set overlap estimation approaches allow to efficiently approximate the result size [142]. In general, applying secure multi-party computation is substantially more resource-intensive than perturbation-based PPRL and may thereby be infeasible. Feasibility queries are concerned with the *cardinality* of the intersection rather than identifying specific record pairs. SMPC techniques for calculating private set intersection (PSI) cardinality should be considered due to their high privacy guarantees [90].

Pseudonym management In addition to generating knowledge that supports healthcare provision as a whole, analysis of pseudonymous patient health data can also uncover findings that are beneficial for individual patients. For instance, the data could indicate an increased health risk for a patient that could be alleviated by prevention measures. Furthermore, data analysts may want to approach patients or their health providers to request additional information. Such processes require the resolution of pseudonyms to their original patient identifiers which is a task for data trustees. Independent of the chosen linkage method, a data trustee institution is thus often needed to enable such workflows [5, 55].

Incremental linkage in patient registries The previous use cases primarily involve batch linkage where static datasets from multiple data owners are matched once. This is also known as *offline matching* [17]. Once the linkage outcome has been established and the global identifiers (or pseudonyms) have been assigned, the encoded data is no longer required. However, some use cases call for an incremental approach, where records are progressively added and linked to a long-term patient database. Cancer registries are one such example. Some patient registry systems, including Mainzelliste [77, 123] and E-PIX [54, 56], provide support for Bloom filter based PPRL.

EMBEDDING OF PPRL IN THE DATA LINKAGE PROCESS

The effort required to link records across institutions, together with the increased privacy risk, is only justified for certain data use projects. Accordingly, the choice on whether to perform matching is individual for each DUP. This decision should be based on the consideration of various aspects.

a) Estimation of the expected number of duplicates The overlap depends on the selection criteria of the research project and thus on the chosen subsets of patients at the data owner sites. It can be estimated qualitatively from expert predictions or quantitatively by applying set overlap assessment methods [142].

b) Description of the linkage variables The achievable quality in record linkage depends on the characteristics of the datasets and on the ability to adapt linkage configuration to this dataset. Each data owner should therefore describe the master data available for the locally selected patients, particularly the proportion of missing values. In accordance with recommendation 6.1 of “Good Practice Data Linkage” [87], each data linkage must include a description of the quality of the key variables in the final report.

c) Impact on the research project Depending on the project scope, duplicates in the combined dataset can have different effects on the analysis and thereby lead to different requirements with regard to the linkage quality. While a duplicate detection may be *nice-to-have* for some projects, it will be essential for others, for example in long-term studies on rare diseases where the focus is exactly on those patients that were treated in multiple sites [76]. High recall has priority, e.g., when all individuals of a certain group need to be identified in order to inform them about a possible life-saving treatment where false positives do not have bad consequences or can be resolved easily after contact. Nevertheless, precision is more often the priority, e.g., when it is important to avoid merging data items of different persons which may lead to wrong conclusions during analysis.

Overall, it should be noted that linkage errors only affect the analysis results when the information obtained from mistakenly linked or unlinked records diverges from the conclusions that would have been obtained if the linkage quality were perfectly correct [30]. Because linkage errors can strongly affect downstream analyses, it is essential to propagate uncertainties along the linkage result [82].

These three aspects could be studied in the context of feasibility queries. The data owner would then not only determine how many patients satisfy the selection criteria, but also provide a description of the characteristics of variables relevant to record linkage. The analysts receive an estimated number of duplicates and have to assess the impact on their project. On this basis, the coordinating institution decides whether cross-site record linkage should be carried out.

Post-processing (Fully automated) matching algorithms are not able to achieve perfect linkage quality. Even with plaintext data, there are difficult cases that require manual intervention, like requesting additional information. In conventional record linkage, the match results can be checked on a sample basis. In PPRL, however, this is typically not possible due to the encoded representation.

According to recommendation 6.4 of the “Good Practice Data Linkage” [87], after each data linkage the number of linked and non-linked records should be checked on the basis of the original files. This includes the plausibility of the determined number of matches and potential differences between matching and non-matching records with regard to variables used in the analysis.

3

Robustness evaluation framework

In this chapter, we present the framework and implementation for studying PPRL methods which is used throughout this thesis. Section 3.1 motivates the design choice of the framework. Section 3.2, which is based on [120, 121], describes the architecture and implementation of the service-oriented approach for the PPRL process itself as well as further components which are necessary for evaluating linkage algorithms more broadly: Dataset generators create a variety of test datasets for the linkage process. The analysis components help to understand strength and limitations of linkage approaches. In Section 3.3, the framework is employed to study the robustness of PPRL with record-level encodings, focusing on attribute weights and comparison to plaintext linkage. Finally in Section 3.4 we discuss both, the framework design and the evaluation results.

3.1 MOTIVATION

Attribute-level encodings create exploitable frequency patterns in encoded data, facilitating rather straightforward reidentification, e.g., by aligning the most common plaintext values with the most common Bloom filters [154]. Consequently, these encodings are appropriate for data linkages that require a flexible matching strategy to achieve high-quality outcomes and privacy risks are limited. Usually, attributes are combined into a single record-level encoding for improved attack resilience. This approach strongly limits the flexibility of the matching algorithm. The linkage unit only uses a simple threshold-based classification based on a single similarity score. Despite this simplistic approach, the results of PPRL are often surprisingly good because the linkage problems are usually less complex in comparison to other domains, e.g., when linking commercial products from data sources with differing schemas.

A key challenge in this process is the determination of suitable encoding and linkage parameters without having information regarding the properties of matching and non-matching records. In conventional record linkage attribute weights are often used during the comparison and classification step for aggregating the attribute similarity scores for subsequent threshold-based classification. In PPRL protocols with record-level encodings, however, the weights must be applied at the time of encoding. Thereby, the computation of suitable weights, e.g., based on the probabilistic approach by Fellegi and Sunter [36], must be done without access to attribute similarities of potential matches and non-matches. Therefore, PPRL protocols using such record-level perturbation methods are intrinsically less flexible and prone to malconfiguration.

Data custodians seeking an appropriate linkage strategy for their particular use case may find themselves without sufficient guidance to select a suitable solution. Consequently, they may either revert to traditional record linkage methods, which necessitate significant organizational efforts for engagement with fully reliable third-party entities, or they may abandon their objectives due to these considerable challenges.

Usually, entity resolution algorithms are evaluated based on benchmark datasets from various domains. Considering their particular application involving sensitive information, the assessment of PPRL algorithms on datasets derived for instance from commercial domains such as Abt-Buy or bibliographic sources like DBLP and Google Scholar is not advisable [72]. PPRL algorithms are typically applied on linkage problems where personal identifying attributes are used for comparison. Naturally, such datasets from real-world databases are rarely publicly available. Research in this domain therefore either uses those few datasets which are publicly available or derived from public sources (in particular temporal snapshots of voter registries), generate synthetic datasets or use datasets which the authors have access to based on their affiliation or special agreements [20]. Although publications frequently offer synthetic or publicly accessible datasets, this is generally not the case for real-world linkage problems. The absence of standardized benchmark datasets makes it challenging to compare the outcomes of various PPRL algorithms. Our approach facilitates the versatile generation of linkage datasets and the derivation of variants, thereby enabling a more comprehensive study of PPRL algorithms.

As outlined in Section 2.4, the PPRL process is typically integrated in a more complex workflow and architecture including pseudonym management and de-identification of the data of interest (MDAT). A variety of heterogeneous software components from academia and industry are involved at the data owners, e.g., hospital information systems, at the data trustee institution as well as for the overall workflow management. It is essential that the PPRL components can be easily integrated into such an ecosystem. To this end, the proposed framework is built on a service-oriented architecture using REST APIs, which makes this possible without encountering compatibility problems that typically arise when integrating software libraries tied to specific programming languages. In particular,

it simplifies the adoption of the FHIR (Fast Healthcare Interoperability Resources) standard for exchanging patient records [37, 127].

Moreover, linkage quality measures are often provided for the entire dataset, which might include a range of data error types. Thus, observers cannot assess whether they can expect a similar quality for their dataset with potentially different properties regarding the available attributes, missingness, etc. The evaluation should include a more detailed report on the performance of a linkage method so that strengths and weaknesses with regard to certain error types or other dataset properties can be assessed.

In this chapter, we present a framework for comprehensive evaluation and quality analysis of PPRL algorithms in order to support data custodians in choosing suitable methods and to facilitate further methodological research in this domain. In particular, we make the following contributions:

- We introduce our framework for evaluating PPRL algorithms by employing a combination of methods for generating, corrupting, and deriving benchmark datasets.
- We present our methodology for analyzing linkage outcomes using descriptive tags, facilitating a comprehensive comparison and refinement of methods.
- We employ the framework to generate and analyze a variety of linkage problems that are used to evaluate perturbation-based PPRL using record-level encodings in comparison to plaintext linkage.

RELATED WORK

There exist numerous general-purpose record linkage frameworks, such as pyJedAI [93] and others [27, 83, 124], as well as dedicated PPRL tools [45, 134] that offer implementations of both record linkage and perturbation encoding methods. PRIMAT [45] in particular offers implementations for a variety of encoding and hardening techniques as well as postprocessing algorithms for [43]. However, the existing tools are either programming libraries or provide only command line interfaces and thus, are not designed as stand-alone services. Further solutions are specialized for medical applications, such as Mainzelliste [77, 123, 144] and E-PIX [54, 56]. However, these tools are primarily confined to executing linkage operations and lack the capabilities necessary for generating test datasets or conducting in-depth result analysis beyond basic quality metrics.

For evaluating record linkage algorithms, benchmark datasets are required, ideally covering various possible scenarios regarding data origin and quality issues to prove wide applicability of these methods.

On the one hand, a number of data generators for entity resolution tasks exist. Tools like FEBRL [18] create records entirely from scratch. The pseudopeople package facilitates the generation of extensive datasets through individual-based modeling of the US population,

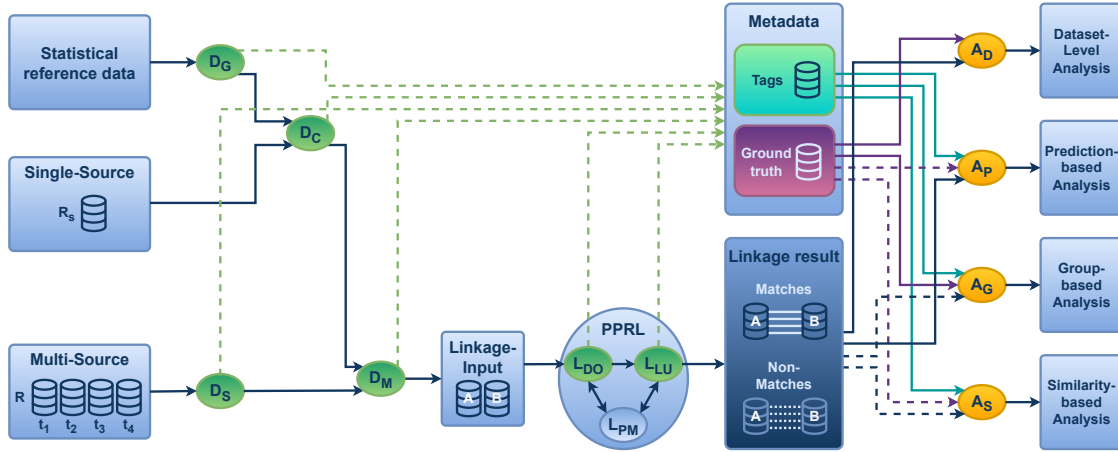


Figure 3.1: Architecture of the evaluation framework with linkage input creation components on the left side (D_G : Synthetic data generator, D_C : Data corrupter, D_S : Selector from multi-source record clusters, D_M : Linkage problem modifier), the PPRL execution services in the center (L_{DO} : Encoding, L_{LU} : Matching, L_{PM} : Protocol management) and the result analysis modules on the right side (A_D : Dataset-level evaluation, A_P : Prediction analysis, A_G : Group analysis, A_S : Similarity analysis). Green components produce descriptive tags while yellow modules consume the metadata.

which also encompasses household structures [53]. Others create a corrupted variant with a variety of error types based on an existing (real-world) input dataset [64]. Another group of data creation tools, such as GeCo [22] and Gecko [66], integrate both the generative and corruptive aspects. Approaches such as [108] facilitate transformer models to synthesize surrogate datasets from existing linkage problems which follow a similar distribution of matches and non-matches.

On the other hand, further tools exist to benchmark matching solutions based on profiling test datasets such as [103]. One of the few tools available for understanding record linkage results is Frost/Snowman [52], which has been proposed for comparing linkage algorithms on certain benchmark datasets or a certain algorithm on multiple datasets using various metrics. Nevertheless, the platform lacks features for generating test data or conducting in-depth analysis of results when dealing with multiple dataset representations (plaintext and encoded).

To the best of our knowledge, there is no solution yet which comprises data generation and detailed linkage result exploration in a holistic framework, in particular with a focus on PPRL.

3.2 SERVICE-ORIENTED FRAMEWORK

The overall architecture of our framework is depicted in Figure 3.1. This thesis focuses on two linkage input sources but the framework is generalizable for multi-party linkage.

On the left side, multiple paths for creating linkage datasets are provided, depending

on the available input data. By utilizing statistical population data, it's possible to create synthetic datasets (D_G) and corrupt them (D_C) while maintaining real-world characteristics. Linkage problems can also be generated based on the corruption of given single-source data or through the selection (D_S) from multi-source record clusters. Ultimately, the dataset can be altered to adjust size and overlap of the data sources (D_M).

In the center, privacy-preserving record linkage components are responsible for the encoding (L_{DO}) and matching (L_{LU}) phases of a perturbation-based protocol. The data flow is orchestrated by a protocol manager service (L_{PM}).

All those components serve as potential sources of descriptive tags which are gathered to facilitate subsequent analysis by the components on the right side. In addition, ground truth information for the linkage input is needed for most analysis types and thus is provided by the data creation components. Basic dataset-level analysis (A_D) of the linkage can be carried out using only the ground truth of a particular test dataset. The remaining components (A_P , A_G , and A_S) depend on tags to facilitate a more in-depth examination.

Our comprehensive framework allows to trace natural or intentionally induced data characteristics through the PPRL algorithm for a fine grained analysis of the linkage outcome even when obscured by additional noise in this process. Notably, post-linkage tag inference could not comprise pairs that are eliminated during the blocking phase. This is facilitated by using a common data model throughout the framework to describe such traits, which we refer to as *tags*, as described in the next section.

3.2.1 TAG CONCEPT

Tags describe linkage problems based on dataset and outcome characteristics across the dimensions of **Granularity**, **Representation**, **Origin**, and **Type**. We denote tags as $\tau_G^R[O](\text{LABEL}, [\text{string}], [\text{number}])$ whereby the origin O is included only if relevant.

Granularity Tags describe individual records (τ_r), attributes (τ_a), record pairs (τ_{rp}), attribute pairs (τ_{ap}), or entire datasets (τ_R). While all tags include references to the respective dataset, the availability of references to the respective record(s) and attributes depend on the granularity level.

Dataset representation Tags may originate from the plaintext data (P) or encoded data (E), or describe structural traits (S).

Origin While some pair-level tags such as the similarity emerge during matching, plaintext records and their attributes are analyzed in advance. Other (pair-level) tags may be derived beforehand, during generation, corruption or selection.

Type Tags have a label and may include numerical/string values for detailed description. The content of the values depends on the tag type. Some tags may only have categorical (string) values. For tags with numerical values, the string is used for assignments to predefined intervals or categories for simplified analysis.

	Scope of provided tags	Examples
D_G	Record-level	$\tau_r^S(\text{HOUSEHOLD}, \text{'single'})$
D_C	Pair corruption method (intention and technique)	$\tau_{rp}^S(\text{MODIFIER}, \text{'rareNameMove'})$ $\tau_{ap}^P(\text{TYPO}, \text{'f->d'})$
D_S	Record and pair-level	$\tau_{rp}^S(\text{YEAR_DIFF}, \text{'0-3'}, 2)$ $\tau_r^S(\text{VOTER_GROUP}, \text{'Democrat'})$
D_M	Dataset-level	$\tau_R(\text{OVERLAP}, \text{'A-B'}, 0.2)$
L_{DO}	Plaintext (Indiv. and pair) (Dynamic) encoding param.	$\tau_a^P(\text{MISSING}), \tau_{ap}^P(\text{SUBSET})$ $\tau_a^E(\text{NUM_HASHES}, \text{'default'}, 20)$
L_{LU}	Encoded data description Intermediate results	$\tau_r^E(\text{FILLRATIO}, \text{'aboveAvg'}, 0.42)$ $\tau_{rp}^E(\text{SIM}, \text{'0.85-0.90'}, 0.865)$ $\tau_r^E(\text{BLOCK_SIZE}, \text{'belowAvg'}, 12)$

Table 3.1: Examples of tags which are provided by data generation and linkage execution components of the framework.

Table 3.1 lists the tag scope and examples from the components in our framework. The implementation stores all tags in a document-oriented database using a unified flat data model. However, tags can also be provided on-demand for ad-hoc analysis without database persistence.

3.2.2 LINKAGE PROBLEM CREATION

In the following, we describe the components provided by our framework for creating the input datasets for the PPRL algorithm.

D_G : SYNTHETIC DATASET GENERATION

Synthesizing realistic data is challenging, as evidenced by ongoing development of new tools and methodologies as outlined above. Such datasets can be generated based on statistical distributions of name frequencies, age cohorts, and residential addresses etc. These values may come from public sources like national censuses or from similar datasets related to the specific use case. Our implementation comprises a data generator tailored for Germany, particularly aimed at tackling the issue of aligning individuals residing within household structures in record linkage [31]. These pairs often exhibit a relatively high similarity because of shared surnames and/or locations. The generator creates datasets in two main stages: individual data generation and household structure modeling.

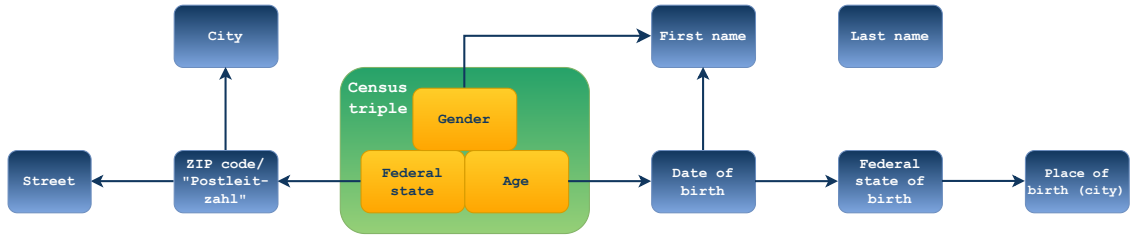


Figure 3.2: Attribute dependency graph of the data generator based on German census data: Records are constructed based on census data triples (federal state, gender and age). [Adapted from [31]]

Individuals generation The generator utilizes German Census data¹ to accurately reflect real-world demographics, including age, gender, and regional statistics. Dependencies between attributes are modeled with a directed acyclic graph to ensure logical consistency between attributes as illustrated in Figure 3.2. For example, gender and date of birth influence the choice of first name. The graph is simplified, excluding certain dependencies. For instance, there’s likely a correlation between current residence and birthplace. Yet, to incorporate these dependencies into the generator, statistical data is required which is not easily accessible. In future work, such dependencies can be integrated as long as no cyclic dependencies in the graph emerge. Values of name, date of birth, and location-related attributes are generated from probability distributions.

Household modeling Household structures are generated in four phases, each focusing on a specific household type. *Family households* are created with realistic parent-child relationships. The number of children and the type of partnership (e.g., married, single parent) are determined using statistical distributions. Dependencies such as age gaps between siblings and parents are modeled. *Childless couples* and *single-person households* are generated based on age (and partnership type) distributions. *Shared housings* such as student apartments, are based on grouping individuals with similar demographic attributes, in particular age.

Apart from the actual records, the generator also provides tags describing the household type of the generated records. This enables post-linkage analysis of household-related linkage bias.

D_C : LINKAGE PROBLEM CREATION BY RECORD CORRUPTION

Based on a foundational dataset composed of either synthetic or real records, additional records of other sources are subsequently generated by inducing typical data quality issues such as typos as well as missing or replaced attribute values. The goal of our corrupter is not primarily to simulate authentic compositions of data errors, but rather to enable the creation of diverse record pair categories that can be classified as true *matches* or *non-matches*. An additional objective is to assign tags according to the modification applied

¹<https://ergebnisse.zensus2022.de/datenbank/online/>

to each pair, enabling the analysis of the linkage output for the expected classification result.

The framework offers various types of corruption that can be combined. At the attribute level, basic manipulations of strings and dates can be applied, such as character swaps or substitutions to simulate typographical errors, and allows for variations in dates, like date of birth adjustments. At the record level, attribute values can be swapped (such as first and last names), substituted, expanded depending on a specified value distribution (for instance, a name change after marriage or an address change following a relocation) or removed to simulate missing values. All modifications are conditional, making it possible to represent data quality issues specific to certain groups, such as young adults being more prone to address changes compared to middle-aged individuals. Every alteration generates a tag, enabling thorough tracking by the post-linkage analyzers. Moreover, these modifications are grouped to simplify the assignment of more general tags $\tau_{rp}^S[D_C](MODIFIER)$. These tags capture the intended type of pairing, such as an individual who relocated which affects multiple residential attributes, or two similar but distinct records with common names, that a human oracle would likely classify as a *match* or *non-match* respectively.

D_S : SELECTION FROM EXISTING RECORD CLUSTERS

The previous methods for generating data face the challenge of choosing suitable parameters to accurately mimic real-world errors. The usefulness of these test datasets largely depends on whether the initial assumptions regarding the dataset’s characteristics are appropriate, including data quality issues. An alternative method employed in the literature involves utilizing historical data to simulate linkage issues across diverse temporal snapshots. This method naturally incorporates real-world characteristics of outdated data. Databases of registered voters, such as the one in North Carolina (NCVR), are frequently utilized as linkage benchmark datasets [20]. Based on the NCVR data, Panse et al. [98] provide a database for large-scale record linkage tasks, containing clusters of historical records sharing the same voter ID. We employ a dataset generator enabling the selection of subsets considering the degree of duplicate *dirty*ness or the snapshot time (t) span within these record clusters. Table 3.2 shows an example of such a cluster from NCVR where different record pairs could be selected for inclusion in the linkage dataset, depending on the desired composition. For example, a higher minimal time span between selected records typically results in more challenging match candidates. A potential issue when creating a linkage problem with ground truth labels like this is that there could be non-intentional perfect matches between records. For example, it is possible that both sources contain records from different clusters which agree in their attributes used for linkage, due to a false negative in the ground clusters or a common name and year of

Voter ID	Snapshot t	First name	Middle name	Last name	YOB	Zip	City
XY123	2008-11-04	VANESA	M.	PATEL	1977	27253	GRAHAM
	2009-10-07	VANESA	MARIE	PATEL	1977	27253	GRAHAM
	2012-11-06	VANESA	MARIE	PATEL	1977	27215	BURLINGTON
	2019-10-08	VANESA	MARIE	TAYLOR	1977	27215	BURLINGTON

Table 3.2: Example record cluster of a certain individual from temporal snapshots of a voter registry grouped by voter ID. The data shows different types of real-world errors which allows to derive different linkage problems based on selection.

birth in a high-density region. Those pairs would be classified as a match by all methods and are therefore not of interest for a comparative evaluation. Thus, the selector prevents selecting records from different clusters that fully agree in the name, year of birth and address.

D_M : DATASET SIZE AND OVERLAP MODIFIER

This component aims to create test datasets with varying sizes and levels of overlap among sources. Variations in size are crucial for studying scalability aspects. Greater overlap in general facilitates higher linkage quality. When the dataset contains true matching pairs for many of the records, the likelihood of false positive classifications decreases because genuine duplicates tend to have a higher similarity than random pairs. The overlap is typically not known prior to the linkage. Thus, conducting experiments on datasets with varying degrees of overlap permits an examination of the effects of these unknowns on the linkage result.

IMPLEMENTATION D_G and D_S are implemented as dedicated web services, enabling configuration-file-based creation and export of datasets as CSV files or JSON objects. These generated datasets, but also existing datasets from external sources can be imported into the data owner services – which are explained in the next section – in both formats as well. The component D_M is part of the implementations of D_C and D_S .

3.2.3 LINKAGE EXECUTION

The PPRL algorithm relies on components for data owners (L_{DO}) and linkage unit (L_{LU}). Both provide tags during the process, e.g., when calculating parameters dynamically, such as attribute weights unique to each record. These neither can be determined on plaintext before linking nor afterwards from the final linkage outcome. In addition, the protocol manager L_{PM} controls the configuration and data flow of both services.

L_{DO} : ENCODING AT DATA OWNERS

The original plaintext is irreversibly transformed to an encoded representation, as described in Section 2.3.3. The implementation comprises multiple encoding techniques which allows to test the effect on the linkage outcome (see also Section 3.3). Furthermore, the data owner module contains an analyzer component for plaintext records, allowing to capture aspects such as attribute lengths or missingness in tags for usage in the linkage result analysis. In practical use cases each data owner would deploy a dedicated L_{DO} . However, for evaluation purposes a single service is used to avoid the overhead of exchanging the secret encoding parameters.

 L_{LU} : (PRIVACY-PRESERVING) MATCHING

The linkage unit service receives the encoded records and applies a matching algorithm to determine entity clusters, as described in Section 2.3.4. The service supports multiple configurable classification models which allows the execution of conventional plaintext linkage algorithms as well. Intermediate and final linkage results including pairs are managed per project, separate from the records themselves, which allows to efficiently test multiple linkage approaches on the same (encoded) dataset. For evaluation purposes the linkage unit also supports adding a ground truth to the linkage problems in order to determine the linkage quality. Due to the rather simple matching approach in PPRL, insights on influencing factors of the encoding technique on the similarity score are valuable for method development. Thus, apart from tags describing the encoded dataset, $\tau_{rp}^E(SIM)$ is provided, encompassing the record pair similarity which is crucial for certain analysis types as outlined below.

 L_{PM} : PROTOCOL ORCHESTRATION

The protocol manager service orchestrates the data flow between the other components, including the data creation services. It follows a workflow definition of modular steps, such as in its simplest form by retrieving an encoded dataset from L_{DO} and passing it to L_{LU} , or more complex, dynamic flows as used in the multi-layer protocol in Chapter 5.

IMPLEMENTATION The PPRL services are implemented as web services and designed for managing multiple linkage projects in parallel. Thereby, it is possible to execute multiple linkages independently on the same L_{LU} instance. For each project, a secret has to be agreed on between the data owners. When requesting encoded records from the data owner service, a secret derivation algorithm is used, e.g., when multiple seeds for the cryptographic hash functions of each attribute are needed for employing attribute salting. Encoding, matching and protocol configurations as well as the datasets and record pairs are persisted independently from each other in MongoDB collections. For

testing purposes, they can be provided by the same database management system, while in practical use cases, distinct instances under control of the respective institutions would be used.

3.2.4 LINKAGE RESULT ANALYZER

This section outlines the analytical capabilities of our framework.

A_D : DATASET-LEVEL EVALUATION

To assess the overall quality of linkage, we apply the conventional performance metrics recall, precision, and F1 score. In the absence of ground truth, these metrics can be estimated in an unsupervised manner based on similarity graphs [41]. In addition to the usual confusion matrix, we differentiate between two types of false positives: FPs(ingleton), which are FP links where both records are true non-matches or singletons, and FPd(uplicate), which encompass all FP where at least one record is a true duplicate. High FPd/FP ratios suggest inadequate or missing postprocessing in linkage scenarios involving *clean* sources without internal duplicates.

A_G : GROUP-BASED QUALITY ANALYSIS

This component utilizes the tags provided by the data creation and linkage modules. The linkage outcomes are categorized by these tags to calculate quality measures for each subset. Examining the outcomes of a benchmark dataset allows observers to determine more effectively if these results might be applicable to their use case, even when using a different composition of records and types of data quality issues. We present an example for this analysis in Section 3.2.5.

A_P : PREDICTION-BASED ANALYSIS

This component is similar to the previous approach but reversed. It provides tools to outline and contrast the characteristics of specific groups in the outcome. Categorizing elements within the confusion matrix, such as exploring prevalent trends among FP or FN pairs, allows to pinpoint systematic deficiencies in the linkage method [30]. Furthermore, match predictions can be categorized by examining possible systematic disparities between records considered as matches or non-matches. These effects might arise from issues with the linkage algorithm in addressing data quality problems effectively within specific groups of records. Thus, A_P can also be applied in situations where the ground truth for the linkage problem is unavailable.

A_S : SIMILARITY ANALYSIS

This component enables an in-depth examination of the outcomes grounded in $\tau_{rp}^E(\text{SIM})$. Similar to previous methods, one can utilize either single tags or collections of tags to study the similarity distribution across the respective groups. It shows how effectively pairs can be distinguished either as *match* or *non-match* using the threshold-based classification models of PPRL with record-level encodings. Additionally, this module enables the examination of how similarity scores of specific record pairs vary with different encoding and matching techniques.

IMPLEMENTATION The analysis modules are implemented as a Python-based Streamlit² application using the web interfaces of the PPRL services for retrieval of the linkage outcome and its descriptive tags. Additionally, CSV export is supported, facilitating customized analysis utilizing external tools.

3.2.5 WORKFLOW EXAMPLE

In this section, we provide an illustrative example of how the framework enables evaluation of PPRL algorithms based on workflow definitions including dataset creation and linkage protocol orchestration. Figure 3.3 outlines the process. First, a synthetic dataset is created and subsequently modified to produce record variants that are either supposed to be matches (here: ID42-44) or non-matches (here: ID42-43 and ID42-45). Afterwards, a Bloom filter based linkage is conducted using the encoding and matching modules. The tags derived in this process can be used for different analysis types, as described in the following sections.

GROUP-SPECIFIC LINKAGE QUALITY ANALYSIS

While dataset-level linkage quality measures (A_D) allow to assess and compare the overall performance of PPRL algorithms, group-based analysis (A_G) reveals possible quality issues for certain subsets (see Figure 3.4). The upper result shows that the chosen linkage algorithm excels at identifying duplicates that differ by zero or one attribute, though it struggles when error rates are higher. Since attaining a high recall for records that are highly similar or identical is straightforward, this distinction enables focusing on the more complex cases, thereby disregarding the proportion of trivial matches that might lead to an overall high quality measure. The bottom result demonstrates that the linkage method employs encodings capable of handling slight attribute variations, such as typos, as well as expansions, like double last names or initials of first names.

²<https://streamlit.io/>

Step 0: Definition of the experiment workflow incl. configurations for D_G , D_C , L_{DO} , L_{LU}

Step 1: Generation of a synthetic dataset with household structures using D_G .

ID	First name	Last name	Date of birth	Zip	City
42	PAUL	POLLMANN	1979-10-25	73102	BIRENBACH
43	THOMAS	POLLMANN	1978-04-10	73102	BIRENBACH

Step 2: Creation of modified/corrupted records using D_C (here: based on record 42).

ID	First name	Last name	Date of birth	Zip	City
44	PAUL	POLLMANN-MEIER	1979-10-25	73102	BIRENBACH
45	PAULA	HOFFMANN	1979-10-25	74629	PFEDELBACH

Step 3: Import of generated linkage input dataset to L_{DO} and encoding.

Step 4: Transfer of encoded dataset to L_{LU} and matching.

Step 5: Detailed evaluation based on collected tags and partially using ground truth.

Tag store collects tags created during steps 1-4 (here: only tags related to record 42).

ID0	ID1	Origin	Attribute	Label	StringValue	Num.
42	43	D_G		RELATION	SIBLING	
42	44	D_C		MODIFIER	DOUBLE_LAST_NAME	
42	44	D_C	LASTNAME	EXTENDED		
42	45	D_C		MODIFIER	SIMILAR_BUT_OTHER_PLACE	
42	45	D_C	FIRSTNAME	REPLACEMENT	'similar'	0.80
42	45	D_C	LASTNAME	REPLACEMENT	'similar'	0.63
42	45	D_C	CITY	REPLACEMENT	'similar'	0.50
42	45	D_C	ZIP	REPLACEMENT	'dependsOnAttribute:CITY'	
42		L_{DO}	FIRSTNAME	LENGTH	'belowAverage'	4
42		L_{DO}	FIRSTNAME	REL_FREQ_RANK	VERY_FREQUENT	0.002
42		L_{DO}	CITY	REL_FREQ_RANK	VERY_RARE	0.851
42	45	L_{DO}	FIRSTNAME	MINOR	'->s'	
42	43	L_{LU}		SIM	'0.70-0.75'	0.716
42	44	L_{LU}		SIM	'0.85-0.90'	0.872
42	45	L_{LU}		SIM	'0.75-0.80'	0.763
42		L_{LU}		BF_FILLRATIO	'0.35-0.40'	0.387
42		L_{LU}	FIRSTNAME	BF_FILLRATIO	'0.35-0.40'	0.398
42	45	L_{LU}	FIRSTNAME	SIM	'0.85-0.90'	0.894

Figure 3.3: Example use of the framework based on the data creation path $D_G \rightarrow D_C$: The steps for generating the linkage input datasets, executing the PPRL protocol and subsequent result analysis are illustrated using example records. The table on the bottom shows corresponding tags, describing plaintext and encoded records as well as record pairs. The lower part of the table illustrates selected tags from another experiment using the same dataset but a different (attribute-level) encoding.

#Attr.diff	Recall	Prec.	F1	TP	FP	FN	FPs	FPd
0	0.9999	0.9999	0.9999	10628	1	1	0	1
1	0.9909	0.9916	0.9912	3793	32	35	32	0
2	0.6519	0.7239	0.686	354	135	189	133	2

Diff type	Recall	Prec.	F1	TP	FP	FN	FPs	FPd
MINOR	0.9575	0.9955	0.9761	1758	8	78	8	0
REPLACEMENT	0	0	0	0	47	0	47	0
EXTENDED	0.9615	0.9868	0.974	300	4	12	4	0

Figure 3.4: Example for group-based quality assessment using $\tau_{rp}^P[L_{DO}]$ for the number of differing input attributes (top) and $\tau_{ap}^P[L_{DO}]$ for descriptions of fine-grained corruption types (bottom).

PREDICTION-BASED ANALYSIS

To investigate potential biases in the linkage outcome, it is advantageous to examine the distributions of specific tags by prediction type (A_P), as depicted in Figure 3.5. Differences in the distributions are not necessarily indicators of systematic linkage flaws if these reflect real-world differences between records belonging to the match or non-match group. Nevertheless, it is improbable that rare or common names, or low fill ratios, which correlate with shorter plaintext values, belong to this category. Therefore, methodological adjustments such as integration of value-specific attribute weights should be considered.

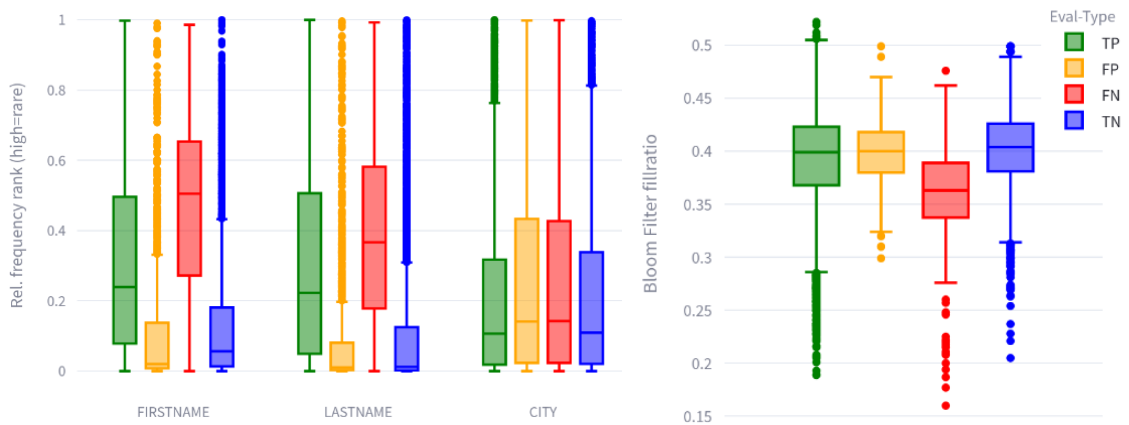


Figure 3.5: Example for prediction-based analysis on distributions of attribute frequencies (left, $\tau_a^P[L_{DO}]$) and Bloom filter fill ratios (right, $\tau_r^E[L_{LU}]$) indicates correlation of common names with false positives and of low fill ratios with false negatives.

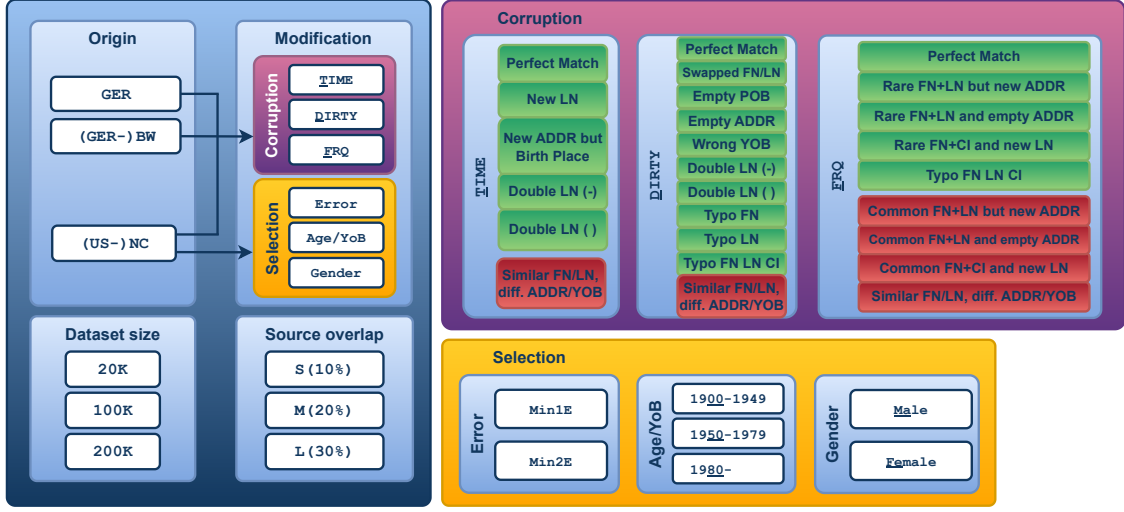


Figure 3.6: Overview of linkage problem dimensions and their creation scenarios. Datasets from different geographical regions are modified, either using corruption configurations for creating true matching or true non-matching pairs, or using selection based on dissimilarity, age or gender from real-world clusters. Datasets of varying size and overlap between sources are created.

3.3 ROBUSTNESS EVALUATION

In this section, we evaluate the linkage quality of existing PPRL methods in particular with regard to their robustness for different datasets. We compare these results with a baseline using plaintext data. The outcomes serve as baselines for our methodological contributions in subsequent chapters.

3.3.1 LINKAGE PROBLEM CREATION

We employ the data creation modules of the proposed framework to derive multiple linkage problems used throughout this thesis. Figure 3.6 gives an overview of the dimensions of these benchmark datasets.

BASE DATASETS We aim to evaluate the linkage methods on multiple datasets with different characteristics, in particular being from different geographical regions. We use D_G to generate a dataset based on German census data (*GER*). We modify the dataset to create a second data source for matching using multiple corruption scenarios as outlined below. Additionally, we use D_S to select real-world datasets based on the US voter registry from North Carolina (*NC*). For better comparability of results from both regions, we introduce another origin dataset restricted to the German federate state of Baden-Württemberg (*BW*) as it has approximately the same population size of 11 million residents as North Carolina.

MODIFICATIONS For the selection-based dataset creation, we intentionally do not include matches with perfect agreement as those are trivial to match by any method. The duplicates are created by randomly sampling records from snapshot '2021-01-01' and using a duplicate from a different snapshot where at least one (*min1E*) or two (*min2E*) of the attributes or attribute combination *first name*, *middle name*, *last name*, *place of birth* and *city and zip code* are non-equal. These basic datasets represent the voters of North Carolina and thereby a large proportion of the population. As such, this is helpful to simulate linkages on population databases. However as described in Section 2.4, data linkages are also conducted for specific use cases where only subsets are selected at each data owner prior to the linkage depending on the specific need of data analysts. These may include for example age-based or gender-based cohorts. Those subsets have different characteristics and thus likely require different parameters for optimal linkage quality. The subgroup selection strategy is based on the *min1E* configuration. From the potential records and pairs only those are selected where at least one fulfills the content condition. We create three age cohorts having year of birth values between 1900 and 1949, 1950 and 1979, and above 1980. In addition, we create two subgroups containing only persons having a gender attribute value of *male* or *female* respectively.

For both base datasets from Germany, GER and BW, only a single record is available for each entity. Therefore we apply the corruption module D_C to derive linkage problems using different data quality scenarios, as described in Figure 3.6. In order to better compare linkage methods on datasets from regions with different language, we also create linkage problems with these corruption scenarios on the base dataset from North Carolina where only a single record is used from each cluster.

TIME represents a scenario where true matches exhibit differences corresponding to typical timely record changes, e.g., changed last name values due to marriage/divorce (replaced or extended attributes) or new addresses due to relocations (replaced attributes). *DIRTY* focuses more on differences that arise due to limited data quality standards which results in missing values, typos, exchanged or extended attributes. *FRQ* contains pairs with identical modification configurations where the ground truth label depends on the frequency of attribute values. For example, two records sharing first name, last name and year of birth while having a different location entry are considered a match only if both name components are rare. We define *common* as being in the top 2 % of unique attribute values whereas *rare* means that the value is in the bottom 65 % of unique values. These boundary values were determined by the practical requirement of obtaining a sufficient number of suitable original records that meet these conditions in the selection process to achieve the desired linkage problem sizes. The purpose of this corruption scenario is the creation of tagged pairs with relatively more common or rare attribute values; the precise frequencies are not critical and could be adjusted if needed.

All three scenarios also contain a small amount of perfect matches as a sanity check for the linkage methods as those pairs should be predicted as a match by any approach. All scenarios also include a proportion of pairs that share the same last name and have a similar first name, or vice versa, but have different year of birth, city and zip code values. Those pairs are labeled as true non-matches and are included to test the ability of the linkage methods to prevent false positive classifications of similar but still unambiguous non-matches.

SIZE AND OVERLAP We create linkage problems with two equally sized data sources having overlaps – share of common entities – of 10 %, 20 % and 30 % and a total size of 20000, 100000 and 200000 records.

RANDOM SEED For each of those linkage problem creation pipelines, we create two datasets using different seeds for the (random) sampling components while keeping all other parameters the same. This allows to check if possible effects are due to, e.g., a specific modification scenario, or an artifact due to the randomness.

DATASET NAMING The dataset notation is composed of these components. For example, BW-T-20S refers to a dataset from Baden-Württemberg in Germany, corrupted using the TIME scenario which results in 20000 records in total and an overlap of 10 % (S). NC-Yob50-100L refers to a dataset from North Carolina with 100000 records in total, an overlap of 30 % (L) and a selection configuration where only persons having year of birth (yob) values between 1950 and 1979 are included.

3.3.2 DATASET COMPARISON

In the following, the characteristics of the created linkage problems are compared. We use dataset variants of GER, BW and NC, each corrupted with TIME, DIRTY, FRQ. Furthermore, we include the 2 error-based and 5 subgroup-based selection settings based on the NC dataset. We restrict the comparison to the 100L size and overlap datasets. In total, this yields 16 linkage problem configurations. We focus on the following 7 attributes which are relevant to the linkage methods used in this work: First name (FN), middle name (MN), last name (LN), year of birth (YOB), city, ZIP code (which corresponds to PLZ or 'Postleitzahl' in Germany) and place of birth (POB). In the following, selected dataset characteristics are shown.

Figure 3.7 and Figure 3.8 describe the attribute lengths and missingness of the datasets. Overall, the GER and BW datasets differ from the NC datasets in that they do not include a middle name attribute and contain substantially longer place of birth values (~ 13 vs 2). The latter is due to different value representations. While the attribute holds the full

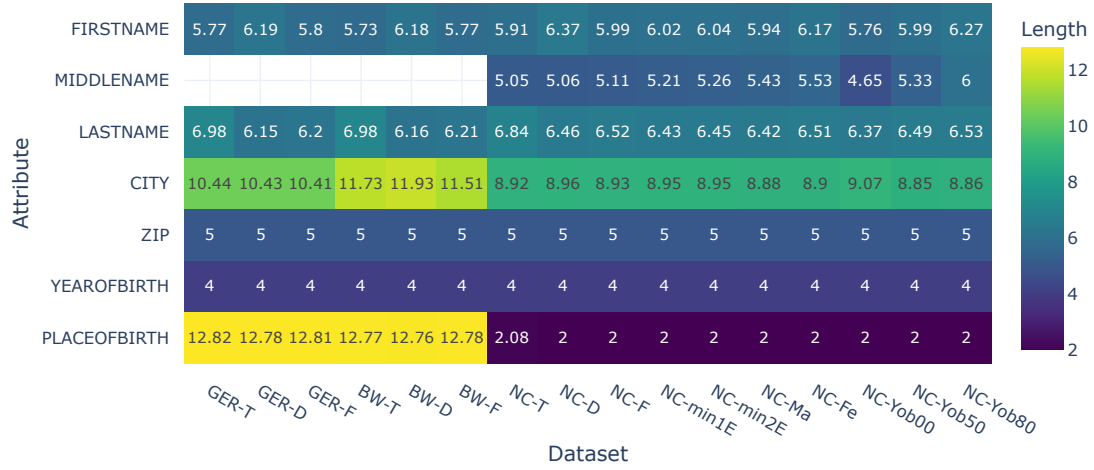


Figure 3.7: Comparison of attribute lengths of the linkage datasets created using the corruption module on base datasets from Germany (GER), Baden-Württemberg (BW) and North Carolina (NC) as well as linkage datasets derived by selection from the NC voter registry.

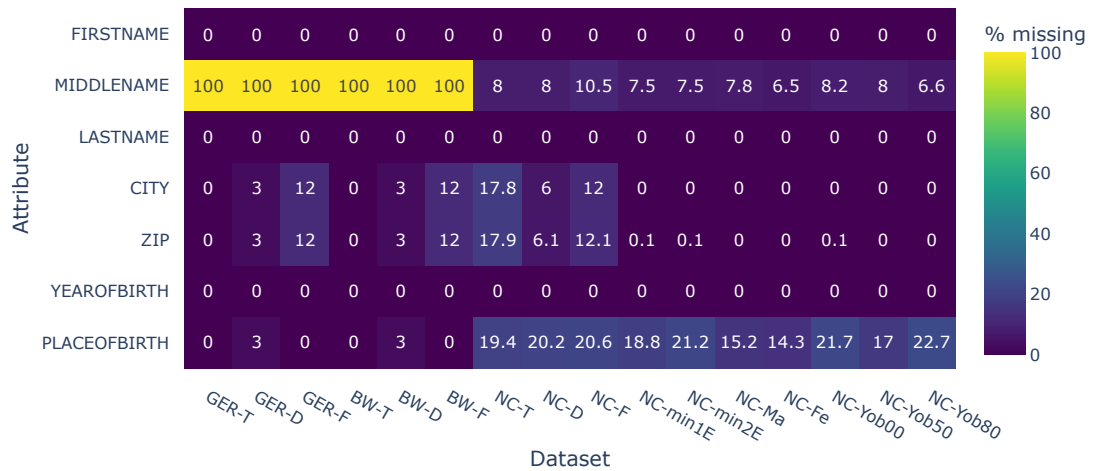


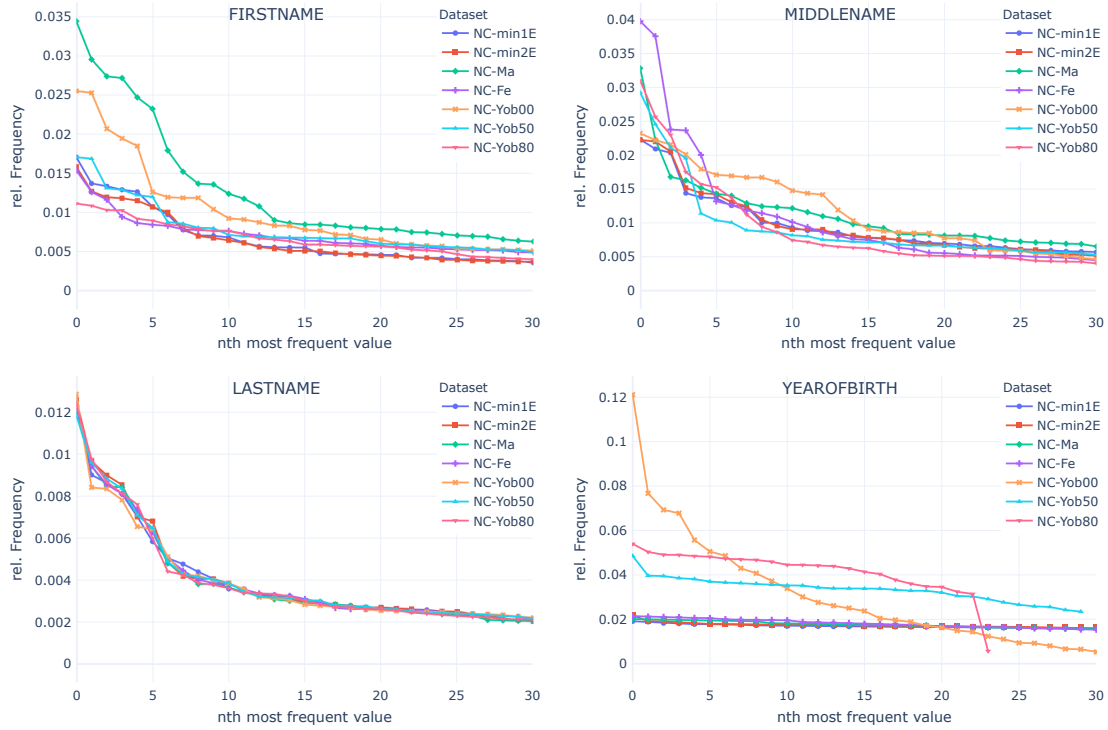
Figure 3.8: Comparison of attribute missingness of the linkage datasets created using the corruption module on base datasets from Germany (GER), Baden-Württemberg (BW) and North Carolina (NC) as well as linkage datasets derived by selection from the NC voter registry.

federal state of birth in GER and BW, it contains a two-character state abbreviation for NC. Furthermore, the city values of BW are on average the longest (~ 11.7), whereas they are shorter for GER (~ 10.4) and NC (~ 8.9). For the age-based subgroups of NC, it can be observed that records of older individuals in Yob00 have, on average, shorter middle name lengths (4.65) than those in Yob50 (5.3) and Yob80 (6). The list of most common values in Figure 3.9 indicates that this is caused by a higher proportion of entries consisting of a single character, namely the initial. The other attributes have comparable lengths in general. None of the datasets contain missing values for the attributes first name, last name, or year of birth. The German-based datasets likewise show no missing values for the place-of-birth attribute, except in the *DIRTY* corruption scenario, where this attribute is removed for some records. The NC datasets, in contrast, exhibit high rates of missing values for this attribute as well as for the middle name.

Figure 3.9 shows the frequency distributions of the most common values of selected attributes from the datasets derived by selection from the NC voter registry. While the distributions of last names are very similar, this does not hold for first and middle names or for year of birth. The most frequent first names in NC-Ma and NC-Yob00 occur more often than in the other datasets, especially compared to NC-Fe and the remaining age groups. The year of birth distributions naturally shows higher frequencies for the age based subgroups as they contain only values within a certain range. NC-Yob00 exhibits a relatively high occurrence of the value '1900', as this value seems to serve as a placeholder when data is missing.

The frequency distributions of selected attributes from the datasets derived for GER and BW are depicted in Figure 3.10. The distribution of first name values shows higher frequencies for the most common values in F than in the T and D variants. This occurs because the corruption procedure explicitly selects records containing rare and common values for the output dataset, which in turn produces the observed distortions. The BW datasets exhibit higher frequencies for the most common city and zip code values, which results from the limitation to this federal state rather than covering the entire country.

Finally, we examine the characteristics of the matched records based on the ground truth, see Figure 3.11. The core idea is that the linkage problems vary in how types of differences between corresponding records are distributed. This is particularly true for the datasets generated through corruption, as different scenarios were employed during their creation. For all record pairs considered true matches, difference schemata based on an attribute-wise comparison are determined. Attributes are compared using the Levenshtein similarity and categorized as being either *equal*, *very similar* ($sim > 0.8$), *very dissimilar* ($sim < 0.2$), or as *medium similar* if they fall in between these thresholds. If one or both attribute values are missing, respective categories are assigned. This resulted in a difference schema for each record pair. The figure shows the most frequent difference patterns for each linkage problem.



	n	NC-min1E	NC-min2E	NC-Ma	NC-Fe	NC-Yob00	NC-Yob50	NC-Yob80
first name	1	james (1.71)	james (1.58)	james (3.44)	mary (1.54)	james (2.55)	michael (1.71)	jessica (1.11)
	2	michael (1.37)	john (1.27)	michael (2.95)	jennifer (1.26)	mary (2.53)	james (1.69)	michael (1.09)
	3	robert (1.34)	william (1.20)	john (2.74)	elizabeth (1.16)	william (2.07)	david (1.31)	christopher (1.03)
	4	william (1.29)	michael (1.18)	william (2.72)	jessica (0.94)	john (1.95)	john (1.29)	ashley (1.03)
	5	john (1.27)	robert (1.15)	robert (2.47)	linda (0.87)	robert (1.85)	william (1.22)	matthew (0.92)
middle name	1	lee (2.23)	marie (2.23)	lee (3.28)	marie (3.98)	m (2.32)	ann (2.93)	marie (3.10)
	2	ann (2.09)	ann (2.20)	michael (2.23)	ann (3.75)	lee (2.22)	lee (2.46)	nicole (2.56)
	3	marie (2.03)	lee (2.05)	edward (1.68)	lynn (2.38)	l (2.16)	marie (2.11)	elizabeth (2.30)
	4	lynn (1.44)	lynn (1.52)	james (1.63)	elizabeth (2.37)	ann (2.01)	lynn (1.95)	lee (1.75)
	5	l (1.38)	l (1.44)	wayne (1.52)	nicole (2.00)	c (1.80)	l (1.14)	lynn (1.57)
last name	1	smith (1.25)	smith (1.26)	smith (1.19)	smith (1.23)	smith (1.28)	smith (1.18)	smith (1.23)
	2	williams (0.90)	williams (0.97)	williams (0.94)	williams (0.94)	jones (0.84)	williams (0.96)	williams (0.98)
	3	johnson (0.86)	johnson (0.90)	johnson (0.85)	johnson (0.85)	johnson (0.83)	johnson (0.88)	johnson (0.87)
	4	jones (0.81)	jones (0.85)	jones (0.84)	jones (0.81)	williams (0.78)	jones (0.84)	jones (0.81)
	5	brown (0.70)	brown (0.70)	brown (0.73)	brown (0.74)	brown (0.65)	brown (0.71)	brown (0.76)
year of birth	1	1994 (1.91)	1994 (2.18)	1990 (2.01)	1990 (2.14)	1900 (12.12)	1979 (4.85)	1980 (5.40)
	2	1971 (1.88)	1970 (1.92)	1900 (2.00)	1986 (2.12)	1949 (7.67)	1971 (3.96)	1990 (5.03)
	3	1979 (1.83)	1979 (1.91)	1986 (1.99)	1989 (2.10)	1947 (6.93)	1970 (3.95)	1985 (4.91)
	4	1964 (1.81)	1971 (1.87)	1989 (1.96)	1985 (2.09)	1948 (6.78)	1977 (3.85)	1986 (4.90)
	5	1970 (1.78)	1992 (1.82)	1988 (1.96)	1984 (2.06)	1946 (5.57)	1978 (3.81)	1989 (4.85)

Figure 3.9: Frequency distributions of the most common values of selected attributes in the linkage datasets derived from North Carolina voter registry (top) and the most common values and their relative frequency in % (bottom).

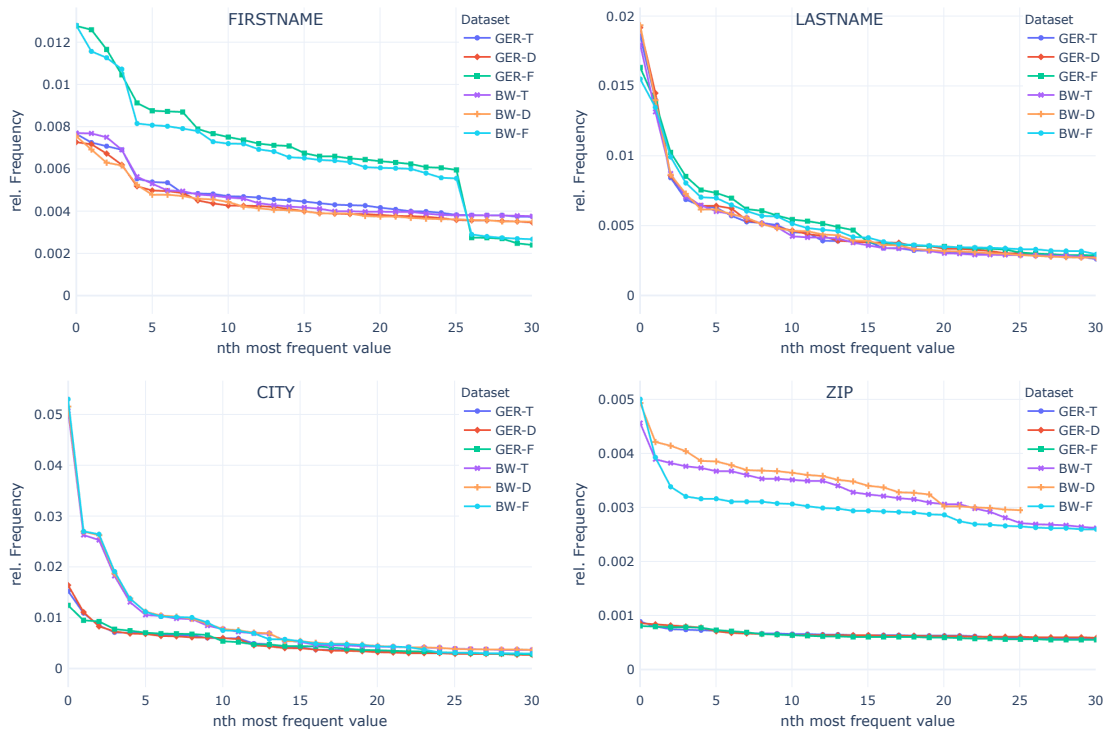


Figure 3.10: Frequency distributions of the most common values of selected attributes in the linkage datasets based on the generator for GER and BW.



Figure 3.11: Overview of the most common difference patterns of true matches in the linkage datasets derived from the North Carolina voter registry.

Only the corrupted datasets contain matches which perfectly agree. In the *TIME* scenario the most common types of differences affect the last name and address attributes, whereas in the *DIRTY* setting, the most common issues are missing address values and differences in the first and last name attributes. The *FRQ* data shows modifications of last name and address in the top 2-4 patterns. Both *DIRTY* and *FRQ* contain pairs with similar values for first name, last name and city whereby the latter is categorized as *medium similar* due to character swap modifier which results in lower similarity scores than the single character replacements applied on the name components. These patterns are expected and in line with the applied record modifications as described in Figure 3.6. The subgroup datasets in the second row show less diverse similarity patterns. Most groups frequently contain duplicates with differing address and missing place of birth. While in NC-Fe the last name is most frequently affected, the dataset containing records with a 'male' gender value is not. The age based datasets contain more address changes in younger groups.

Overall, these analysis results show that the generated linkage problems differ in terms of attribute length, degree of missingness, frequency distributions, and the kinds of discrepancies between duplicate records. All of these factors are important for configuring record linkage algorithms and thus enable a more comprehensive assessment of linkage methods.

3.3.3 LINKAGE METHODS

We evaluate linkage strategies of perturbation-based PPRL with record-level encodings and compare them with plaintext-based record linkage on these linkage problems in order to study possible detrimental effects.

BLOCKING

In this work, we focus on the evaluation of PPRL techniques with regard to comparison and classification quality rather than scalability. However, to run linkage experiments on larger datasets in a reasonable time, a blocking strategy needs to be employed. We use standard blocking to reduce the number of comparisons that need to be computed, as described in Section 2.1.2. To ensure the comparability of the results, we define blocking keys independent of the encoding method so that the same set of match candidates is used in all linkage methods on a certain dataset. For each record we generate blocking keys at the data owners based on the plaintext attribute combinations {FN, YOB}, {LN, YOB} and {Soundex(FN), Soundex(LN)} and encode them using a cryptographic one-way hash function. These hashed blocking keys are transmitted together with the encoded records to enable standard blocking at the linkage unit.

However, this may lead to a *glass ceiling* of the linkage quality results for all techniques as the highest achievable recall is determined by the pair completeness score (PC) of the blocking phase. One possible strategy would be to use the remaining pairs after blocking as the reference of the true matches to be found. However, this may cause distortions as the missed true matches due to blocking could be meaningful when comparing (PP)RL techniques because they may be the *hard* ones where differences between methods become visible. Another strategy is the inclusion of a blocking key from the global record id that is unique for each duplicate pair based on the ground truth. This blocking key ensures that no true match is excluded from the comparison. The impact of the additional blocking function is evaluated in the result section below.

ENCODINGS

Record-level PPRL methods are typically less flexible as important parameters such as attribute weights have to be chosen before the encoding at the data owners and thus before having any information from comparing records of different sources. In addition, record-level encodings conceal record details from the linkage unit which imposes strong limitations on the applicable classification models. Thus, the choice of the encoding technique and its parameters is crucial for the linkage quality outcome.

Bloom filter encoding parameter choices including hardening techniques have been studied before [9, 46]. Hardening techniques aim to reduce the feasibility of attacks by (further) obfuscating characteristics of the original plaintext datasets. Thus, the linkage quality is typically lower or equal but not higher than with *basic* encodings. We do not apply any hardenings apart from attribute salting, as it has been shown to reduce the privacy risk while achieving comparable or even better linkage quality [46]. We choose this setup because we want to study how much worse the linkage quality is *at least* when using PPRL in contrast to plaintext linkage.

Weighting A very important subset of parameters concerns weights which are incorporated in the encoding process to reflect the different discriminatory power of attributes. We define three static weight scenarios as described in Table 3.3.

Scenario	FN	MN	LN	YOB	CITY	ZIP	POB
Equal	1	1	1	1	1	1	1
Core	2	1	2	2	1	1	1
Core+	3	2	3	3	1	1	2

Table 3.3: Weight set scenarios in the robustness evaluation: Equal uses the same weight for all of the attributes, where as Core doubles the weights of three essential attributes and Core+ further increases these weights and lowers the relative weight of residential attributes.

Bloom filter generation The plaintext attributes are preprocessed by removing leading and trailing whitespace, conversion to lowercase and removal of diacritics before being split into overlapping bigrams using padding. For the record-level Bloom filter we use $l = 1024$. The number of hash functions h_i is selected based on the average length of each attribute to achieve an average fill rate of the respective Bloom filter of approximately 40 %. We use the CLK-RBF encoding approach to incorporate weights in record level encodings as described in Section 2.3.3. The number of set bit positions related to a certain attribute depends on the (average) attribute length. Shorter values consist of fewer tokens and thereby fewer bits are set. Thus, we compute h_i with the following equation to ensure that the average number of hash functions of each attribute with respect to the total number of hash functions is proportional to the relative weight of this attribute.

$$\frac{h_i \cdot n_i}{h \cdot \sum_{i=0}^{N-1} n_i} = \frac{w_i}{\sum_{i=0}^{N-1} w_i} \quad \rightarrow \quad h_i = \frac{w_i \cdot h \cdot \sum_{i=0}^{N-1} n_i}{n_i \cdot \sum_{i=0}^{N-1} w_i} \quad (3.1)$$

where w_i is the weight, n_i the average number of features, and h_i the number of hash functions for attribute i . h is the reference number of hash functions and determines the average fill rate (amount of 1-bits relative to the length l) of the Bloom filters.

Please note that h_i is computed based on the *average* attribute length which still leads to weight variations depending on the actual length of each attribute due to the interdependency of length, number of tokens and weight. For example, when using Bloom filter encodings with bigrams, padding and $h_i = 12$, an attribute of length 4 results in $(4+1) \cdot 12 = 60$ bit selections. An attribute of doubled length 8 results in $(8+1) \cdot 12 = 108$ bit selections, which corresponds approximately to a weight increase factor of 1.8. Thus, these record-level encodings employ weights differently than weighted attribute-level similarity aggregation.

BASELINE As a baseline for state-of-the-art plaintext linkage, we apply Unicorn [145] on the benchmark datasets. Unicorn aims to provide a unified data matching model based on three deep neural networks: an encoder which converts record pairs of different tasks into a shared representation, a mixture-of-experts model (MoE) that enhances these representations by aligning the matching semantics to improve the performance of the final binary classification model. We finetune the pretrained *UnicornPlus* model using the provided script with its default parameters³ based on the DeBERTa language model [59]. The dataset for this task was produced with the same configuration as NC-min1E-100L, but using a different sampling seed in D_S , resulting in an independent training set. The training samples are created using the ground truth based plaintext blocking strategy

³<https://github.com/ruc-datalab/Unicorn> (Commit 5424e585f6c739db3c1ca610e0db079392dfbfd6)

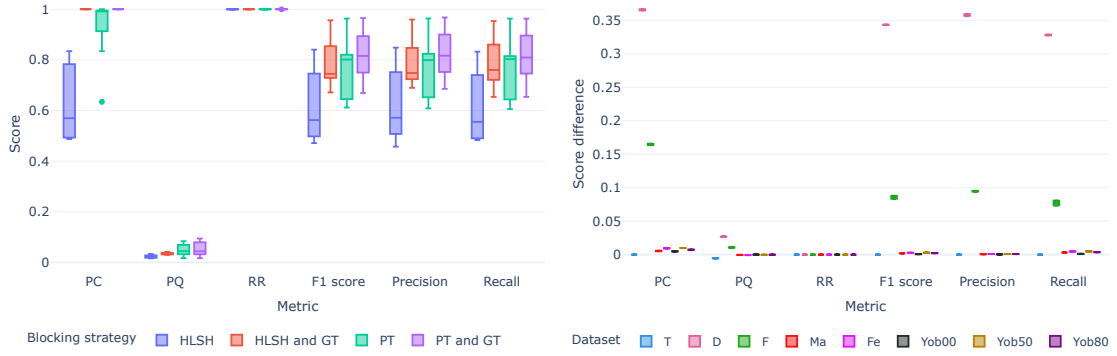


Figure 3.12: Comparison of quality metrics using different blocking strategies: On the left, plaintext (PT) and HLSH results, each with and without ground truth IDs (GT) are shown. On the right, score differences by inclusion of GT to plaintext blocking are shown for each dataset.

outlined above. A class balancing using random downsampling is employed which yields 15 000 positive and negative samples respectively.

METRICS We compare different linkage methods on various datasets. As noted by [57] it is pointless to compare linkage quality metrics for a certain threshold between methods that use different similarity functions as the scores may be unrelated. Therefore, we report the results for thresholds that lead to an equal and constant weighting of recall and precision according to [57], as described in Section 2.2.1.

3.3.4 RESULTS

BLOCKING

We conduct the following experiments to assess the impact of the chosen blocking strategy. We use record-level CLK-RBF Bloom filter encodings with *equal* weights for all NC dataset corruption and subgroup variants of 100 000 records and an overlap of 30 % (100L). In addition to the plaintext blocking, we include HLSH blocking (see Section 2.3.4) for comparison. Based on the results of [44], we use $\Lambda = 15$ keys as it yields the best results with respect to the PC measure in their study. We choose the length $\Psi = 24$ to produce approximately the same amount of match candidates ($\sim 400\,000$) as the plaintext strategy ($\sim 360\,000$). For each linkage, we determine the blocking quality measures PC, PQ and RR, as well as the linkage quality measures recall, precision and F1 score.

Figure 3.12 shows the overall results of this blocking experiment. Without the ground truth blocking key, the LSH-based blocking strategy has a low median PC score of 0.57, resulting in even lower recall values, whereas the plaintext blocking strategy reaches very high PC scores (> 0.99). As intended, the addition of the ground truth blocking key leads to perfect PC scores as well as small improvements of PQ. Furthermore, it has only a very low effect on the median results with plaintext blocking keys, but strongly improves

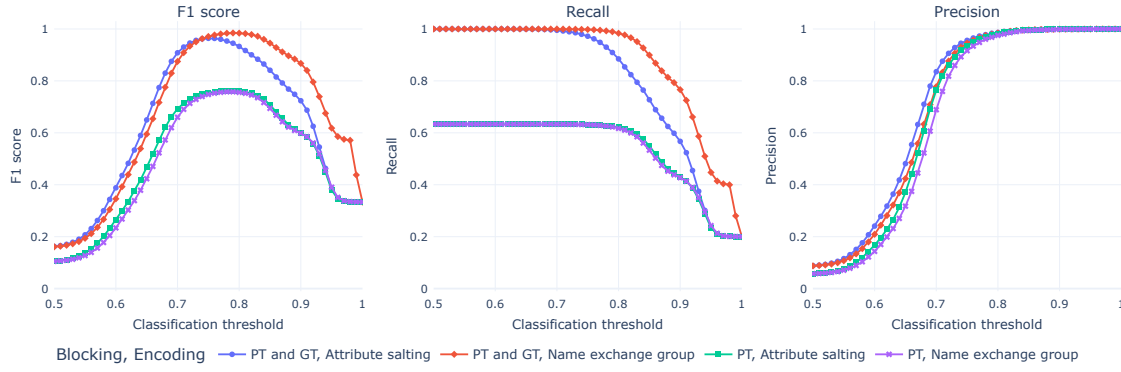


Figure 3.13: Comparison of linkage quality results of different blocking strategies on plaintext (PT) with and without ground truth IDs (GT) and encoding strategies for the dataset NC-D-100L.

the recall of the HLSH-based strategy. On the right side of Figure 3.12, the measures changes of the plaintext blocking strategy by inclusion of the ground truth blocking keys are shown for each linkage problem. The blocking strategy modification has almost no impact on most datasets, but it substantially increases PC, F1 score, precision, and recall for NC-F and especially for NC-D. The reason in the latter case is that it contains true matches with exchanged first and last name. All three blocking keys include at least one of these components, which causes these pairs to be dropped during blocking when the ground truth blocking key is not used.

Figure 3.13 shows another aspect of the ground truth based blocking strategy. Without it, the differences of record-level Bloom filter encoding configurations which either use distinct attribute salting for all fields or identical hashing seeds for the name-based attributes (FN, MN, LN) are hardly visible. When including all pairs in the comparison, it is revealed that the name-exchange-group approach strongly improves the F1 score for higher threshold values due to improved recall and only comparably reduces the precision. Based on these observations the ground truth and plaintext based blocking strategy is employed in this thesis.

ENCODINGS

The following results are based on the 20S, 20L, 100S, 100L size and overlap variants of the three corruption scenarios (TIME, DIRTY, FRQ) of each origin (GER, BW, NC) as well as the five age- and gender-based subgroups from NC, each in two variants using different seeds for the generation process. In total, this yields 112 linkage problems.

WEIGHTS: We study the impact of the weight scenarios on the linkage quality when using record-level encodings. Figure 3.14 shows that in scenario D, the *Equal* weight set clearly yields the best performance, whereas for F, *Equal* performs only slightly better than *Core* but substantially better than *Core+*. Scenario T is less sensitive to the choice of

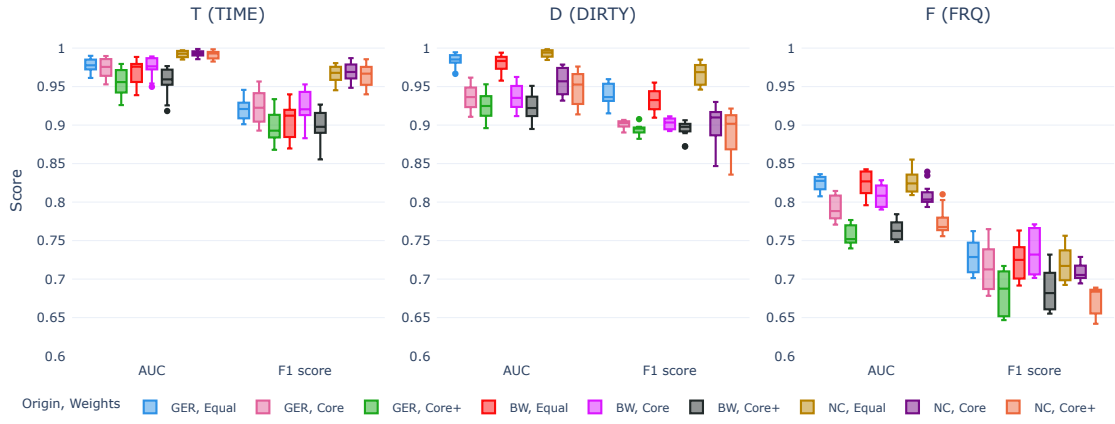


Figure 3.14: Linkage quality with different weight sets based datasets from the origins GER, BW and NC and different corruption scenarios T, D and F.

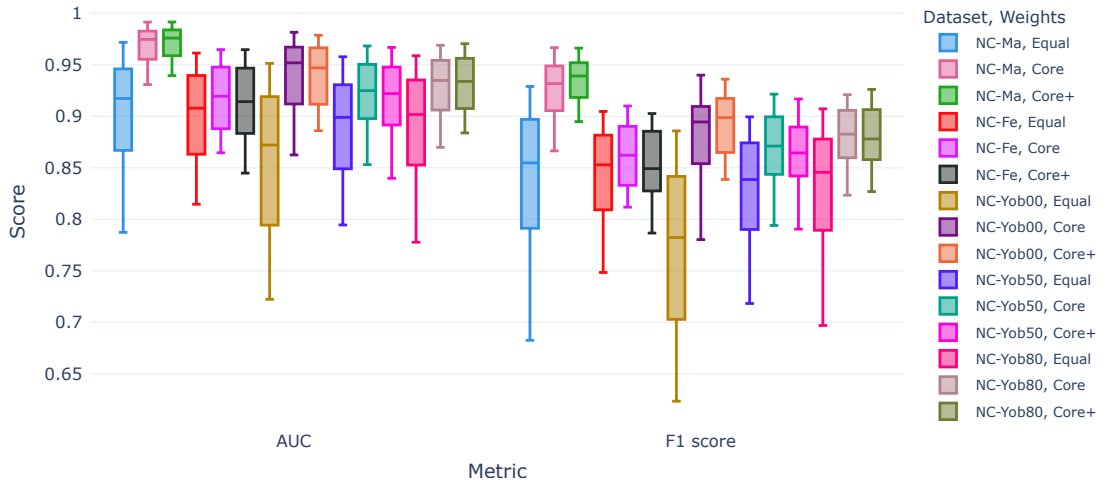


Figure 3.15: Linkage quality with different weight sets on subgroups datasets from North Carolina (NC).

weight set, although *Core* is slightly superior in particular with regard to the F1 score. Overall, the linkage quality for F is considerably worse than in the other settings, since matching and non-matching pairs often differ only in the frequency of attributes, which makes it necessary to rely on background knowledge about frequency distributions to separate them.

Figure 3.15 shows that, overall, the *Core* and *Core+* weight sets achieve similar performance, while *Equal* performs noticeably worse with respect to both quality measures. *Core* and *Core+* exhibit comparable performance across the age groups, whereas AUC and F1 score are both substantially higher for Ma than for Fe. The latter can be explained by the high rate of changes of last name attributes in Fe, as illustrated in Figure 3.11, and the increased weight assigned to this attribute in *Core* and *Core+*.

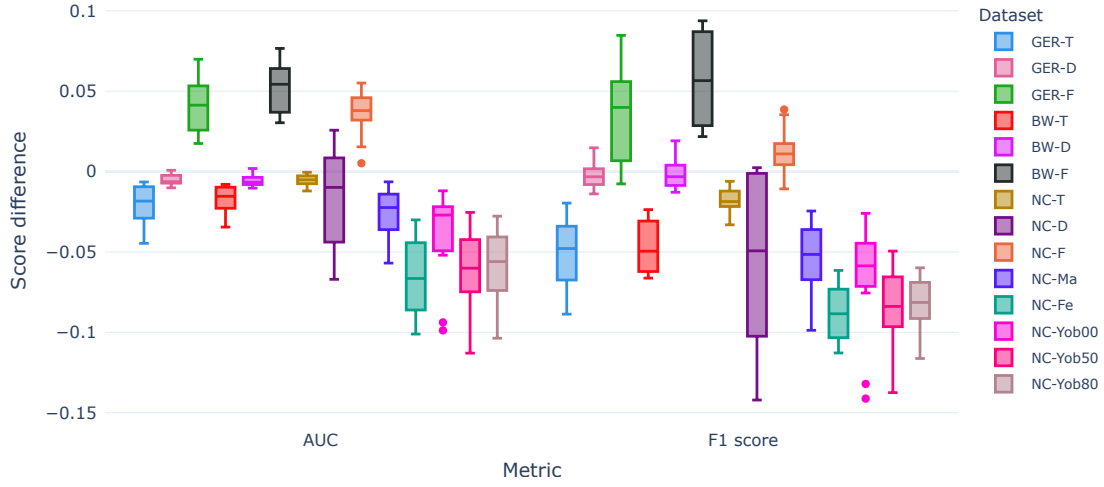


Figure 3.16: Linkage quality differences of PPRL with record-level encodings (*Equal weights*) to a plaintext baseline using the Unicorn deep learning model. Negative values indicate that this baseline performs better.

In the following, we restrict our analysis to the *Core* weight set, since *Equal* performs poorly on the subgroup linkage tasks, while *Core+* exhibits weaknesses across all corruption-based datasets and achieves comparable results to *Core* on the subgroup datasets.

COMPARISON TO PLAINTEXT BASELINE Figure 3.16 compares the results when using weighted record-level Bloom filter encodings to the plaintext baseline using Unicorn. The differences of the quality metrics are determined for each dataset, where negative values indicate that the baseline achieves superior results. For the corrupted datasets, we find that Unicorn performs better on T, increasing the median F1 score by 0.05 for GER and BW, and by 0.02 for NC. The robustness, however, does not improve to the same extent, which is similarly observed for D, where both AUC and F1 score, are on par. An exception is NC-D, for which Unicorn again attains a higher median F1 score by 0.05. For F, in contrast, the situation is reversed and the PPRL approach yields superior linkage quality.

Unicorn addresses the subgroup linkage problems more effectively, resulting in increases of AUC between 0.02 and 0.07 and of the F1 score between 0.05 and 0.09. This is, however, to be expected, since the model is finetuned on a generalized variant of NC. Nevertheless, the trained model seems to have learned similarity patterns of how records change over time, since it also shows strong performance on the T linkage problems from Germany. However, the notion that the importance of agreeing attributes depends on how frequently they occur is not taken into account. The superior performance of Unicorn on NC-T compared to GER-T and BW-T suggests that the difference may be partly due to insufficient background knowledge for non-English attribute values.

3.4 DISCUSSION

The proposed framework comprises multiple dataset creation paths which enables choosing the best approach for each use case. Generally, using real-world datasets $\mathbf{R}$ is preferred since they don't require generator and corrupter modules parametrization. Dataset size and overlap can be modified to assess scalability and align the problem with the expected number of matches for the intended use case. The selection component can also be used for assessing PPRL methods in scenarios where datasets are not representative for the entire population, for example when linking a student database with records from a local residents' registration office in a certain city. The corrupter module allows to simulate various data quality scenarios to assess their impact on linkage outcomes, using a single-source dataset $\mathbf{R}_i$ as input, such as a data owner's database. If no real-world dataset $\mathbf{R}_i$ is available, the generator component can synthesize databases that mimic real-world characteristics. It should be noted that our implementation for creating German population records requires further evaluation in that regard, which was out of scope for this work. Furthermore, the corruption methodology enables researchers to construct datasets with particular characteristics that are in the focus of newly developed PPRL encoding techniques, such as the adaptability to value frequencies.

The framework is designed with perturbation-based PPRL in mind, focusing in particular on Bloom filter encodings. However, the proposed evaluation framework can be applied to alternative externally managed linkage protocols by retrieving plaintext records from the data owner services, executing the external linkage protocol, and subsequently importing the results to the linkage unit module L_{LU} , potentially accompanied by descriptive tags.

The analytical components facilitate comprehensive insights into PPRL algorithms and the comparative evaluation to study applicability in given real-world scenarios. These methods depend on tags, which must be provided by data generation and linkage algorithm modules. Therefore, extending current data generation and corruption tools to include descriptive tags in their outputs would be advantageous, as it would allow their integration into the framework. Enhancing transparency on potential sources of linkage bias in PPRL is an initial step towards a more cohesive integration between the data linkage process and the subsequent data analysis. While this work focuses on analyzing the linkage quality in the PPRL context, it is worthwhile to extend the framework to study privacy aspects of encodings as well including assessment of risks related to reidentification attacks.

We created a variety of datasets for studying linkage algorithms that differ in data aspects highly relevant for this task such as the frequency distributions of attribute values, as shown in Section 3.3.2. One can certainly argue about the chosen configuration of the synthetically corrupted datasets. The scenarios are not designed to be realistic in

terms of their composition of error types. Instead, they incorporate common kinds of discrepancies, making them useful for examining how PPRL encoding methods affect these sets of record pairs, for example with exchanged attributes (see Figure 3.13). The subgroup datasets have a more realistic composition of data quality issues due to their real-world origin, apart from the exclusion of perfectly agreeing duplicates.

Assigning ground truth labels to generated pairs is not unambiguous and thus are the linkage quality metrics which are computed based on them. The question is which level of difference should be considered as a match and which as a non-match. For example, consider a record pair with equal first name, last name and date of birth but different residential attributes like city and zip code. If the name is not very common, this is intuitively the same person which moved and thus, a match. Nevertheless, there is still a degree of uncertainty. When using a real-world dataset like a voter registry, the uniquely identifying voter ID can be used to resolve that ambiguity. The voter ID assignment is usually based on additional verification during the registration process, e.g., based on a drivers license. Thus, the ground truth of the North Carolina voter registry is mostly accurate which makes this data source suitable for evaluations of record linkage algorithms [98].

For synthetic dataset generation and corruption, there is no additional information and the ground truth labels are assigned based on assumptions of the operator. The example with the presumably relocated person above is part of the FRQ corruption scenario, where the ground truth labels are *match* if the name is rare and *non-match* if it is a frequently occurring name. This labeling clearly does not apply to all pair patterns of this kind in the real world. Therefore, it is reasonable to study this linkage problem based on the record pair similarities and match probabilities, as is done below in Section 4.4.3.

The experiments using record-level Bloom filter encodings show that the attribute weights have a strong impact on the linkage quality in most scenarios. This is not specific to *privacy-preserving* record linkage, but it highlights the need to study how these parameters can be optimized in the PPRL context. Perturbation-based linkage methods apply weights during the encoding phase, which prevents them from estimating attribute-specific error rates when facilitating the Fellegi–Sunter approach for weight estimation. Compared to the state-of-the-art plaintext linkage results, the PPRL approach performs worse on nearly all linkage tasks, except in the FRQ corruption setting.

3.5 CONCLUSION

We introduced a comprehensive evaluation framework for privacy-preserving record linkage. It offers multiple modules to create versatile benchmark datasets for linkage tasks depending on the available input data. Analyzing linkage quality outcomes based on record (pair) characteristics, or *tags*, enables researchers to better comprehend their linkage results and evaluate those of others. This also aids data custodians in selecting suitable PPRL methods for specific applications and datasets. In future work, we intend to integrate further linkage benchmark datasets in our selection component for more general use.

The framework was employed to generate a set of benchmark linkage problems which are used for an assessment of the linkage quality which is achieved using a common PPRL linkage approach. The comparison of PPRL using record-level Bloom filter encodings with a state-of-the-art baseline for plaintext linkage shows that the linkage quality is worse in almost all scenarios. The created datasets are used throughout this thesis, in particular in the next chapter for studying adaptive weighting schemes.

4

Value-specific attribute weight adaptation

Typically, each record in the datasets to be linked is encoded using identical, fixed parameters to ensure comparability of their encoded representations. In this chapter, we examine methods for dynamically adapting encodings using attribute weights that take value frequencies into account, implying that each record may be encoded in a (slightly) different way. This chapter is based on [122] with an extended evaluation section using the linkage problems introduced in Section 3.3.1.

4.1 MOTIVATION

In plaintext record linkage, records can be compared pairwise based on their attributes, such as first name, last name, and date of birth. The attribute similarities are used to classify pairs as match or non-match. Often weights are involved in this step to take the different discriminatory power and error rates of attributes into account [159]. For example, an equal date of birth is a stronger indicator for a match than an equal gender as there are much more values (and thus each value occurs less often) for date of birth than for gender.

Simple weight-based classification approaches only use attribute-specific weights that are equal for all values of a certain attribute. Thus, the very common last name *Smith* would result in the same weight as the rarer last name *Voigt*. Therefore, the use of value-specific weights based on the frequency of a specific attribute value can increase the linkage quality [159]. For uncertain duplicate candidates, e.g., due to a different address as in the examples in Table 4.1 and Table 4.2, the likelihood of a match is higher if the agreeing attributes – here first and last name – are rare. This is reflected in a higher record similarity score (weighted average) due to increased weights of those attributes.

	First name	Last name	Date of birth	ZIP code	City	Total
Record a	LISA	SMITH	23.09.1973	28451	LELAND	
Record b	LISA	SMITH	23.09.1973	28075	HARRISBURG	
Similarity	1.0	1.0	1.0	0.4	0.0	
Weight	12	13	15	7	7	0.79

Table 4.1: Example of a similarity computation of two records with *common* first and last name.

	First name	Last name	Date of birth	ZIP code	City	Total
Record a	WYNONA	VOIGT	23.09.1973	28451	LELAND	
Record b	WYNONA	VOIGT	23.09.1973	28075	HARRISBURG	
Similarity	1.0	1.0	1.0	0.4	0.0	
Weight	20	25	15	7	7	0.85

Table 4.2: Example of a similarity computation of two records with *rare* first and last name.

In general, Bloom filter based encodings (both attribute-level and record-level) allow for weighting attributes. Attribute-level encodings are very similar to traditional (plaintext) record linkage with regard to weighting. The attribute similarities can be aggregated to a record similarity, for example, by using a weighted average. The only difference effectively is the use of a similarity function that is suited for the encoded data structure. When applying record-level encodings, the data owners can use different parameters per attribute to change the attributes' relative weight in the joint Bloom filter. However, weight adaptation and application in the PPRL context with record-level encodings differ from traditional record linkage as this must be conducted by the data owners.

Specifically, we make the following contributions:

- We study the challenges that arise when applying value-specific weighting in the PPRL context to record-level encodings, e.g., the handling of name variations and missing values during the encoding phase.
- We modify record-level encoding techniques for PPRL to allow for frequency-dependent weight adaptation.
- We thoroughly evaluate these techniques and compare them to existing weighting approaches on attribute-level and record-level encodings. Moreover, we analyze the effects of using limited information on value frequencies as the complete information is considered sensitive in the PPRL context.

The chapter is structured as follows. In the following section, we address Related Work. Then, we discuss weighting-based classification approaches in the PPRL context (Section 4.3) and present an extensive comparative evaluation of the different approaches using a real-world dataset (Section 4.4). Finally, we conclude our work in Section 4.5.

4.2 RELATED WORK

Fellegi and Sunter introduced a probabilistic linkage framework as described in Section 2.1.2. It incorporates the likelihood of attribute agreements between unrelated records which is influenced by their frequencies. Herzog et al. [63] propose a method to adjust match and non-match weights based on the frequency of individual attribute values. Consequently, an attribute-specific and a value-specific weight is used. The authors provide a detailed discussion about the calculation of these weights. Similarly, Zhu et al. [163] propose a scaling factor that is applied directly to the attribute weights of the Fellegi-Sunter approach. The scaling factor is derived from the current dataset itself, without relying on any external source of (name) frequencies.

Attribute weighting has been used in the PPRL domain as well. The record-level encoding methods CLK-RBF [147] and RBF [32] have been proposed to integrate weights, as described in Section 2.3.3. The record linkage and pseudonymization service Mainzelliste, which supports Bloom filter based matching, only uses the agreement weights to combine attribute similarity scores to a record similarity using the weighted average [25, 123]. In [13], weights are estimated based on partial agreement models for each individual attribute of a sensitive dataset. However, this approach can only be utilized for attribute-level encodings. Ranbaduge et al. proposed decay weights for record-level encodings based on time distances [112].

Value-specific weighting approaches, however, have received limited attention so far in PPRL. Giersiepen et al. apply the Fellegi-Sunter approach with frequency-dependent u -probabilities to encrypted attribute-level hashes [47]. This approach is the standard procedure used by German cancer registries. To the best of our knowledge, no prior work has studied value-specific weight adaptation based on individual value frequencies for record-level encodings so far.

4.3 METHODS

We describe methods to adapt the weights depending on the attribute values and their frequencies. Furthermore, we outline methods for estimating weights and discuss the limitations that emerge when these methods are applied to the PPRL setting using record-level encodings.

4.3.1 FREQUENCY-DEPENDENT WEIGHT ADAPTATION

In this section we discuss methods to determine value-specific weights that reflect the relative significance of the respective attribute value based on its frequency.

Value frequencies can be incorporated in the Fellegi-Sunter approach by computing a value-dependent u -probability with $u_i = f_i/T$, where f_i is the absolute frequency of value i and T is the total number of values. For example, suppose an attribute can take three values, 'A', 'B', and 'C', which occur in the dataset with absolute frequencies 100, 50, and 2, respectively. In that case, we obtain $u_A = 100/152 \approx 0.66$ and $u_C = 2/152 \approx 0.01$. The likelihood that two random records agree on this attribute is much larger for the most frequent value 'A' than for the rarest value 'C'. Hence, the weights w_m and w_u are 0.45 and -1.77 for value 'A' and 6.10 and -3.30 for value 'C' (see Equation (2.4)), assuming a constant $m = 0.9$.

Another approach to modify attribute-level weights is a value-dependent scaling factor S , so that $w' = S \cdot w$. This approach is independent of the method used to determine default attribute weights. Furthermore, it is applicable to other parameters such as the number of hash functions h in CLK-RBF encodings. We therefore focus on this weight adaptation method. Zhu et al. [163] proposed a scaling factor defined as

$$S_{\text{Zhu},i} = \sqrt{\frac{T}{Q \cdot f_i}} \quad (4.1)$$

where T is the total number of values, Q is the number of unique values and f_i is the absolute frequency of the value i . For values that are more common than the average, S_{Zhu} is in $[0, 1)$ and for rare values the factor is larger than 1. In our previous example we would compute $S_A = \sqrt{152/(3 \cdot 100)} \approx 0.71$ and $S_C = \sqrt{152/(3 \cdot 2)} \approx 5.03$. Zhu's scaling factor has, however, two unfavorable properties: (1) The reference for $S = 1$ is fixed to the mean frequency. Since value frequency distributions, e.g., of names, typically have few very common values and many rare values, this reference can be quite low leading to $S < 1$ for many mid-common values. (2) The values of the scale factor are biased towards the lower bound. As a consequence, the scale factors are low for values that are not very common. To address these issues, we propose an alternative scaling factor based on the inverse document frequency (idf) which is defined as $\text{idf} = \log_2(T/f_i)$. To achieve $S = 1$ for a desired reference frequency f_{ref} , we define the scaling factor as

$$S_{\text{idf}} = 1 + \text{idf}(f_i) - \text{idf}(f_{\text{ref}}) \quad (4.2)$$

Moreover, we define f_{ref} as the frequency of the median attribute value, which is the value in the middle of the ordered list of values with repetition according to the respective frequency. For example, given a distribution $[\{A, 4\}, \{B, 3\}, \{C, 1\}, \{D, 1\}, \{E, 1\}]$ the (lower) middle of the list $[A, A, A, A, B, B, B, C, D, E]$ is position 5 or value B . This results in $S_{\text{idf}} < 1$ for values that are more frequent than 3. For S_{Zhu} the reference (mean) frequency is $(4 + 3 + 1 + 1 + 1)/5 = 2$. The median-based approach results in half of the values having a scale factor of below 1 and half above 1.

In practice, the scaling factor S_{Zhu} can be unreasonably low or high. For instance, in one of the datasets used in our evaluation, we have $T = 200\,000$ records with $Q = 20\,060$ unique first names. For the most common name 'James' ($f_i = 3401$ (1.7%)) we get $S_{\text{Zhu}} = 0.05$ and for the rarest names with $f_i = 1$ we get $S_{\text{Zhu}} = 3.2$. This very large weight reduction for the name 'James' would result in an almost complete disregard of this attribute in the classification which is not desirable. S_{idf} can even be negative for common values which makes normalization inevitable. Therefore, we normalize and restrict the scales as follows: (1) The scaling factor is normalized to the interval $[0, 2]$. We use a separate min-max normalization for scaling factors below and above 1 to ensure that this value is not modified. (2) S is restricted to a more narrow interval $[S_{\min}, S_{\max}]$, e.g., $[0.75, 1.5]$, to constrain the effect of the weight adaptation.

$$S_{\text{lowest}} = \min(S_i) \quad (4.3)$$

$$S_{\text{highest}} = \max(S_i) \quad (4.4)$$

$$S_{\text{norm},i} = \begin{cases} \frac{S_i - S_{\text{lowest}}}{1 - S_{\text{lowest}}} & \text{for } S_i < 1 \\ 1 + \frac{S_i - 1}{S_{\text{highest}} - 1} & \text{for } S_i \geq 1 \end{cases} \quad (4.5)$$

$$S_{\text{restricted},i} = \begin{cases} S_{\min} + S_{\text{norm},i} \cdot (1 - S_{\min}) & \text{for } S_{\text{norm},i} < 1 \\ 1 + (S_{\text{norm},i} - 1) \cdot (S_{\max} - 1) & \text{for } S_{\text{norm},i} \geq 1 \end{cases} \quad (4.6)$$

The weight adaptation method that relies on a scaling factor involves only a small number of straightforward computations, as illustrated in Algorithm 1. Therefore, it can be easily integrated into existing frameworks that already support weighted Bloom filter encodings. The scale factor is applied to weights at the linkage unit for ABF (see Algorithm 2) and at the data owner for RBF (see Algorithm 3) whereas in CLK-RBF the scale factor is applied to the number of hash functions h (see Algorithm 4).

ALGORITHM 1: Computation of value-specific scale factors

Input: **AF**: Lookup table for attribute value frequencies

a_i : Attribute value i

Output: $scale_i$: Scale factor for value i

```

1 if  $a_i$  is in AF then
2    $freq_i \leftarrow \text{GetFrequency}(\mathbf{AF}, a_i);$ 
3    $scale_i \leftarrow \text{ComputeScale}(freq_i);$                                 /* Eq. (4.1) (Zhu) or Eq. (4.2) (idf) */
4    $scale_i \leftarrow \text{MinMaxNormalize}(scale_i);$                         /* Equation (4.5) */
5    $scale_i \leftarrow \text{RescaleToBoundaries}(scale_i);$                     /* Equation (4.6) */
6 else
7    $scale_i \leftarrow 1.0;$ 

```

ALGORITHM 2: Linkage with attribute-level similarities and value-specific weighting

Input: $\mathbf{R}$: Dataset with attribute-level encoded records
 $\mathbf{w}_{\text{default}}$: Default attribute-specific weights
 t : threshold

Output: $\mathbf{M}$: Matching record pairs

```

1 Candidates  $\leftarrow$  GenerateRecordPairsWithStandardBlocking( $\mathbf{R}$ );
2  $\mathbf{M} \leftarrow []$ ;
3 for  $\text{Candidate} \in \mathbf{Candidates}$  do
4    $\mathbf{AP} \leftarrow \text{GenerateAttributePairs}(\text{Candidate})$ ;
5    $\mathbf{AS} \leftarrow \text{ComputeAttributeSimilarities}(\mathbf{AP})$ ;
6    $\mathbf{w} \leftarrow \mathbf{w}_{\text{default}}$ ;
7   for  $\text{sim}_i \in \mathbf{AS}$  do
8     if  $\text{sim}_i = 1$  then                                /* Adapt only for equal attributes */
9        $\text{scale} \leftarrow \text{GetScale}(\text{AP}_i)$ ;                    /* Algorithm 1 */
10       $w_i \leftarrow w_{\text{default},i} \cdot \text{scale}$ ;
11    $RS \leftarrow \text{ComputeWeightedRecordSimilarity}(\mathbf{w}, \mathbf{AS})$ ;      /* Equation (2.1) */
12   if  $RS > t$  then  $\mathbf{M.append}(\text{Candidate})$ ;

```

ALGORITHM 3: Record-level Bloom filter (RBF) [32] with value-specific weighting

Input: $\mathbf{A}$: Attributes of plaintext record r
 $\mathbf{w}_{\text{default}}$: Default attribute-specific weights
 $l_{\text{RBF}}, l_{\text{ABF}}$: Length of the record-level / attribute-level Bloom filter
 h_{ABF} : Attribute-level number of hash functions

Output: E_{RBF} : Encoded Bloom filter record

```

1  $\mathbf{B} \leftarrow []$ ;  $\mathbf{w} \leftarrow \mathbf{w}_{\text{default}}$ ;
2 for  $a_i \in \mathbf{A}$  do
3    $\text{ABF}_i \leftarrow \text{GenerateBloomFilter}(a_i, l_{\text{ABF}}, h_{\text{ABF}}, i)$ ;
4    $\text{scale} \leftarrow \text{GetScale}(a_i)$ ;                                /* Algorithm 1 */
5    $w_i \leftarrow w_{\text{default},i} \cdot \text{scale}$ ;
6  $\text{share} \leftarrow \text{ComputeProportionalNumbersOfBits}(\mathbf{w}, l_{\text{RBF}})$ ;
7 for  $\text{ABF}_i \in \mathbf{ABF}$  do
8    $\mathbf{B.append}(\text{SampleBits}(\text{ABF}_i, \text{share}_i))$ 
9  $E_{\text{RBF}} \leftarrow \text{Permute}(\mathbf{B})$ ;

```

ALGORITHM 4: CLK-RBF encoding [147] with value-specific weighting

Input: $\mathbf{A}$: Attributes of plaintext record r
 h_{default} : Default attribute-specific number of hash functions
 l : Length of the record-level Bloom filter

Output: $E_{\text{CLK-RBF}}$: Encoded Bloom filter record

```

1  $E_{\text{CLK-RBF}} = \text{InitializeEmptyBloomFilter}(l)$ ;
2 for  $a_i \in \mathbf{A}$  do
3    $\text{scale} \leftarrow \text{GetScale}(a_i)$ ;                                /* Algorithm 1 */
4    $h_i \leftarrow h_{\text{default},i} \cdot \text{scale}$ ;
5    $E_i \leftarrow \text{GenerateBloomFilter}(a_i, l, h_i)$ ;
6    $E_{\text{CLK-RBF}} \leftarrow E_{\text{CLK-RBF}} \cup E_i$ ;

```

4.3.2 WEIGHT ESTIMATION

In this section we describe how weights can be estimated and the issues that arise when applying these methods in the PPRL domain. As described in Section 2.1.2, a popular method to determine weights is based on the probabilistic approach of Fellegi and Sunter. The computation of the weights requires estimates of the m - and u -probabilities for the attributes. Given ground truth data, we can calculate $m = 1 - e$ where e is the error rate, i.e., the share of true duplicates with a different value for that attribute. In practical applications, the error rate must either be determined using expert and domain knowledge or determined beforehand in a preliminary study.

A naive approach to estimate attribute-specific u -probabilities is $u = 1/\#\text{uniqueValues}$, which means that the probability that two attribute values agree by chance is equal to the average relative frequency. We use a different approach which is sensitive to the frequency distribution by setting $u = \sum_{i=0}^{Q-1} p_i^2$, where Q is the number of distinct values and p_i the relative frequency of the i -th value. As an example, consider the attribute gender that can take the categorical values 'female', 'male' and 'other'. The first two values are nearly equally frequent ($p = 0.48$), the last value, however, is less common ($p = 0.04$). The estimated probability that the values of two random records agree is $1/3$ in the naive approach, but 46 % when considering the frequency distribution.

In PPRL methods with record-level encodings, it is not possible to assign weights based on whether attributes match or differ. The reason is that weights have to be set by the data owners and thus at a stage where the outcome of the comparisons is still unknown. Therefore, the two weights of the Fellegi-Sunter model have to be combined into a single weight. Durham et al. [32] proposed the range $w = w_m - w_u$. The combined weight, however, can be dominated by w_m if the attributes have a large variety of values which is typically the case for names. As a consequence, we normalize w_m and w_u with respect to the maximum/minimum value across all attributes.

$$w_{mi,norm} = \frac{w_{mi}}{\max(w_m)} \quad (4.7)$$

$$w_{ui,norm} = \frac{w_{ui}}{\min(w_u)} \quad (4.8)$$

$$w_{i,norm} = \max(w) \cdot (w_{mi,norm} - w_{ui,norm}) \quad (4.9)$$

4.3.3 LIMITED FREQUENCY INFORMATION

The application of this approach in the PPRL domain comes with additional challenges, for attribute-specific weights as well as for frequency-dependent value-specific weights. The estimation of u -probabilities and the computation of weight scale factors are based on frequency distributions which are not readily available in the PPRL context. This information is considered sensitive as it could be used to perform frequency attacks on Bloom filter encodings. In the following, we discuss possible solutions to deal with this limitation.

Frequency distributions can be gathered from an external source on similar datasets, e.g., statistical data from a census of the same geographical region or be computed for the data to be linked. The first approach is especially useful for smaller datasets where the calculated value counts may not represent the real-world frequency distribution well. The latter ensures that the used frequencies correspond to the actual properties of the dataset. Each data owner can determine its own source-specific frequency distribution and compute weights based on it. This will likely result in different weights for identical values. We discuss these effects in Section 4.3.4.

Data owners cannot exchange the complete frequency information as this would leak information on rare values. However, the data owners might be willing and allowed to exchange and combine the relative frequencies of their most common values. While this can increase the linkage quality, it does not affect the privacy as the linkage unit does not learn this information.

For S_{idf} , an alternative reference frequency has to be chosen, since the median is not a reasonable reference point of an incomplete frequency distribution containing only the most common values. We therefore propose to use the least frequent value in the list of most common attribute values as the reference. The scaling factor for this value as well as values not in the list is 1. For values on the list, it is below 1.

Restricting the frequency-dependent weight adjustments to common values will probably result in smaller impacts on the linkage outcome. Additionally, the weight application can be restricted to certain attributes, e.g. first and last name, because frequency information on other attributes is missing. However, it still could be beneficial with respect to the precision. We experimentally evaluate this effect in Section 4.4.

4.3.4 EFFECTS OF ATTRIBUTE DIFFERENCES IN DUPLICATES

Real-world data often contains typographical errors. Besides, names can have natural variations, for instance, the German last name 'Schmidt' with its variants '*Schmid*' or '*Schmitt*'. These varying values occur with different frequencies which leads to different frequency-dependent weights.

		First name	Last name	Year of birth
	Record <i>a</i>	LISE	DONOVAN	1956
	Record <i>b</i>	LISA	DONOVAN	1956
Base	n (avg. #features)	7	8	5
	w_{default}	12	11	14
Scale	S_b	0.5	1	1
	S'_b	0.67	1	1
RBF	$w_a (= w_{\text{default}})$	12	11	14
	w_b	6	11	14
	w'_b	8	11	14
	% of sampling for <i>a</i> (w_a)	$12/37 = 32\%$	$11/37 = 30\%$	$14/37 = 38\%$
	% of sampling for <i>b</i> (w_b)	$6/31 = 19\%$	$11/31 = 36\%$	$14/31 = 45\%$
	% of sampling for <i>b</i> (w'_b)	$8/33 = 24\%$	$11/33 = 33\%$	$14/33 = 43\%$
CLK-RBF	h_a (for $S = 1$)	18	15	30
	h_b (for S_b)	9	15	30
	h'_b (for S'_b)	12	15	30

Table 4.3: Example illustrating the effects of a weight variation of single attribute when using record-level encodings ($l = 1024$) based on RBF ($l_{ABF} = 256$) and CLK-RBF ($h = 20$).

The consequences of varying weights depend on the weight application technique that is used.

For the RBF approach a single different attribute weight changes the proportions of the weights and hence the sampling rates for all other attribute-level Bloom filters. The fill rate of the RBF remains nearly the same across different weights settings, provided that the attribute-level Bloom filters have equal fill rates.

Using CLK-RBF with weight adaption, as described in Algorithm 4, the scaling factor of an attribute i is applied by changing the number of hash functions h_i for that attribute only. This does not affect the number of hash functions for the other attributes. Nevertheless, the fill rate of the final Bloom filter is changed as the total number of hash functions is modified.

In Table 4.3, we illustrate the effect of changing a single weight using the two encoding methods. Based on the default weights and the average number of features per attribute, we compute the sampling rates for RBF and the number of hash functions h for the CLK-RBF approach for an example record *a*, that has $S = 1$ for all attributes. Record *b* has the same last name and year of birth, but a different and very common first name. Therefore, we set the scaling factor $S_b(\text{FN}) = 0.5$. The sampling rates for all attributes of record *b* are changed due to the decreased sum of all weights. For CLK-RBF, however, only $h_b(\text{FN})$ is adapted. As a consequence, the generated Bloom filter encodings based on the RBF method are more affected by weight variations compared to the CLK-RBF approach. The same applies if we compute a different scaling factor $S'_b(\text{FN}) = 0.67$

based on a different frequency information, e.g., when using source-specific frequency distributions (as described in the previous section) where in each distribution the name '*Lisa*' occurs often, but with different relative frequencies. If two sources encode an identical record b based on their respective scaling factors S_b and S'_b , the resulting Bloom filters are different. In contrast, using the CLK-RBF approach, this affects only a few hash functions and thus the difference between the Bloom filters is lower.

To avoid different weights for attribute variations, such as '*Lisa*' and '*Lise*', we consider the use of frequency distributions based on generalizations of the plaintext values, e.g., using the Soundex phonetic encoding function [95]. Consequently, the weight for each value is computed based on the frequency of its generalized value (Soundex code). For example, the Soundex code for both '*Lisa*' and '*Lise*' is 'L200', and thus, the same weights are computed, although the original values have different frequencies.

4.3.5 HANDLING MISSING VALUES

Apart from erroneous attributes, missing values often occur in real-world datasets and a strategy is needed to handle them. When working with plaintext or attribute-level encodings, the linkage unit can detect missing values. Multiple strategies can be used, e.g., ignoring the attribute in the similarity score aggregation or setting its similarity score to 0. When working with record-level encodings the linkage unit cannot detect missing values. If a data owner detects a missing value during the encoding phase the respective weight could be redistributed to the other attributes. However, as the missingness is source-specific, a true match with that value set would be encoded with different weights for all attributes. This would in turn lead to differences in the resulting Bloom filters and thus likely to misclassifications. We therefore do not adapt weights of missing values and simply treat them as empty attributes.

4.4 EVALUATION

We evaluate the methods described in the previous section with respect to the linkage quality, while focusing on the following aspects: (1) Quantification of the effects of frequency-dependent weight adaptation. (2) Comparison of weight application approaches in PPRL. (3) Study of the effects if only limited information on frequency distributions is available. (4) Application of the weight adaption scheme on a variety of linkage problems to study the robustness.

Name	A	B	A ∩ B	Attributes
F	100k	100k	20k	FN, MN, LN, YOB, CITY, ZIP, POB
R	100k	100k	20k	FN, LN, YOB, CITY, ZIP

Table 4.4: Description of used datasets for evaluation of weight adaptation and application methods.

4.4.1 SETUP

DATASETS To study the effects on real-world data, we use NC-min1E-200M created by the D_S component of the evaluation framework as described in Section 3.3.1.

We define a set of person-related attributes named **F** (*‘Full’*) which are taken into account in the linkage: First name (FN), middle name (MN), last name (LN), year of birth (YOB), city, ZIP code and place of birth (POB). Based on **F** (*‘Full’*) we derive another dataset, named **R** (*‘Reduced’*), where we removed the attributes middle name and place of birth, thus, making the dataset more challenging to match, see also Table 4.4.

For the experiments with external statistical information we use frequencies of first and last names from the 1990 US Census.¹

LINKAGE PARAMETERS We apply the same encoding and blocking strategy as described in Section 3.3.3. The dataset properties and derived weights and encoding parameters for **F** and **R** are shown in Table 4.5. We set a fixed length of $l = 256$ for the attribute-level Bloom filter. RBF encodings are based on the same ABF parameters. We set the length of the record-level Bloom filter to $l = 1024$. For the computation of attribute-specific h_i in the CLK-RBF encoding we use the reference number of hash function $h = 12$ (**F**) and $h = 15$ (**R**) which results in an average Bloom filter fill rate of approximately 40 %.

Attr.	Avail.	$\varnothing l$	Properties			ABF h	RBF		CLK-RBF	
			m-prob	u-prob	w_{norm}		%(F)	%(R)	h(F)	h(R)
FN	99.99 %	6	0.9300	0.0027	12.83	18	19.95	24.07	15	18
MN	92.16 %	5.1	0.4380	0.0037	7.43	21	11.55	–	12	–
LN	100 %	6.4	0.7187	0.0010	11.14	17	17.32	20.90	11	15
YOB	100 %	4	0.9900	0.0135	14.44	26	22.45	27.09	24	29
CITY	99.97 %	8.9	0.6507	0.0193	6.63	13	10.31	12.44	6	7
ZIP	99.89 %	5	0.5318	0.0031	8.27	21	12.86	15.51	11	14
POB	79.13 %	2	0.6436	0.1513	3.57	43	5.55	–	10	–

Table 4.5: Attribute properties (availability, average length $\varnothing l$, m - and u -probability, normalized weight) and derived encoding parameters for Attribute-level Bloom filter (number of hash functions h), Record-Level Bloom filter (attribute share) and CLK-RBF (number of hash functions h).

We conduct additional experiments where a post-processing routine on the set of matches is applied, using a symmetric best match strategy (Max1-both) to restrict the

¹https://www.census.gov/topics/population/genealogy/data/1990_census.html

result to 1:1 links. In many practical use cases this is reasonable when the sources can be considered *clean*, or duplicate-free, and therefore each record of a database has at most one duplicate in the other database [43].

EVALUATION MEASURES We use the standard measures recall, precision and F1 score to evaluate linkage quality. We evaluate these measures for similarity thresholds t in the range of $[0.7, 1.0]$ in steps of 0.01. In practical record linkage, however, ground truth data is not available and the used thresholds are rarely optimal. For a high linkage quality in real-world applications the results should be stable for a broader range of thresholds. We therefore use a loss measure L_M^d for the linkage quality that describes the maximal loss of measure M in the threshold range $[t_{\text{opt}} - d, t_{\text{opt}} + d]$. For example, $L_{F1}^{0.03}$ describes the maximal loss of the F1 score within a distance of 0.03 to the optimal threshold. Furthermore, we report the area under the curve (AUC) of precision-over-recall as a threshold-independent measure.

4.4.2 RESULTS

The threshold-dependent quality measures are reported for the classification thresholds t_{opt} that are optimal with respect to the F1 score for this linkage setup and dataset.

WEIGHT ADAPTION First, we evaluate different weight adaptation methods based on the scaling factors S_{Zhu} and S_{idf} (see Table 4.6). We use ABF encodings as described in Algorithm 2 and the full frequency distribution computed for the respective datasets and test multiple scale factor intervals.

All weight adaption configurations improve the linkage quality compared to static weights. However, the S_{idf} -based approaches generally show a higher rise of the AUC than the S_{Zhu} -based with a maximum improvement by 0.043 (**R**) and 0.022 (**F**) each with a constraint interval of $[0.5, 2.0]$. The optimal F1 scores show similar increases by 0.048 (**R**) and 0.028 (**F**). Therefore, we employ this weight adaptation strategy in all subsequent experiments.

WEIGHT APPLICATION We compare the results of the value-specific weight adaptation strategy for the encoding techniques ABF, CLK-RBF and RBF (see Table 4.7). Two baselines are included, with and without attribute-specific weights. The latter is implemented by using the arithmetic mean of the attribute similarity scores (ABF), equal number of hash functions h for all attributes (CLK-RBF) and by sampling equal shares from each attribute (RBF). The value-specific weighting scheme achieves AUC improvements for CLK-RBF comparable to those of the attribute-level application despite the missing restriction of weight adjustments to equal attributes: +0.038 (**R**) and +0.021 (**F**) with respect to

DS	S	S _{min}	S _{max}	AUC	t _{opt}	Rec.	Pre.	F1	L _{F1} ^{0.01}	L _{F1} ^{0.03}	L _{F1} ^{0.05}
R	–	–	–	0.841	0.85	0.786	0.787	0.787	0.029	0.089	0.180
	Zhu	0.5	2.0	0.869	0.82	0.813	0.813	0.813	0.032	0.094	0.189
		0.5	1.5	0.867	0.82	0.810	0.812	0.811	0.031	0.093	0.188
		0.75	1.5	0.856	0.84	0.786	0.817	0.801	0.022	0.080	0.160
	idf	0.5	2.0	0.884	0.84	0.830	0.839	0.835	0.030	0.091	0.184
		0.5	1.5	0.877	0.84	0.817	0.839	0.828	0.024	0.083	0.174
		0.75	1.5	0.866	0.85	0.795	0.837	0.815	0.019	0.087	0.156
F	–	–	–	0.918	0.83	0.809	0.914	0.859	0.008	0.050	0.151
	Zhu	0.5	2.0	0.933	0.79	0.853	0.904	0.878	0.019	0.080	0.210
		0.5	1.5	0.932	0.79	0.849	0.904	0.875	0.017	0.078	0.208
		0.75	1.5	0.927	0.81	0.834	0.906	0.868	0.014	0.066	0.185
	idf	0.5	2.0	0.940	0.81	0.863	0.913	0.887	0.017	0.072	0.191
		0.5	1.5	0.937	0.81	0.851	0.916	0.882	0.014	0.064	0.180
		0.75	1.5	0.931	0.82	0.838	0.914	0.875	0.012	0.058	0.167

Table 4.6: Linkage quality comparison of weight adaptation methods on Attribute-level Bloom filter.

the attribute-specific weight baseline and +0.126 (**R**) and +0.093 (**F**) to the averaging baseline. However, with the sampling-based approach (RBF) AUC decreases for value-specific weighting. As we discussed in Section 4.3.4, this is because even a single different weight, e.g., due to a typo, leads to considerably dissimilar Bloom filters. Even with a low threshold of 0.76 the recall is as low as 0.373 (for **R**). Thus, we subsequently focus on the CLK-RBF encoding.

In general, we observe that weight adaptation methods lead to lowered optimal thresholds with increases in recall as well as in precision. While the first is expected when lowering the threshold, the rise of precision suggests that non-match candidates with comparatively high similarity due to common values are less often wrongly classified as matches as these attributes are weighted lower. Moreover, we note that the improved results are also equally or more stable regarding the threshold selection. For **R** with an increase of the F1 score by 0.051, L_{F1}^d is reduced from 0.028 to 0.015 in $d = 0.01$ and decreases by 0.023 in $d = 0.03$, which indicates that a higher linkage quality can be achieved in real-world applications with non-optimal threshold selection (see also Figure 4.1).

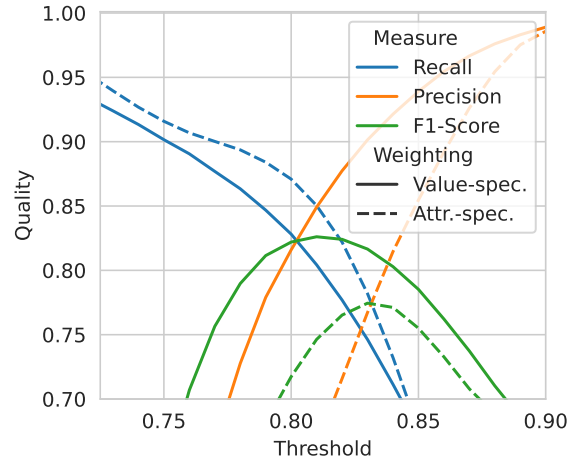


Figure 4.1: Comparison of quality measures for attribute- and value-specific weighting on CLK-RBF for **R**.

DS	Enc.	Weighting	AUC	t_{opt}	Rec.	Pre.	F1	$L_{F1}^{0.01}$	$L_{F1}^{0.03}$	$L_{F1}^{0.05}$
R	ABF	Averaging	0.777	0.86	0.609	0.803	0.693	0.015	0.045	0.052
		Attribute-specific	0.841	0.85	0.786	0.787	0.787	0.029	0.089	0.180
		Value-specific	0.884	0.84	0.830	0.839	0.835	0.030	0.091	0.184
	CLK-RBF	Averaging	0.749	0.85	0.580	0.806	0.674	0.014	0.047	0.078
		Attribute-specific	0.837	0.83	0.782	0.767	0.775	0.028	0.093	0.221
		Value-specific	0.875	0.81	0.804	0.849	0.826	0.015	0.070	0.189
	RBF	Averaging	0.787	0.86	0.605	0.828	0.699	0.007	0.024	0.043
		Attribute-specific	0.845	0.85	0.770	0.806	0.788	0.015	0.074	0.179
		Value-specific	0.675	0.76	0.373	0.973	0.539	0.001	0.001	0.001
F	ABF	Averaging	0.900	0.80	0.808	0.857	0.832	0.009	0.046	0.134
		Attribute-specific	0.918	0.83	0.809	0.914	0.859	0.008	0.050	0.151
		Value-specific	0.940	0.81	0.863	0.913	0.887	0.017	0.072	0.191
	CLK-RBF	Averaging	0.845	0.77	0.714	0.804	0.756	0.009	0.034	0.067
		Attribute-specific	0.917	0.80	0.824	0.903	0.861	0.019	0.079	0.192
		Value-specific	0.938	0.77	0.866	0.917	0.891	0.019	0.072	0.185
	RBF	Averaging	0.874	0.77	0.793	0.815	0.804	0.020	0.120	0.291
		Attribute-specific	0.920	0.81	0.834	0.905	0.868	0.024	0.099	0.248
		Value-specific	0.801	0.77	0.677	0.958	0.793	0.005	0.023	0.054

Table 4.7: Comparison of linkage quality results when using averaging, attribute-specific and value-specific weighting ($S_{\text{idf}} [0.5,2]$) for different weight application methods.

POSTPROCESSING The restriction of the linkage result to 1:1 links is reasonable in scenarios with *clean* sources. In order to study whether weight adaption further improves the results, we evaluate the weighting methods for CLK-RBF where the links have been post-processed before the linkage quality assessment (see Table 4.8). The results show increases of AUC, +0.026 (**R**) and +0.011 (**F**), indicating that the weight adjustment technique is beneficial under these conditions as well.

DS	PP	Weighting	AUC	t _{opt}	Rec.	Pre.	F1	L _{F1} ^{0.01}	L _{F1} ^{0.03}	L _{F1} ^{0.05}
R	yes	Attribute-specific	0.867	0.81	0.833	0.799	0.816	0.016	0.058	0.117
		Value-specific	0.893	0.80	0.820	0.876	0.847	0.007	0.038	0.093
F		Attribute-specific	0.936	0.79	0.848	0.915	0.880	0.007	0.032	0.077
		Value-specific	0.947	0.76	0.881	0.927	0.904	0.012	0.045	0.101

Table 4.8: Linage quality comparison of attribute-specific and value-specific weighting ($S_{\text{idf}} [0.5,2]$) for CLK-RBF where the found matches have been restricted to 1:1 links in postprocessing (PP).

LIMITED FREQUENCY INFORMATION Finally, we study how variations of available frequency information affect the results (see Table 4.9). Again, we report two baselines, with and without value-specific weight adaption, to allow for a comparison with the current state-of-the-art of attribute-specific weights and with value-specific weights under ideal conditions. The results with frequency distributions based on Soundex encodings instead of plaintext values show lower linkage quality, because weights of rare values can be decreased in this setting if these values share the encoding with a common value.

DS	Weighting	AUC	t_{opt}	Rec.	Pre.	F1	$L_{F1}^{0.01}$	$L_{F1}^{0.03}$	$L_{F1}^{0.05}$
R	Attribute-specific	0.837	0.83	0.782	0.767	0.775	0.028	0.093	0.221
	Value-specific	0.875	0.81	0.804	0.849	0.826	0.015	0.070	0.189
	⊢ Soundex-based	0.858	0.82	0.788	0.832	0.809	0.016	0.075	0.207
	⊢ Source-specific	0.875	0.81	0.791	0.864	0.826	0.009	0.058	0.171
	⊢ Top 10	0.858	0.83	0.789	0.836	0.811	0.014	0.077	0.218
	⊢ Top 20	0.864	0.82	0.825	0.813	0.819	0.032	0.117	0.286
	⊢ Top 100 Names (Census)	0.851	0.82	0.802	0.769	0.785	0.025	0.082	0.195
F	Attribute-specific	0.917	0.80	0.824	0.903	0.861	0.019	0.079	0.192
	Value-specific	0.938	0.77	0.866	0.917	0.891	0.019	0.072	0.185
	⊢ Soundex-based	0.930	0.78	0.852	0.915	0.882	0.016	0.073	0.197
	⊢ Source-specific	0.938	0.77	0.858	0.924	0.890	0.013	0.062	0.168
	⊢ Top 10	0.928	0.79	0.845	0.913	0.877	0.018	0.080	0.212
	⊢ Top 20	0.931	0.79	0.844	0.924	0.882	0.013	0.065	0.183
	⊢ Top 100 Names (Census)	0.929	0.79	0.841	0.906	0.872	0.016	0.067	0.168

Table 4.9: Linkage quality comparison of weighting methods with different limitations of the available frequency information (CLK-RBF).

When using source-specific frequency distributions, the results are almost identical to those obtained with the global frequency data, because the distributions are very similar. When linking smaller datasets, the distributions will have larger differences, in particular for rare values. We therefore conduct additional experiments where we limit the available frequency information to the most frequent values, as described in Section 4.3.3. The results show that even with a limitation on the 20 most frequent values AUC increases by 0.027 (**R**) and 0.014 (**F**) compared to attribute-specific weights. However, when using external statistical data on the 100 most frequent first and last names only, the linkage quality improvements are comparatively low. The inclusion of information on the frequencies of additional (geographical) attributes could potentially improve the results.

Generally, we see that the quality improvements for **F** are lower than for **R** because the inclusion of information on value frequencies is more relevant in linkage scenarios where fewer attributes are available.

4.4.3 ROBUSTNESS

The following experiments have been conducted in addition to those presented in [122].

We evaluate the effect of the value-specific weight adaption on CLK-RBF encodings with $S_{\text{idf}} [0.5, 2]$ on further datasets, whose properties are described in Section 3.3.1. The same 112 linkage problems are included as in the weight study in Section 3.3.4. The attribute set used for linkage is identical to that employed in **F**.

Figure 4.2 shows the differences of the AUC and F1 score metrics when the weight adaption scheme is used on base weights estimated using the Fellegi-Sunter approach as before. The weight adjustment leads to performance gains for all subgroup-based linkage problems, but causes declines in the T and D scenarios. It performs best for NC-Yob00

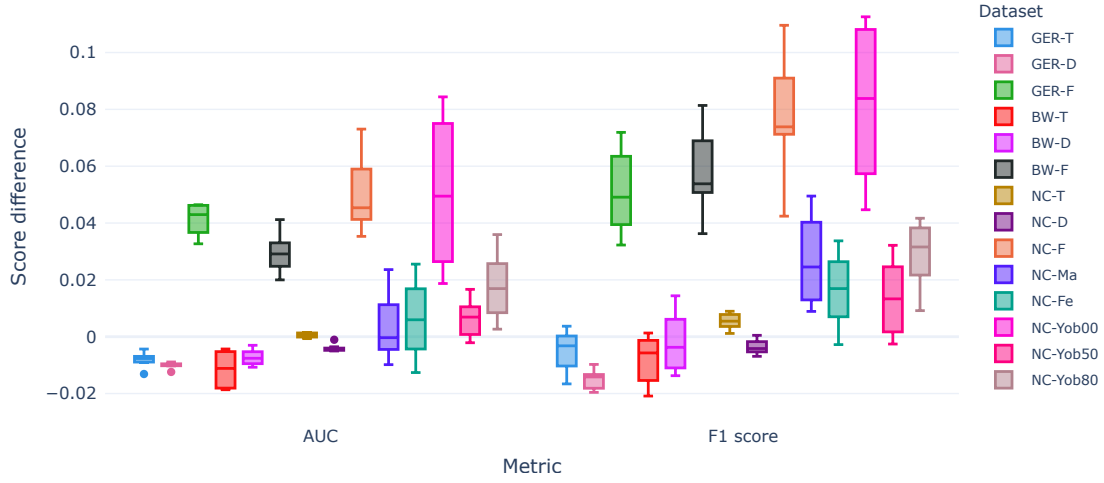


Figure 4.2: Linkage quality metric differences of value-specific weighting ($S_{idf} [0.5,2]$, CLK-RBF) compared to attribute-specific weights estimated using the Fellegi-Sunter approach.

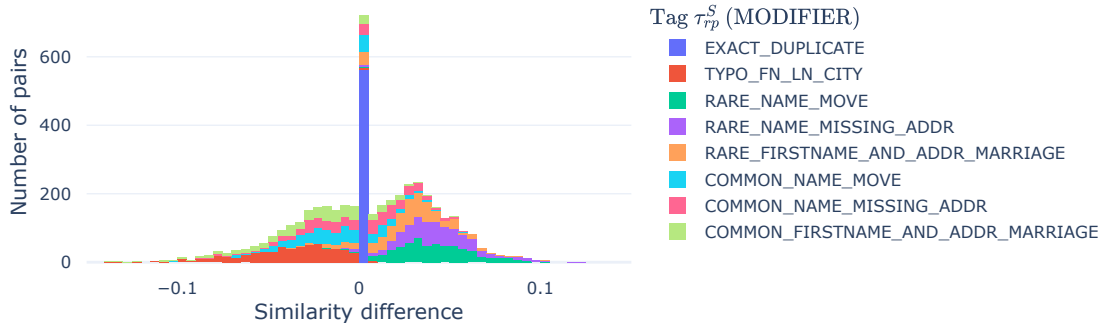


Figure 4.3: Histogram of similarity shifts of record pairs due to value-specific weight adaptation on NC-F-20-L with estimated weights using the Fellegi-Sunter approach. The pairs are grouped by their respective modification tag from the corruption module D_C .

(AUC +0.049, F1 score +0.084). This is due to the fact that it contains values for first name and year of birth which are very frequent (see Figure 3.9) and those attributes are largely error-free in this cohort (see Figure 3.11). The first aspect causes lowered attribute weights for these values and thereby improved precision. The latter helps maintain high recall because the weight scaling factor is not affected by variations in the values. In contrast, T and D are often affected by value discrepancies and therefore by the negative consequences described in Section 4.3.4. The corruption scenario F leads to high median improvements of AUC between +0.029 to +0.049 and of the F1 score between +0.049 to +0.074. This is expected as it explicitly assigns pair labels in the ground truth based on the frequency of attribute values.

Figure 4.3 shows the similarity changes of corresponding record pairs in NC-F-20-L due to the weight adaptation based on the similarity analysis module A_S (see Section 3.2.4). Record pairs that contain rare values in general have positive shifts, while the similarities

of pairs with more common values tend to become lower. As a result, the groups can be more easily distinguished using a threshold-based classifier and are less sensitive to the specific threshold selected. The employed method for estimating default, attribute-specific weights based on the Fellegi-Sunter framework depends on error rates for individual attributes. These are difficult to determine in the PPRL setting when using record-level encodings, since comparisons at the attribute level cannot be performed. Therefore, we study the application of the weight adaption scheme on static weight scenarios, as defined in Table 3.3. In addition, source-specific frequency distributions are used.

Figure 4.4 presents the differences of the AUC and F1 score metrics for the corruption modifications of the base datasets GER, BW and NC respectively. The FRQ corruption scenario again shows a strong impact of the value-specific weight adaptation with median AUC improvements of +0.015 to +0.028 for *Equal*, +0.013 to +0.03 for *Core* and +0.02 for *Core+*. In the other two settings, DIRTY and TIME, the linkage quality declines when the weights are adapted, similar to the previous results. This is particularly true for TIME and *Core* and *Core+* weight sets. This scenario comprises a large proportion of duplicates with changed last name values (see also Figure 3.11). The relatively high base weight assigned to this attribute in these weight sets, together with varying scaling factors caused by inconsistent frequency distributions, results in greater dissimilarity of the constructed Bloom filter encodings.

Figure 4.5 shows the similarity score shifts in NC-F-20-L, comparable to Figure 4.3. In general, rare values again lead to positive shifts and more common values tend to decrease the similarity. In addition to this intended effect however, it can be seen that the perfectly agreeing pairs as well as the typo-afflicted pairs only exhibit negative shifts. The reason for this is that the source-specific similarity distributions differ, resulting in deviating weights being assigned to identical attribute values. This does not necessarily result in false negative predictions, since the pairs still achieve high similarity scores that exceed the classification threshold which is typically within $[0.7, 0.8]$. Still, this similarity shift is unwanted behavior for exact duplicates.

Figure 4.6 depicts the results for the subgroup datasets based on NC. In general, the weight adaptation scheme achieves the highest metric improvements for the *Equal* base weights. This is reasonable, as these configurations perform considerably worse in their static weight application (see Figure 3.15) and therefore have a greater potential for improvement. The other weight sets likewise indicate enhanced quality and robustness for Ma, Yob80, and especially Yob00, whereas for Fe and Yob50 the performance is slightly and more substantially impaired, respectively. Yob00 has mostly identical values for the attributes first name and last name, as well as a larger proportion of agreeing residence data. In contrast, Fe has a changed last name as the most common duplicate difference pattern. This indicates that value-specific weight adaption is beneficial when applied on attributes which exhibit a rather low error-rate.

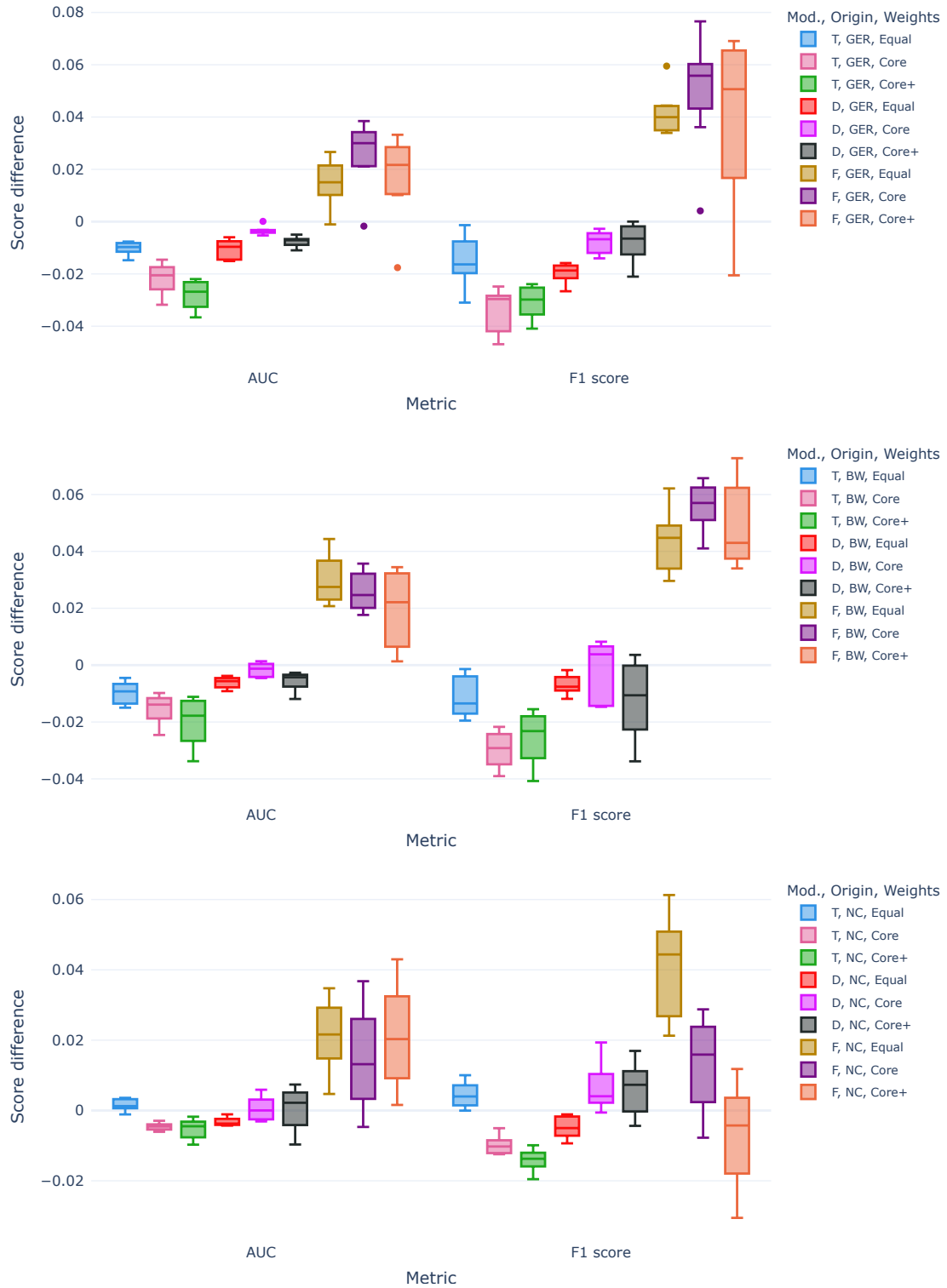


Figure 4.4: Linkage quality metric differences of value-specific weighting compared to attribute-specific weights for corruption-based linkage problems from GER (top), BW (middle) and NC (bottom).

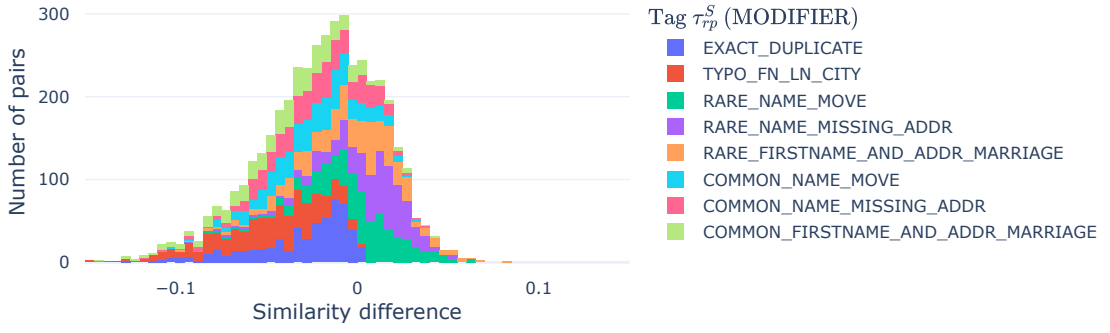


Figure 4.5: Histogram of similarity shifts of corresponding record pairs due to value-specific weight adaptation on NC-F-20-L with the Core+ weight set and source-specific frequency distributions. The pairs are grouped by their respective modification tag from the corruption module D_C .

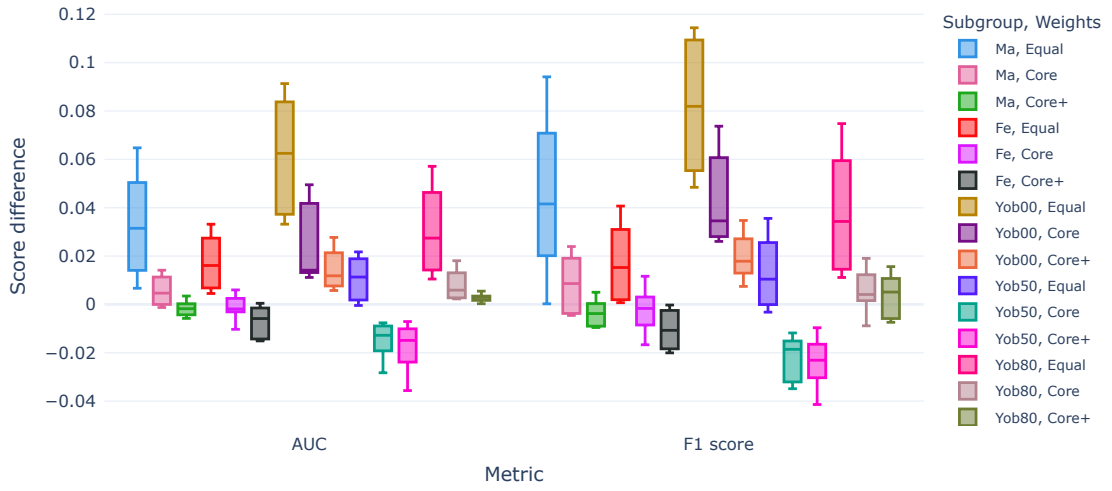


Figure 4.6: Linkage quality metric differences of value-specific weighting compared to attribute-specific weights for the subgroup linkage problems from North Carolina (NC).

Overall, the results show that the effects of value-specific weight adaptation not only depend on the characteristics of the linkage problem but also largely on the selection of the base attribute weights which are modified using the scaling factor. Furthermore, the results demonstrate that while the weight adaptation scheme improves the linkage quality and robustness in some scenarios it leads to quality impairments for others.

4.5 CONCLUSION

A common element of many record linkage algorithms is the classification of record pairs as match or non-match based on a weighted aggregation of attribute similarities from these records. Methods for applying attribute- and value-specific weights to plaintext data have been extensively researched. In contrast, only few works adapt these methods in the context of PPRL.

In this chapter, we apply existing record-level Bloom filter encodings and combine them with frequency-dependent weight adaptation approaches. We extensively evaluate the adaptive encoding schemes and compare them with attribute- and record-level Bloom filter encodings. The results show that the modified CLK-RBF encoding outperforms the existing (record-level) methods and achieves comparable results to attribute-level weight application techniques regarding linkage quality and robustness. However, the latter require attribute-level encodings, which are susceptible to cryptanalysis and thus may not be considered for use in practical applications.

Nevertheless, weighting schemes are still limited for record-level encodings, in particular since weight adaption cannot be restricted to agreeing attributes as the similarity is not known at the time of the weight application during encoding.

5

Multi-layer protocol with clerical review based on gradual information disclosure

Perturbation-based PPRL extracts a single feature from record-level comparison of pairs and employs threshold-based classification methods that assume linear separability. Predictions for match candidates close to the decision boundary are especially uncertain. In general, PPRL methods do not support the inspection of those links, which would allow to resolve the ambiguity, assess the quality of the results and refine the matching strategy. In this chapter, which is based on [119, 121], we propose a PPRL protocol that integrates clerical review with minimal information disclosure.

5.1 MOTIVATION

Selection and configuration of record linkage algorithms typically require training data with information on known matches and non-matches. In practical linkage applications there is generally no such ground truth data available though [20]. Linkage on plaintext data can reliably achieve high linkage quality by manually reviewing (uncertain) classifications. Such partial ground truth from clerical review also allows to evaluate and adapt the chosen linkage algorithm. When linking sensitive data, however, a clerical review on plaintext data is usually not feasible. There is limited work investigating a privacy-preserving clerical review (PPCR) system for record linkage where attribute values are gradually disclosed and displayed using (visual) masks [73, 75, 109]. However, these masks are applied for display only and the reviewing institution still receives full plaintext data. Moreover, the approach does not aim at reducing the labeling effort or improving an automatic classification model based on labeled samples.

We therefore propose a protocol for PPRL that uses active learning to achieve high-qualitative and reliable linkage results with a low labeling effort. Our linkage protocol employs perturbation-based encodings and uses multiple layers to gradually disclose limited information only if needed. First, the linkage is conducted on record-level encodings where all identifying attributes are combined in a single encoding for each record. These encodings are more secure but do not permit sophisticated classification approaches. Therefore, an active learning process is initiated where uncertain match candidates are iteratively resolved by (re-)classification using attribute-level encodings or ultimately by a masked clerical review. In contrast to a linkage solely based on attribute-level encodings the resistance against reidentification attacks is greatly improved by using pair-specific keys for those encodings and thereby avoiding exploitable frequent bit patterns.

In particular, we make the following contributions:

- We present a novel multi-layer active learning protocol that combines automatic privacy-preserving record linkage and manual masked clerical review while minimizing the amount of shared sensitive data.
- We analyze the implications of our protocol with regard to reidentification attacks.
- We conduct experiments on real-world data to evaluate the performance of our protocol in terms of labeling effort, linkage quality improvement and privacy risk.

5.2 BACKGROUND AND RELATED WORK

In the following, we describe the technical background from related work and derive requirements for our protocol. We focus on the most popular PPRL encoding technique based on Bloom filters. However, other perturbation approaches (see Section 2.3.3) are applicable as well. As outlined in the background Section 2.3, attribute-level encodings are especially vulnerable to reidentification attacks when a sufficient number of them is available to compute meaningful frequency distributions. Thus, we derive the first requirement for our protocol:

Requirement R1. *Attribute-level encodings must not be used with the same parameters for all records to mitigate the risk of successful frequency attacks.*

Instead, multiple or all attributes are combined in a single encoded representation. Typically, the linkage unit computes a single similarity score for each pair of encoded records and classifies it based on a threshold. The selection of an appropriate threshold is therefore essential for a high linkage quality. Moreover, record-level encodings conceal whether a certain difference originates from one or multiple attributes. Table 5.1 shows an example where the linkage unit cannot differ between two non-matching records (A1

	ID	First	Last	Birth Date	City
Plain	A1	PAUL	SMITH	1976/09/07	RALEIGH
	B2	PAUL	SMITH	1974/06/07	RALEIGH
	A3	PETER	COHEN	1976/09/07	LELAND
	B3	PETER	COHEN	1976/09/07	RALEIGH
Rec.-level	A1				
	B2			0.82	
	A3				
	B3			0.82	
Attr.-level	A1				
	B2	0.8	1.0 (freq.)	0.7	1.0
	A3				
	B3	1.0	1.0 (rare)	1.0	0.2
Masked	A1	****A		***\$/*@/*	
	B2	****	✓(freq.)	***%/*\$/*	✓
	A3				
	B3	✓	✓(rare)	✓	✗

Table 5.1: Quality issue of record-level encodings: Matching and non-matching pairs might have the same record similarity score, when they have either multiple slightly different attributes (in A1 and B2) or a single replaced attribute (A3 and B3). Attribute-level similarities and masked clerical review with limited information disclosure enable better classification. Please note that the full plaintext records are only known to their respective data owners (A and B) and are displayed here in subsequent rows for better comparability by the reader.

and B2) that have similar but different first names and dates of birth, and two matching records (A3 and B3) where these attributes are equal but the city is very different, e.g., because the person moved. A threshold-based classifier that should classify the second pair as a match will therefore misclassify the first pair (and vice versa) due to the equal overall similarity score. A classifier with access to attribute-level similarities would be able to distinguish these cases.

In [55], a PPRL use case is presented in which uncertain match candidates from record-level comparisons are examined using plaintext attributes, illustrating that practical applications demand high linkage quality by integrating a clerical review.

PRIVACY-PRESERVING CLERICAL REVIEW (PPCR)

Tuning linkage parameters, such as the threshold, requires the availability of (partial) ground truth data which is generally not available in practical linkage projects. Ground truth labels can be determined in a clerical review process where potential matches are decided upon manually. During clerical review, record pairs and potentially additional information are presented to an oracle, typically a human. The display of such information in plaintext obviously does not preserve privacy and is therefore not applicable to sensitive data. In [162], a protocol is proposed where ground truth labels are obtained by blindly evaluating boolean questions, formulated as match conditions, under homomorphic

encryption. Moreover, systems with a masked display were proposed for manual clerical review that conceal the plaintext by default, present categorical value frequencies, and gradually disclose selected information [73–75, 109]. The studies showed that the masking had only little impact on the error rate of the labeling using the incremental disclosure approach in [75] and moderate impact depending on the level of disclosure in [109]. Table 5.1 (bottom) shows an example of such a masked display. Attributes that are identical or very dissimilar are replaced by respective (dis)agreement symbols. Attributes with a medium similarity are displayed partially either by showing the differing plaintext characters (here: first name) or placeholders (here: birth date). Although the data is shown only partially disclosed, the responsible institution for the clerical review has access to the full plaintext records in the backend services. They are used to determine appropriate masks depending on the attribute similarities. However, to enable such a privacy-preserving clerical review (PPCR), the system requires merely selected plaintext based on the information whether an attribute pair is equal, dissimilar, or somewhat similar. Only in the latter case, the attribute values are needed to determine replaced or swapped characters (groups). Based on these observations, we derive the following requirements for our protocol:

Requirement R2. *The protocol must determine attribute-level similarities.*

Requirement R3. *The institution responsible for the (masked) clerical review should only have access to those plaintext attributes that are displayed (partially).*

The risk to conduct a successful reidentification attack on an encoded dataset increases when more information is disclosed (see below in Section 5.3.2). While some data owners might be willing and allowed to provide more information to improve the linkage quality, others might not. From this assumption, we derive another requirement:

Requirement R4. *Data owners must have control over the amount of information they share for each record.*

For example, the provision could be limited by a restrictive or missing consent of some persons. In such a linkage scenario with heterogeneous consents, it would be beneficial to use additional information from records with less strict consent to improve the classification performance for all other records.

ACTIVE LEARNING

We aim to use the partial ground truth from clerical review to improve the classification performance for non-reviewed record pairs. Our protocol is similar to active learning approaches that strive to minimize manual labeling, which in the PPRL context is based on sensitive data. In pool-based active learning, samples are selected from a set of unlabeled instances using a query strategy and labeled by an oracle. The labeled samples are then

used to train a classification model. This process is repeated until an exit condition is met, e.g., by reaching a budget of allowed queries. The most important component is the query strategy. A variety of techniques was proposed [99]. Some, such as heuristic-based methods using the feature vectors of unlabeled instances, are not suitable for record-level encodings, as only a single similarity score feature is available. Margin-based strategies are applicable and select the most uncertain instances, typically close to the decision boundary, i.e., the threshold.

While many studies on active learning assume the (human) oracle to be flawless [104], this is not a reasonable model for our protocol due to the restricted access to the data for enhanced privacy. A lack of handling label noise could lead to poor models when used for training. Crowd-based approaches could be used where the output label is determined collectively, e.g., by a majority vote [14]. However, in our privacy-sensitive setting, additional queries and oracles would increase the privacy risk. We therefore consider only a single human oracle.

Requirement R5. *The model update process must not expect the oracle labels to be error-free as its predictions are based on limited information.*

5.3 METHODS

In the following, we present our multi-layer active learning PPRL protocol based on the requirements listed above. Our protocol follows the need-to-know principle [126]: The protocol is comprised of multiple layers of classification with increasing levels of disclosure. The data owners share only as much information for each record as needed for a classification decision with high probability. While the majority of pairs are classified with high certainty in the top layer, some pairs require additional information. The underlying assumption is that the accuracy of the classification benefits from such disclosure.

Without loss of generality, we focus on a protocol with three layers (see Figure 5.1). First, in L_R , encodings with a higher focus on privacy protection are used to determine certain matches and certain non-matches, e.g., record-level encodings E_R . Batches of uncertain match candidates are selected and pushed to *wishlists* for review by the next layer L_A . To determine attribute similarities (R2) attribute-level encodings E_A are necessary. A pair-specific secret key is used to ensure that each record pair is encoded differently (R1). These keyed attribute-level encodings serve two purposes: (i) Classify uncertain record pairs with a higher probability for an unambiguous match decision than with record-level encodings and thus ideally make further clerical review unnecessary and (ii) Determine attributes that have an intermediate similarity score and should be visually masked in clerical review (R3). The masked clerical review is the last layer L_C

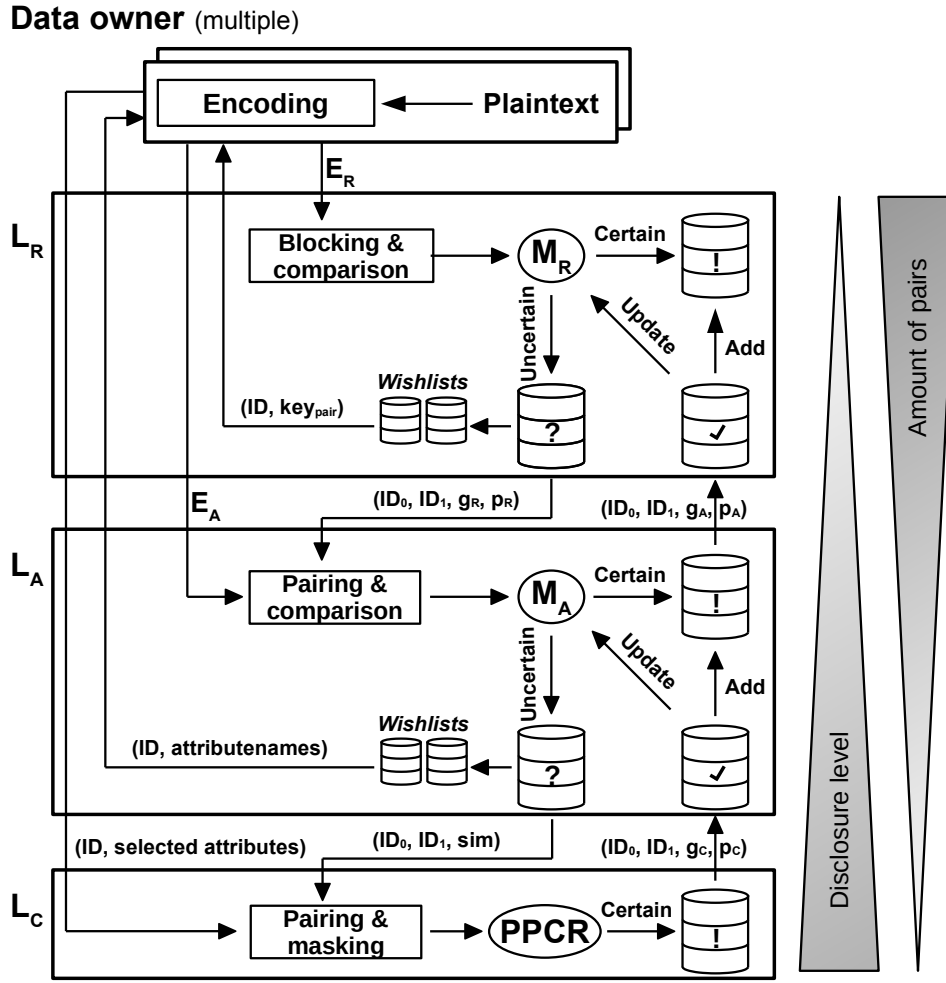


Figure 5.1: PPRL protocol with three linkage layers: Initially, data owners provide record-level encodings (R). Uncertain pairs are reviewed using keyed attribute-level encodings (A) and finally with masked disclosure of selected attributes only (C).

to resolve remaining uncertain match candidates. For each batch, the revised labels are reported to the upper layer and used to update the model. Non-revised pairs are reclassified with the updated model. Predictions can change in subsequent iterations and are propagated again upwards in order to hamper learning from erroneous labels (R5). After each batch, the first layer holds the best possible result that is achieved based on the given information. A protocol with i layers uses $i - 1$ active learning processes. Our three-layered protocol consists of AL_R and AL_A . In AL_R , the classifier M_A of layer L_A is used as an oracle to label instances and train the classification model M_R from layer L_R . In AL_A , a human oracle provides the labels for layer L_A and its model M_A .

The protocol requires data owners to actively provide additional information through their encoding component (R4). They retrieve batches of requests for further information on selected records from the linkage unit. However, the data owners may limit the level of disclosure independently for each record. If no additional information is provided at all by the data owners to layers L_A and L_C , the protocol is equivalent to the common

single-provision setting of a PPRL process utilizing a record-level encoding. If no plaintext attributes are provided, no labeled instances are available for training the attribute-level model. Using a pre-trained model (trained on a sufficiently similar dataset) the protocol could be run in a basic configuration with a single active learning process based on the upper two layers.

5.3.1 COMPONENTS

Encoding Without loss of generality, we focus on Bloom filter encodings, as described in Section 2.3.3. For these encodings the parameters $\mathbf{P}$ comprise at least the length of the Bloom filter m , the q-gram length q and the set of h hash functions $\mathbf{H}$. We use a pseudo random number generator (PRNG) to generate h values in the range $[0, m - 1]$. The PRNG is initialized with a seed s that is constructed by a keyed cryptographic hash function $s = \text{HMAC}_{\text{SHA256}}(f, \text{key}_{do})$, where f is the feature (q-gram) to be hashed and key_{do} is a secret key as part of the $\mathbf{P}$ known only to the data owners.

For the first layer any error-tolerant record-level encoding technique can be used, e.g., record-level Bloom filters as proposed by Durham et al. [32] or Cryptographic Longterm Keys (CLK) as proposed by Schnell et al. [128]. Error tolerance means that the encoded entities must be comparable with approximate similarity functions. For the second layer attribute-level encodings are used, e.g., keyed attribute-level Bloom filter (KABF). For each attribute a_i a secret key is constructed as $\text{key}_i = \text{concat}(\text{key}_{do}, \text{key}_{pair}, \text{an}_i)$ where key_{do} is the secret key known to the data owners only, key_{pair} is a pair-specific key and an_i is the name of the attribute. key_{do} is necessary to prevent dictionary attacks by the linkage unit. key_{pair} is a random key generated by the linkage unit individually for each uncertain record pair. It ensures that these specific encoded records are created using the same hash functions and thereby are comparable. The inclusion of the attribute name an is called *attribute salting* and leads to different hash functions for each attribute of a record [46]. Thus, the same q-grams from different attributes are hashed to different bit positions which hampers frequency analysis. The masked clerical review in the third layer requires selected plaintext attributes without any obfuscation. Visual masking depends on the paired records and therefore has to be applied at L_C .

Blocking, comparison and classification For each layer a separate linkage strategy is used. First, pairs of encoded records are generated. We use standard blocking to reduce the number of pairs in the first layer. For lower layers, no further blocking is required as the match candidates have been identified already. For each pair the corresponding encoded parts in $\mathbf{E}$ are compared which results in a similarity vector $\mathbf{sim}$. We use the Dice coefficient to compare Bloom filters and compute normalized similarity scores in the range $[0, 1]$ [29]. The classification model is a function $(g, p) = \text{classify}(\mathbf{sim})$, where g is the binary classification target ($\{\text{Non-match}, \text{Match}\}$) and p is the probability of

that target $([0.5, 1])$. We denote the classification model of layer L_R as M_R (Record-level) and the model of L_A as M_A (Attribute-level). Arbitrary classification methods utilizing similarities are applicable. In the first layer, a simple threshold-based classification model is used. For record-level Bloom filter additional features, e.g., based on the bit vector fill rate, could be computed. Some hardening techniques distort these features though, so for the sake of generalizability we use only the minimal available feature. In the second layer, we use an evolving Random Forest model as suggested by [104]. The classifier is updated gradually by adding and replacing trees as explained below.

Query strategy The aim of the query strategy in Active Learning is the selection of samples that are most important for the training. We consider uncertainty sampling which selects those samples where the probability of the model prediction is low. For a binary threshold-based classification these are typically the record pairs whose similarity is close to the threshold value. However, a sampling strategy considering only the minimal distance to the threshold potentially results in homogeneous samples for training. Therefore, we use a bucket-based strategy to select pairs with varying similarities. We divide the samples where $p < p_t$ into x buckets of equal width $(p_t - 0.5)/x$, where p_t is the probability threshold. In multiple iterations, a random sample is selected from each of the buckets ordered by the lower bound of that bin. For each selected pair we generate a pair-specific secret key_{pair} and push both records to a queue of oracle requests (*wishlist*). According to R4 the data owners retrieve a batch of their respective wishlist and may or may not provide the requested record representations to the next layer. The batch size depends on the overall size of the datasets as well as on the expected response rate of the data owners. A larger batch size is required to gather a sufficient number of pairs if the response rate of the data owners is low (as the data owners' responses are independent of each other). In AL_A , we again apply this query strategy, based on the probabilities p that are calculated by M_A .

Oracle The oracle assigns a (preliminary) ground truth label to a record pair. In AL_R , the attribute-based M_A is used as the oracle based on attribute similarities, whereas in AL_A a human assigns the label based on the visually masked display. Both oracles have limited information and therefore may assign wrong labels with a non-negligible error rate. We denote the error rate of the masked clerical review as err . As the oracle in the intermediate layer L_A evolves with more updates from the lower layer, the oracle may revise its prediction as explained in the next Section.

Update and back-propagation Predictions of the oracle are reported from that lower layer and used to update the model. The revised model is then used to reclassify all non-reviewed instances. After that, all instances with changed outcomes (g, p) are reported again to the next upper layer. The update is based on all labeled instances (also from previous iterations). This is because instances may be reported multiple times from the lower layer if the prediction has been revised by the reclassification in the lower layer.

The classifier for record-level encodings M_R uses a single threshold t . We implement a straightforward threshold optimization algorithm as follows : The reported labeled pairs with their similarity are classified using various thresholds. The threshold for which the quality measure q is optimized is selected as the new threshold. We consider only thresholds within a maximal distance dt to the initial threshold because we presume an approximately suitable default value. We also restrict the maximal shift per update (dt_{step}) in order to prevent selecting thresholds where few labeled samples are available yet. For attribute-level encodings initially a Random Forest model is bootstrapped based on the predictions from M_R . On update, another small temporary Random Forest model is trained based on all labeled instances. The trees are added to the larger Random Forest model. If the number of trees exceeds a limit, the oldest trees are removed. Thereby, the model gradually adapts to the current set of labeled instances and previous potentially erroneous samples are forgotten. In both layers, the instances are weighted based on their probabilities p . The weights of instances labeled by L_C are doubled as they are assumed to be more reliable.

5.3.2 PRIVACY ANALYSIS

Perturbation-based PPRL protocols are commonly based on the Honest-but-curious adversary model [84]. It is assumed that each party follows the protocol but tries to learn as much as possible about the other parties based on the data it receives. Moreover, the linkage unit is assumed to not collude with any data owner. Otherwise a data owner could share the encoding function including its parameters, such a key_{do} , with the linkage unit that could transform various possible records r , e.g., from a public source, and thereby conduct a dictionary attack to assign individuals to (encoded) records.

Attribute-level BF have been shown to be vulnerable to frequency attacks [21, 23]. These attacks determine frequencies of BF encodings and align these with plaintext value frequencies (see step 4 and 5 in Figure 5.2). A sufficient number of records with the same encoding is required for gathering meaningful frequency information. We therefore use compound secret keys where the linkage unit provides distinct secret shares for each record pair as described in Section 5.3.1. Thus, at most two records are encoded the same way and frequencies cannot be determined for the complete dataset. Nevertheless, attacks based on similarity graphs [151] are still possible in principle. For each attribute a corresponding similarity graph could be constructed. However, as only a subset of the full graph of the first layer is compared on the attribute-level, it is rather unlikely that an attacker could construct a sufficiently similar graph based on plaintext values.

Additional attack scenarios arise in our protocol when multiple linkage layers are conducted by the same linkage unit. In the following, we analyze possible attacks by combining different (encoded) representations of the same records from multiple layers.

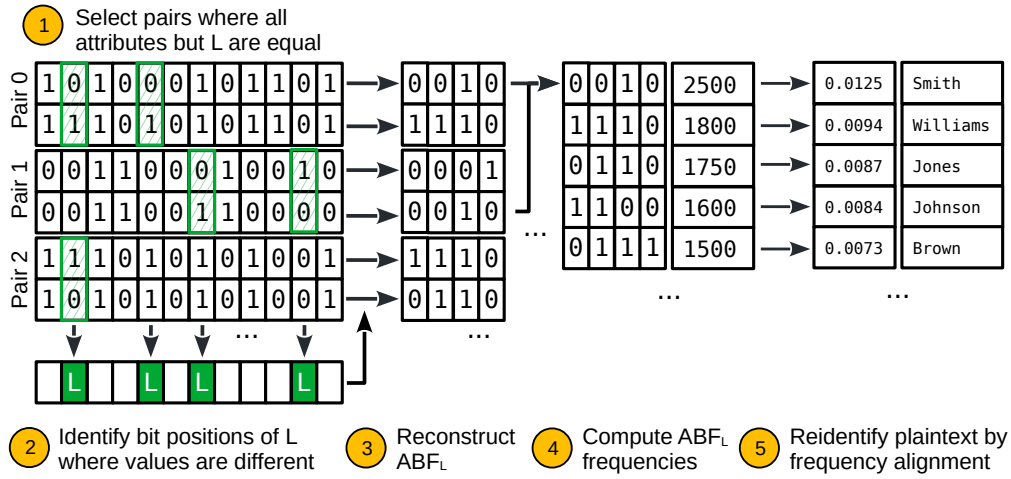


Figure 5.2: Example of a reidentification attack for an attribute $L(\text{astname})$ based on RBF encodings and known attribute similarities

Attack record-level encodings with KABF Durham et al. proposed an encoding method where first attribute-level BF are generated from which bits are sampled according to the respective attribute weights to construct a record-level BF (RBF) [32]. In Figure 5.2, we outline an attack on those RBFs when attribute-level similarities are known. That would be the case if a single organization is responsible for L_R and L_A . This attack is possible because each bit position in the record-level Bloom filter corresponds to exactly one attribute. The bit positions of a certain attribute can be identified using a set of pairs where all but this attribute are equal.

In CLK, tokens from different attributes are hashed directly into a joint Bloom filter and therefore can be mapped to the same bit positions [128]. These collisions have the effect that bit positions cannot be assigned unambiguously to attributes. Thus, the attack on RBF, as described above, is not applicable, in particular if CLK with hardening techniques are used, e.g., with record-specific salting or xor-folding [46].

Attack KABF with plaintext An attacker with access to a plaintext attribute pair (from L_C) and a KABF pair (from L_A) can presume which bit positions correspond to which token, similar to the attack on RBFs described above. However, as distinct hashing secrets are used for each attribute as well as for each record pair, the attacker cannot infer any information about other encoded records.

Attack record-level encodings with plaintext The most hazardous scenario arises when an attacker has access to plaintext attributes and a record-level encoding of the same record(s) which would be the case when a single (malicious) organization is conducting L_R and L_C . Similar to the attacks with KABF above, the attacker may infer correspondences of bit positions/patterns and use this information to attack all other record-level encodings.

To prevent those scenarios, the first layer should be conducted by a different independent organization than the other layers. It receives only the classification outputs of pairs from lower layers.

Membership inference Furthermore, the query strategy may leak information to the data owners and allow them to infer the membership of some of their records in other databases. This could already reveal sensitive information if the other linkage participants and some common characteristics of the records in their databases are known. For example, the knowledge that there is a duplicate of a known person in a cancer registry leaks private information. However, the data owners do not learn the classification outcome of their records and thereby cannot tell whether a re-encoding request corresponds to a certain match, a certain non-match or an uncertain match candidate.

5.4 EVALUATION

We assess our proposed protocol in terms of improvements in linkage quality under varying levels of labeling effort, as well as its associated privacy implications.

Quality The proposed protocol aims to enhance the overall linkage quality by using labeled instances from clerical review. The improvement is achieved in two ways: (i) The original uncertain labels are replaced by those from lower layers, which are likely more accurate due to additional available information. (ii) The labeled samples are also used to update the classification models and improve the labels of non-reviewed record pairs by reclassifying them.

To evaluate linkage quality we use the F1 score. The quality assessment is repeated after each batch of reviewed pairs with subsequent post-update reclassification in the first layer.

Privacy risk We focus on quantifying the privacy risk of lower layers as the top record-level layer is not directly affected by our protocol with regard to feasible attacks. There is no universal privacy measure for perturbation-based PPRL as the risk depends on the considered attack types and background knowledge as outlined above. For Bloom filter based encodings several privacy risk scores have been proposed [46]. We report the Gini coefficient (G) as well as the Jensen-Shannon divergence (JSD) for measuring the dissimilarity of the bit frequency distribution with a uniform distribution. The notion of these measures is that the risk of frequency attacks is reduced if all bit positions have the same likelihood of being set to '1'. The scores of both measures range from 0 (identical, low privacy risk) to 1 (maximal different, highest privacy risk). As the chances of successful attacks rise in general with more accessible information, we also consider the number of available pairs/records in L_A and L_C .

For the lowest layer with masked clerical review, we report the share of attributes that have been provided (on request) by the data owners as some attributes are more relevant for the privacy risk than others. However, reidentifications are mostly feasible using attribute combinations as this may allow to unambiguously map partial records with (uncommon) values to their original representation. We therefore also report the k -Anonymized Privacy Risk (KAPR) score which has been proposed for measuring the privacy risk based on the revealed information in masked display [80]. The normalized risk score in $[0, 1]$ is higher, the more plaintext of the records is disclosed and the lower the number of records that are indistinguishable based on the level of disclosure. For each record i the number of possible records (k_i) is determined based on the available information. The overall KAPR score is computed using an adapted function from [80]: $KAPR = \frac{1}{ND} \sum_{i=0}^{2n-1} \frac{d_i}{k_i}$ where N is the total number of records (in this layer), D is the total number of attributes and d_i is the number of provided attributes of record i in this layer. The original formula includes the proportion of disclosed characters, but our KAPR variant measures the risk based on the data available to the reviewing institution instead of based on the information that is displayed.

5.4.1 SETUP

Datasets For evaluation of the proposed protocol we use real-world personal records from the North Carolina voter register (NCVR). We derive multiple datasets with two sources of each $50k$ records, where the source overlap and level of disparity between duplicates varies, as described in Section 3.3.1. These dimensions of dataset variation – overlap of data sources and disparity of duplicates – have been chosen as they represent dataset characteristics that typically are not known prior to linkage. We use the attributes *first name* (FN), *middle name* (MN), *last name* (LN), *year of birth* (YOB), *CITY*, *ZIP code* and *place of birth* (POB) for linkage. In Table 5.2, the used datasets in this chapter and their corresponding name in Section 3.3.1 are summarized. The datasets names are composed of their error rate (E1, E2) and overlap (S, M, L).

Encoding We use CLK encodings with $l = 1024$ and $h = 12$ for the first layer (E_R).

Dataset	Error rate	Overlap	Corresponding dataset name in Section 3.3.1
E1S	min. 1 attr. / match	0.1	NC-min1E-100S
E1M	min. 1 attr. / match	0.2	NC-min1E-100M
E1L	min. 1 attr. / match	0.3	NC-min1E-100L
E2S	min. 2 attr. / match	0.1	NC-min2E-100S
E2M	min. 2 attr. / match	0.2	NC-min2E-100M

Table 5.2: Datasets based on the North Carolina voter registry with different duplicate disparity and varying overlap between sources are used in the evaluation of the multi-layer PPRL protocol.

Additionally, we test an encoding variant where the XOR-folding hardening technique is applied to these CLK. As attribute-level encodings E_A we use KABF with $l = 256$ and attribute-specific h (see Table 5.3), following the encoding procedure described in 4.4.1. Both, CLK and KABF, have an approximate average fill rate (proportion of 1-bits) of 40%. For L_A and L_C , data owners also provide approximate frequency information for the attributes, together with the attribute-level encodings.

Attribute	FN	MN	LN	YOB	CITY	ZIP	POB
h	18	21	17	26	13	21	43

Table 5.3: Number of hash functions h for the Keyed Attribute-level Bloom filter encodings in the L_A .

The possible values are: '1' for the top 1% most frequent values, '2' for the top 5%, and '3' for rarer values. Those frequency labels are used as additional features for M_A when the corresponding attributes of a pair are equal, otherwise the value is set to '0'.

Linkage In L_R , the blocking strategy based on hashed plaintext attribute combinations is used, as described in Section 3.3.3. This procedure ensures that the same candidate pairs are generated during the initial batch matching on the first layer even with different encoding techniques.

Query strategy and models We select the parameters $p_t = 0.8$ and $x = 10$ for our uncertainty-bucket-based query strategy. The prediction probability p of M_R is computed for a similarity sim and the threshold t as $p = 0.5 \cdot (1 + \min(1, \text{abs}(sim - t)/d))$ with $d = 0.05$ for $sim < t$ and $d = 0.1$ for $sim \geq t$. For M_R , we apply the threshold shift approach (see Section 5.3.1) with accuracy as the quality measure. Although this measure is typically not suitable for unbalanced classification tasks like record linkage, it is applicable in this context as the number of matches and non-matches close to the threshold is way less imbalanced than in the complete linkage result. For M_A , we use the shifting Random Forest classifier based on the RF implementation from the WEKA library [39] using the options "-P 70 -I 10 -J 10 -N 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1 -depth 6" where -I, -J, -N determine the initial, added and maximum number of trees. The model is bootstrapped using the initial batch of pre-labeled pairs from L_R . In each iteration the model is updated with all samples and their respective predictions, either from L_R or L_C .

Test protocol Initially, a batch matching on the first layer is conducted. Afterwards, our iterative protocol is composed of two phases: In the warm-up-phase, 5 smaller batches of 100 uncertain record pairs from the first layer are selected for review by the second layer. After classification based on attribute-level encodings with M_A , 2 batches of uncertain pairs on that layer are reviewed by the final oracle (PPCR). After each reviewed batch, the model is updated based on the revised predictions and all non-reviewed record

pairs are reclassified based on the revised model. The back propagation to the first layer is conducted in each iteration of AL_R after completion of the 2 batches of AL_A . The batch size in AL_A is computed as $\frac{b}{10}$. After this initial phase, the budget is reached and the clerical review layer is omitted. Consequently, M_A is not updated anymore. The batch size is increased to 1000 pairs per iteration for 4 additional batches to study the performance of the trained M_A . The number of reviews for the top layer can still rise as the limiting budget b refers to the masked clerical review.

Evaluation procedure We evaluate different scenarios with regard to the PPCR layer: We vary the simulated error rate $err = \{0.0, 0.1, 0.2\}$ and the clerical review budget $b = \{100, 200, 300\}$. For each dataset, we determine the top-level threshold t_{opt} that optimizes the F1 score given the global ground truth. Each scenario is evaluated using initial thresholds between $t_{opt} - 0.05$ and $t_{opt} + 0.05$ with a stepsize of 0.01 to study the performance of the protocol with regard to threshold optimization. We repeat each experiment three times and report the micro average F1 score as well as the minimum-maximum range.

Baselines In the following figures, grey-dotted vertical lines mark the end of the warm-up phase and green dashed horizontal lines represent the F1 score with optimal threshold for the respective dataset without any revised labels as a baseline. Furthermore, orange-dotted horizontal lines refer to the optimal F1 scores achieved using a linkage strategy solely based on attribute-level encodings. Attribute similarities are aggregated to a record similarity using a weighted mean average. Weights are determined based on a probabilistic approach [36], using the default average value frequencies and error rates of the Epilink Matcher based on a German cancer registry [123]. For missing attributes, the corresponding similarity scores are excluded from the aggregation.

Attribute	FN	MN	LN	YOB	CITY	ZIP	POB
w	12.04	15.15	5.12	6.58	8.23	10.95	6.63

Table 5.4: Attribute weights **w** used for the baseline linkage solely based on (non-keyed) attribute-level Bloom filters.

5.4.2 RESULTS

Dataset E1M Figure 5.3 (left) shows the results for dataset E1M with $b = 100$ and varying err . The worst initial F1 score is 0.791 for the threshold $t_{opt} + 0.05$. The F1 score for t_{opt} is 0.863 and the average initial F1 score over all thresholds is 0.838. As expected, the linkage quality improves with growing number of reviewed pairs. After reviewing 4500 pairs on the top layer, average F1 scores reach 0.899 ($err = 0.0$), 0.888 ($err = 0.1$) and 0.881 ($err = 0.2$). The final results of the best runs are very similar with a

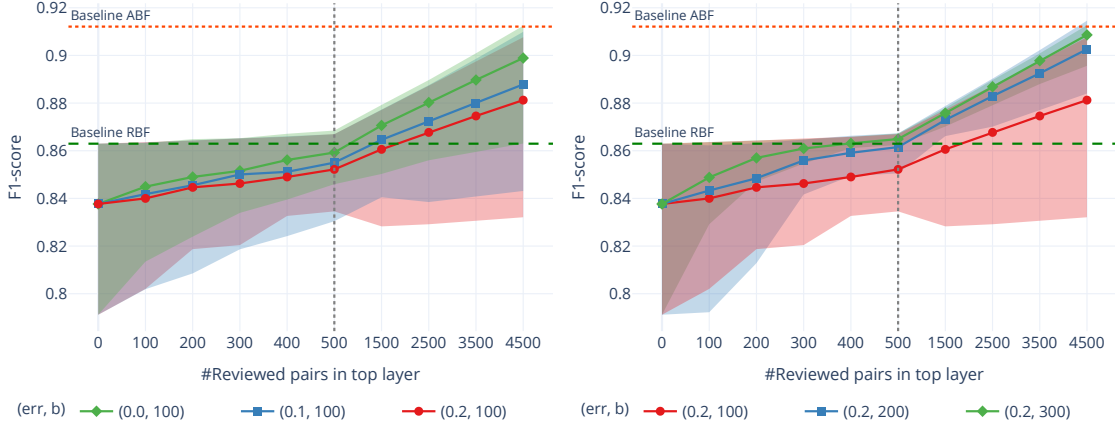


Figure 5.3: Linkage quality development for dataset E1M with fixed budget $b = 100$ and varying error rates (left) and varying budget b and fixed error rate of $err = 0.2$ (right).

F1 score of 0.910 ± 0.002 . The worst runs lead to F1 scores between 0.832 ($err = 0.2$) and 0.863 ($err = 0.0$). Hence, higher error rates of the PPCR oracle lead to higher outcome variability.

In the following, we set the error rate err to 0.2 which is in line with the empirical studies based on visual disclosure in [75, 109]. As depicted in Figure 5.3 (right), higher budgets $b = 200$ and $b = 300$ improve the performance considerably. The final average F1 scores are $0.903 (+0.065)$ and $0.909 (+0.071)$ which is comparable to the best runs with lower budget.

Higher budgets also improve the reliability of the results, as even the worst runs achieve F1 scores of 0.884 and 0.896 . The range of outcomes is reduced from 0.072 to 0.031 ($b = 200$) and 0.017 ($b = 300$). Furthermore, F1 scores rise already with a low number of reviews in the top layer. In our setup, the optimal threshold can be reached from the worst starting condition ($t_{opt} \pm 0.05$) after three iterations due to the shift limit $dt_{upd} = 0.02$. For $b = 300$ the average F1 score reaches the baseline in three to four rounds ($90 - 120$ manual reviews) whereas for $b = 100$ it is not achieved at all during the warm-up phase despite a comparable number of manual reviews. Therefore, the quality improvements with larger budgets in that phase are mostly the result of the threshold shift algorithm.

Datasets E1S and E1L Further experiments using dataset variants with lower and higher overlap show enhancements of the average F1 scores from 0.786 by 0.101 (E1S, $b = 300$) and from 0.862 by 0.048 (E1L, $b = 300$), see Figure 5.4 (left). The minimal F1 score is increased from 0.727 by 0.131 (E1S, $b = 300$) and from 0.802 by 0.083 (E1L, $b = 300$). The range of outcomes is reduced from 0.094 to 0.046 (E1S, $b = 300$) and from 0.083 to 0.033 (E1L, $b = 300$). The final results for $b = 200$ are comparable, but the improvements are achieved later and with higher variance in the process.

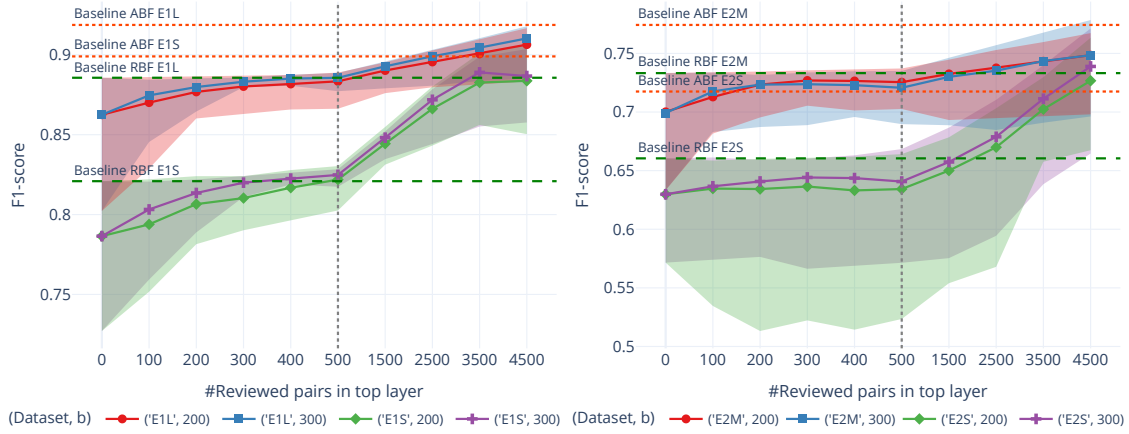


Figure 5.4: Linkage quality development for datasets E1S and E1L (left) and E2S and E2M (right) with varying budget b and fixed error rate of $err = 0.2$.

Figure 5.5 shows the linkage quality development for selected individual runs on dataset E1L with varying budget. In contrast to the other results in this chapter, the F1 scores are measured twice per iteration: after provision of revised labels from lower layers as well as after reclassification of pairs based on the updated model. As the number of reviews is identical for these evaluations, M_R threshold adaptations are visible as comparably large vertical steps of the F1 scores in the warm-up phase of the protocol. The run with the lowest budget of 100 reviews illustrates the detrimental effect of a low quality M_A which results in declining linkage quality when uncertain pairs are reviewed by this model after the warm-up phase.

BF Hardening The experimental outcomes based on E_R with XOR hardening (see Figure 5.6) are comparable in terms of the average F1 scores, which improve by 0.108 (E1S-XOR), 0.076 (E1M-XOR) and 0.049 (E1L-XOR). However, the linkage quality is in general slightly lower as a trade-off for the improved privacy of the encodings, which is illustrated, e.g., for E1M-XOR compared to E1M by decreases of G from 0.224 to 0.095 and JSD from 0.172 to 0.079.

Datasets E2S and E2M Finally, we study the results for datasets E2S and E2M (see Figure 5.4 (right)). In general, the linkage quality is lower due to the higher dissimilarity of duplicates. The average F1 score is raised by 0.109 (E2S, $b = 300$) and 0.048 (E2M, $b = 300$), which is comparable to the E1 datasets. However, the progress is less stable, leading to a larger range of outcomes and for some runs even quality losses (E2S, $b = 200$). For most of the datasets the respective reference results solely using (non-keyed) ABF encodings (orange lines) are not reached on average. Please note, though, that the baseline results are reported for an optimized classification threshold. Nonetheless, the distances of the final average F1 scores are rather small: -0.008 (E1S), -0.003 (E1M), -0.008 (E1L), 0.012 (E2S) and -0.026 (E2M) (all results for $b = 300$ and $err = 0.2$).

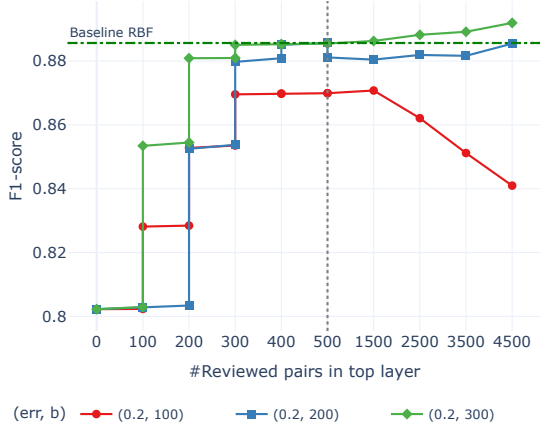


Figure 5.5: Linkage quality development for selected runs shows F1 score changes due to reviewed pairs followed by vertical steps due to updates of the threshold in L_R in the warm-up phase.

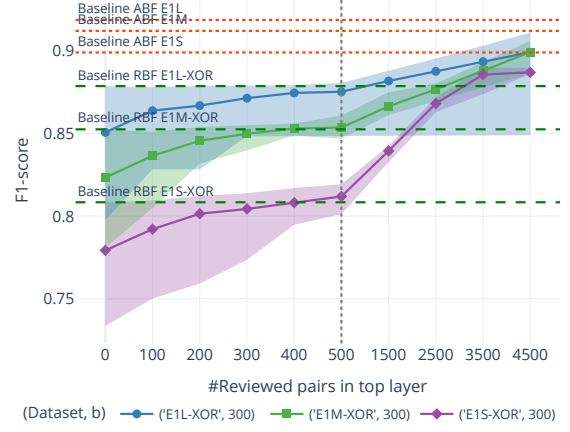


Figure 5.6: Linkage quality development for datasets E1S, E1M and E1L with XOR hardened CLK encodings, fixed $b = 300$ and $err = 0.2$.

	Enc.	FN	MN	LN	YOB	CITY	ZIP	POB
Number of encoded attributes	ABF	100k	92.2k	100k	100k	100k	99.9k	80.8k
	KABF	9k	8.4k	9k	9k	9k	9.0k	7.6k
Gini coefficient (G)	ABF	0.171	0.157	0.147	0.388	0.212	0.330	0.373
	KABF	0.010	0.009	0.010	0.010	0.009	0.009	0.010
Jensen-Shannon diverg. (JSD)	ABF	0.132	0.121	0.114	0.299	0.162	0.251	0.296
	KABF	0.008	0.007	0.007	0.008	0.007	0.007	0.008

Table 5.5: Privacy measures in L_A for Keyed Attribute-level Bloom filter (KABF) compared to the baseline approach using only ABF-based linkage on E1M. The given values for KABF are the mean over all experimental runs for that dataset.

Privacy Table 5.5 compares the privacy measures of these ABF baselines with our approach using *keyed* attribute-level Bloom filters. The number of available encodings in L_A is decreased from up to 100,000 (total number of records in the dataset) to up to 9,000 (two times the number of record pair review requests by L_R). For some attributes, in particular middle name and place of birth, the counts are lower due to missing values. Both Bloom filter privacy measures are heavily reduced from 0.147 – 0.388 (G) and 0.114 – 0.299 (JSD) to at most 0.01. For attributes with a high variety of values, in particular name components, the decreasements are the lowest. The largest improvements are achieved for the year of birth attribute, likely due to very frequent bigrams such as '19' which are reflected in having very frequent corresponding bit positions in conventional ABF. This illustrates the privacy-enhancing effect of the pair-specific hashing secrets in our protocol.

Figure 5.7 shows the distributions of the privacy measures in L_C for datasets E1M and E2M based on the final states of the runs with $b = 200$ and $err = 0.2$. Each measure

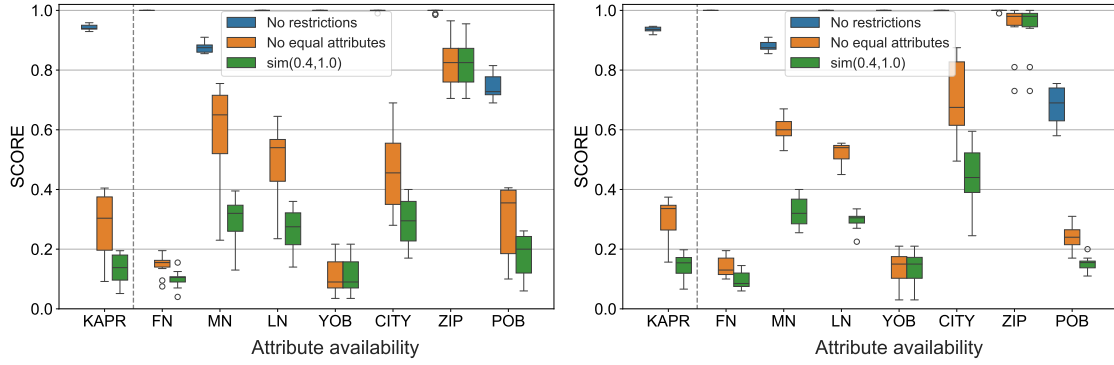


Figure 5.7: Comparison of the privacy measures in L_C for datasets E1M (left) and E2M (right) using different attribute selection strategies. All measures to the right of KAPR refer to the availability of the respective attribute in this layer.

is computed for three attribute selection methods: *No restrictions* refers to the baseline where all attributes of uncertain pairs in L_A are provided to L_C . In *No equal attributes* the data owners are asked to provide all attributes whose similarity is below 1, whereas in the last setting, an additional filter is applied to exclude very dissimilar attributes ($sim < 0.4$) as well. In the first setting, the results represent the availability of the attributes in the plain dataset. With stronger restrictions, the share of attributes that have been requested and provided decreases for most attributes. While all attributes of E1M apart from MN and POB are available for (nearly) all records in the unrestricted setting, the average sharing rate is reduced to a high (FN, YOB) or moderate (LN, CITY) degree using the proposed selection strategies. The ZIP code, however, is still available in plaintext for most records (median $\approx 83\%$). For E2M the availability of the location-related attributes is even higher, close to 100% for the ZIP code. While the median scores are otherwise comparable to E1M, the results show a lower variance. The KAPR score for both datasets is reduced from above 0.9 to below 0.4 and 0.2 using the two filtering methods.

5.5 DISCUSSION

In general, the results show that the proposed protocol achieves its aim of improving the linkage quality with a limited amount of manual labeling.

Quality While higher error rates of the masked clerical review naturally are detrimental to the overall performance, it can be observed from Figure 5.3 (right) and Figure 5.4 (left) that protocol runs with increased budgets achieve F1 score improvements by approximately 5 – 10% on average. Both aspects, clerical reviews as well as the model updates, contribute to that, as shown in Figure 5.5. The tuned threshold of M_R is largely responsible for improvements in early iterations with few reviewed instances. F1 scores above the green reference lines represent results that are beyond the optimal initial threshold configuration and therefore could not have been achieved without clerical review based

on additional data in L_A and L_C . In addition, the range of possible outcomes is reduced substantially. This means that data custodians can expect a high linkage quality, less depending on the initial classification threshold choice.

This also applies to the E2 dataset variants in principle, however, a higher budget is required for stable results. As the oracle error rate is identical to the E1 experiments, the cause must be the training of M_A and M_R . The high availability of the residence attributes and the low variance of other attributes in L_C (Figure 5.7) indicates that the reviewed pairs are fairly similar, leading to a biased and poor M_A . In consequence, the performance of M_R also deteriorates.

In our setup, the warm-up phase with AL_A in the first five iterations determines the performance of M_A , because the training will be stopped as soon as the clerical review budget b is reached. M_A serves as the bridge between the more secure comparison of the majority of records based on record-level encodings and the masked clerical review for a small set of uncertain match candidates. For this functionality, it is not necessary that M_A performs better than M_R . In fact, the protocol would even work without the model when (g, p) of M_R is reused in this layer. In that scenario, the computed attribute similarities based on the attribute-level encodings would merely serve to restrict the requested plaintext data for the subsequent clerical review to non-equal attributes.

In order to achieve higher quality improvements, an increased number of reviews is required, because M_R cannot be improved much due to its privacy-induced simplicity with a single feature. However, the number of manual reviews should be kept as low as possible. Therefore, it is important that M_A has a high predictive performance to provide the majority of corrected links. In our evaluation setup we used a fixed budget of 4500 reviews by L_A , analogous to the clerical budget b in L_C . In principle, additional reviews are possible and may raise the linkage quality even further. However, the likelihood that pairs having similarities with larger distance to the threshold in M_R are wrongly classified declines, deteriorating the ratio of corrected to reviewed links.

A possible approach to improve our setup could be a more sophisticated query strategy in L_A based on the available feature vector instead of the probability score only. Furthermore, it could be beneficial using a pre-trained M_A instead of bootstrapping it based on the initial predictions of M_R . Such an approach requires a sufficiently similar reference dataset with a given ground truth for training though.

The ABF baseline results are higher than the RBF baselines as they make use of attribute weights and weight redistribution in case of missing values, even with weights determined on an independent dataset (German cancer registry). The experiments show that our protocol achieves only slightly lower linkage quality compared to this baseline linkage approach. However, the risk of reidentification attacks based on frequent patterns in the underlying attribute-level encodings is greatly reduced, as illustrated by Table 5.5.

Privacy The majority of reviews is handled automatically by the intermediate layer without plaintext access while only a small fraction is reviewed manually based on partial display. After the warm-up, $\frac{b}{500} = 20/40/60\%$ of the uncertain pairs from the initial layer have been reviewed using PPCR, the remainder by M_A . At the end, the proportion is reduced to $\frac{b}{4500} = 2/4/7\%$. We studied basic filtering approaches for lowering the number of requested plaintext attributes based on the attribute similarity. The observed reductions vary between the attributes. In particular, the effect for the numerical ZIP code attribute is comparatively low, because it has a high similarity on average due to the records being from the same US state. As a consequence, it is requested for a high share of pairs. This problem may be addressed in future work by using a dynamic data-driven approach where the lower bound is determined by the average attribute similarity in L_A .

The KAPR scores are drastically decreased due to the attribute selection. Fewer persons can be uniquely reidentified based on the remaining attributes. This is also illustrated by the observation that the availability is particularly reduced for strongly identifying attributes like name and year of birth. Nevertheless, malicious reviewers could still target certain persons, e.g., with rare attribute combinations. The data owners may implement their own selection rules to protect their high profile records.

We want to point out that although our protocol was described and evaluated using Bloom filter encodings due to their popularity, any perturbation technique is applicable that supports record-specific keys and approximate similarity computation. As the experimental results using hardened Bloom filter encodings (see Figure 5.6) suggest, the overall linkage quality might be lower though, despite comparable improvements.

Additional layers could be integrated in the protocol, e.g., using a second record-level encoding technique before requesting attribute-level encodings. It could include attributes that were excluded in the first layer due to a high ratio of missing values. In order to reduce the number of record-level Bloom Filter available for a frequency analysis, an initial comparison round based on non-error tolerant encodings such as simple hashing could be employed. Thereby, identical matches can be identified. Depending on the data quality of the dataset, this type can make up a large proportion of matches. Following the data-minimization-principle for those records no token-based encoding is necessary.

5.6 DEMONSTRATION

Based on this protocol, we demonstrate SecUREmatch [121], an open-source privacy-preserving record linkage framework which integrates clerical review with minimal information disclosure. This tool allows end-to-end execution of such a protocol on datasets without ground truth and thus without pretrained classification models by providing a graphical user interface for control and clerical review.

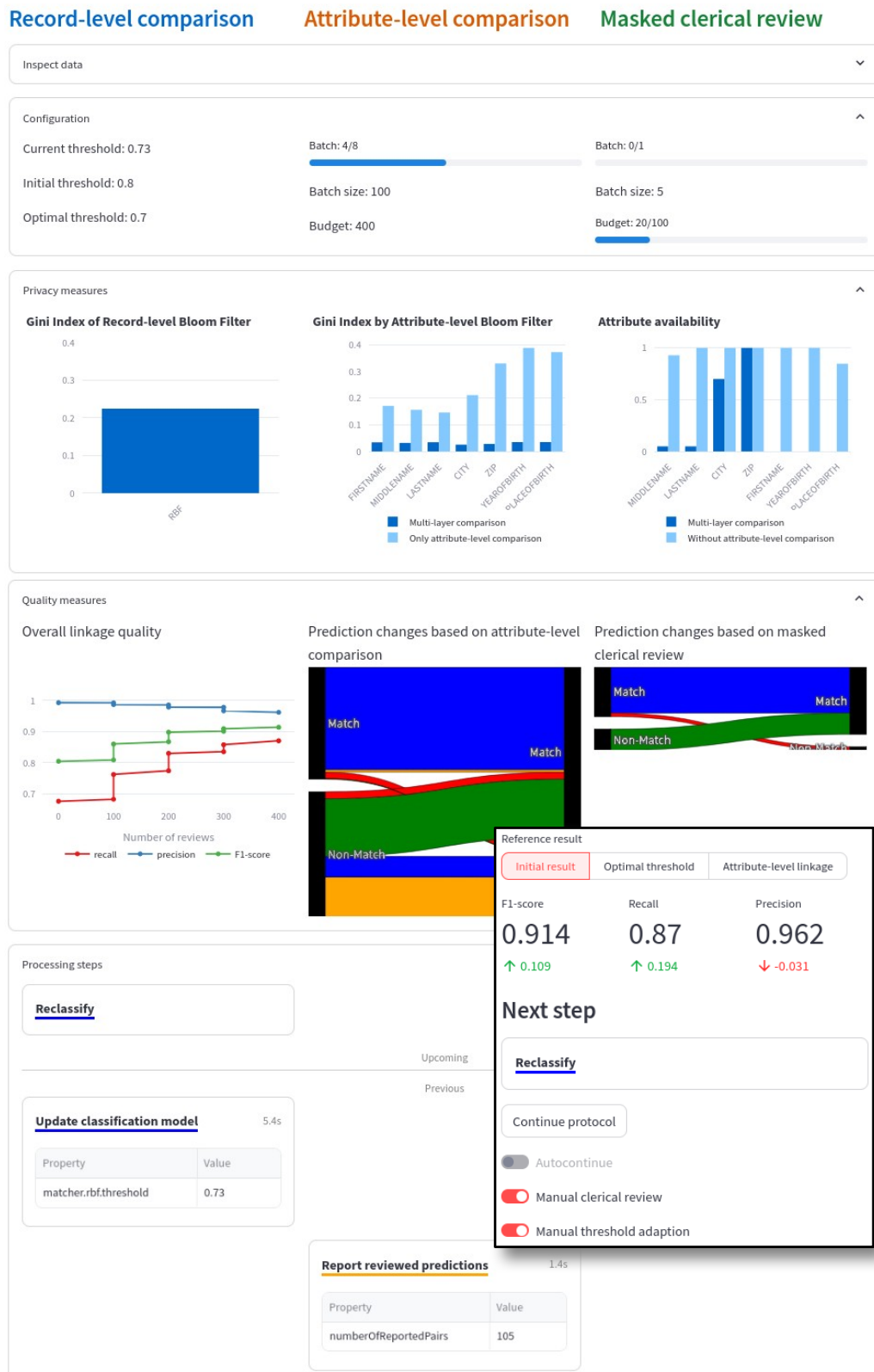


Figure 5.8: Linkage execution interface of SecUREmatch with columnar views for comparison layers. From top to bottom, users can inspect the database contents, the protocol configuration, privacy and quality measures and an overview of the previous processing steps. The overlaid box allows comparison of the current linkage quality with reference results.

Mask equal, very dissimilar attributes, as well as equal characters

FIRSTNAME	LASTNAME	DATEOFBIRTH	CITY
****	● Very Frequent	**.*G.**76	✗
****a		**.*G.**97	

Mask also differing characters with placeholders

FIRSTNAME	LASTNAME	DATEOFBIRTH	CITY
****	● Very Frequent	**.*X.**AB	✗
****X		**.*X.**BA	

Figure 5.9: Illustration of different masking methods for record pairs where differing characters of somewhat similar attributes are display in plaintext or using placeholders whereas very dissimilar or equal attributes are visualized using respective symbols.

Backend services The protocol execution is based on the modular PPRL services described in Section 3.2.3. These services are designed to support multiple linkage projects in parallel. In the multi-layer protocol, the layers are modeled as separate projects and can be managed by different service instances and data custodians. For evaluation purposes, a single L_{LU} service instance can handle all layer projects. Lower comparison layers are given a reference project id that they use for reporting their predictions. The protocol manager service L_{PM} is responsible for triggering the operations of the other services such as providing attribute-level encodings with pair-specific secrets for the attribute-level comparison or selected plaintext attributes for the masked clerical review.

Protocol execution The protocol execution progress is shown and controlled using a web frontend implemented with Streamlit¹, as illustrated in Figure 5.8. The demonstrator displays the previous processing steps as well as a preview of the upcoming step. Each step of the linkage protocol can be triggered manually so that the state can be inspected in between. For convenience, there is an autocontinue option to run the protocol until an optionally defined breakpoint is reached.

Visual masking Figure 5.9 illustrates different levels of disclosure for human decision making. For very similar or dissimilar attributes, the actual values are hardly relevant for the match classification and thus, are not displayed. In contrast, the (categorical) frequency of agreeing attributes is important and therefore included in the visualization. For somewhat similar attributes, the agreeing characters are masked. The full masking even replaces disagreeing parts with placeholders. For convenience, the review process can also be simulated with an adjustable error rate given a benchmark dataset with ground truth information.

Control Apart from the masked clerical review itself, the demonstration allows to manually adapt the classification threshold of M_R using the similarity distribution in L_R with the labels from clerical review, as illustrated in Figure 5.10.

¹<https://streamlit.io/>



Figure 5.10: Manual threshold selection based on the predictions of different layers allows to control the process. The overall similarity distribution (left) illustrates the challenge of M_R for choosing a suitable threshold whereas focusing on the more reliable predictions of lower layers ABF and CR (right) is beneficial for threshold selection.

Linkage quality development The overall purpose of the multi-layer clerical review is an improvement of the linkage quality with limited labeling. Therefore, after each batch of reviewed labels and after each subsequent model update, the linkage quality is computed based on the ground truth of the experimental datasets. Linkage quality is shown using standard metrics for binary classification tasks with imbalanced classes, namely recall, precision and F1 score. The tool can also be used in an alternative execution mode when no ground truth is available or shall not be displayed in order to simulate a real-world linkage scenario. The user interface illustrates the match prediction changes based on attribute-level comparison and masked clerical review which allows the users to assess their usefulness.

Privacy analysis To illustrate the privacy benefits of the multi-layer approach, privacy measures are compared to a linkage result solely based on attribute-level encodings as well as a linkage process where uncertain pairs from record-level comparison are manually reviewed without the intermediate attribute-level comparison.

5.7 CONCLUSION

By their very nature, privacy-preserving classification algorithms are difficult to parameterize in practice due to the lack of labeled training data, e.g., from clerical review. In privacy-preserving record linkage, record-level Bloom filter encodings are frequently used to improve resistance against reidentification attacks but suffer from potential quality issues and the need for a well-selected classification threshold. We presented an active learning based protocol that enables increased and more stable linkage quality using multiple layers of clerical review. Merely a low number of masked manual reviews is required because the majority of uncertain pairs can be classified automatically with higher accuracy based on attribute-level features with the trained models. Sharing additional information with the linkage unit increases the risk of reidentification. We therefore employ record-specific salting and attribute selection to hamper such attacks. Furthermore, data owners still remain in full control of the information they are willing to share for each record.

The proposed multi-layer clerical review is also applicable in incremental linkage, e.g., for pseudonymous patient registries, and would allow monitoring of the linkage quality over time. In future work, it would be interesting to investigate whether the suitability of record-level encoding parameters can be assessed based on the ability of the record-level model M_R to reproduce labels from the attribute-level model M_A . In particular, since attribute-level comparison allows to determine attribute-specific error rates – which is not possible with record-level encodings – this information can be used to estimate weights and compare them to the assumed setting. However, the labeled examples are not representative for the linkage problem, since they are chosen based on their uncertainty, which necessitates a different sampling strategy for this purpose. Another potential application of the labeled samples is the computation of calibrated confidence scores in L_R , enabling data analysts to incorporate the uncertainty of the linkage result.

6

Conclusion and Outlook

This doctoral thesis presented research contributions towards more robust, transparent and reliable PPRL. In Section 6.1 the key challenges are outlined and the contributions to tackle them. Finally, limitations and future research directions are discussed in Section 6.2.

6.1 CONCLUSION

Privacy-preserving record linkage enables the detection of corresponding records of real-world entities in databases of different organizations without revealing identifying plaintext to each other or to a third party. It is rarely an isolated process but typically embedded in a more complex data integration project which aims to combine distributed sensitive, often person-related information. The (legal) requirements for conducting such linkage projects are substantial and in many cases affected persons need to give their consent. Without doubt, it is essential to demonstrate and improve the resistance of PPRL methods against reidentification attacks to gain trust. Even if the linkage itself is secure and private, there is a risk of reidentification from the combined, supposedly anonymized or pseudonymized data. Furthermore, the algorithm must also make linkage of larger datasets practical feasible and enable high linkage quality despite the presence of data errors. A widely used PPRL approach designed to meet this requirement relies on Bloom filter encodings, which is why this thesis concentrates on this technique. The principles of the presented contributions should be transferrable to other perturbation based protocols though. Record-level encodings are a recommended approach to reduce the risk of attacks but impose considerable limitations on the linkage algorithm. This work examines methods to enhance the robustness and quality of the results focusing on this encoding scheme, while still maintaining a high level of privacy protection.

DESIGN OF AN EVALUATION FRAMEWORK FOR PPRL TO IMPROVE TRANSPARENCY

A comprehensive evaluation framework including components for dataset creation, the linkage execution itself as well as the analysis of the results is introduced. It enables broader experimental studies of linkage methods on a variety of datasets, including selected subsets of existing (real-world) datasets and generation of linkage problems with different data quality scenarios in a holistic tool. The integration of the tag system throughout this framework makes it possible to analyze which plaintext and encoding characteristics of individual records and record pairs prevail in missed matches and false links, thereby supporting the identification of limitations in encoding techniques.

STUDY ROBUSTNESS OF RECORD-LEVEL PPRL ON A VARIETY OF DATASETS The framework is utilized to study the robustness of PPRL approaches based on record-level Bloom filter encodings. For that, a variety of benchmark linkage problems are created that exhibit different data characteristics having particular influence on the parametrization of the linkage algorithm such as value frequency distributions. The findings show that the choice of weights substantially influences the quality of the linkage in most cases. The PPRL approach is mostly outperformed by the state-of-the-art deep learning based entity resolution solution. The outcomes across various subgroups of personal real-world records indicate that the plaintext method achieves F1 scores and AUC values that are higher by approximately 0.05 to 0.1. Both the strong weight dependency and the shortcomings compared to plaintext linkage are not surprising, given the rather simple PPRL linkage approach. Nevertheless, they emphasize an important aspect that needs attention and indicate that there is substantial potential for improvement in order to *close the gap*.

ADAPTIVE WEIGHTING SCHEME TO IMPROVE ROBUSTNESS While several encoding schemes were suggested to embed attribute weights into record-level encodings, the determination of appropriate weight values and their overall effect has not been examined thoroughly. An adaptive weighting scheme is proposed based on frequencies of individual attribute values to improve the robustness and quality of PPRL results. A key challenge for attribute weighting in the PPRL context is that weights must be incorporated at the encoding stage, which prevents their flexible adjustment in the linkage process. Furthermore, relevant information on the frequency distributions and error rates is not readily available. The evaluation shows that the CLK-RBF encoding method is rather tolerant to weight variations which arise when attributes of matching records differ. The experiments on various datasets indicate that the scheme still shows improvements in certain scenarios if only source-specific frequency distributions are available. However, this approach benefits from consistent frequency distributions, resulting in gains of F1 score and AUC for all subgroup linkage problems of up to 0.08 and 0.05, respectively.

INTEGRATION OF CLERICAL REVIEW FOR IMPROVED RELIABILITY, CONTROL AND ROBUSTNESS PPRL linkage results based on record-level encodings are highly sensitive to the chosen threshold and are particularly affected by uncertain links whose similarity scores are close to the decision boundary. A linkage protocol is presented that enables labeling of record pairs using a multi-layer clerical review process where information is gradually disclosed only if needed. Only a subset of the uncertain pairs is reviewed manually by an oracle, using a view that shows only selected masked attributes. Most labeling is done without human intervention utilizing hardened attribute-level encodings and a classification model built in active learning cycles based on the manually reviewed pairs. While adding complexity to the otherwise straightforward linkage protocol, it enables to control the linkage outcome with respect to the balance of false and missed matches and to gain confidence in the otherwise unverified results.

Overall, the contributions of this work address the challenges for achieving reliable and robust results in PPRL, while being transparent with regard to the influencing factors. They enable data custodians to exercise more control, giving them increased confidence in the results, while providing a reasonable level of privacy protection.

6.2 OUTLOOK

In the final section, limitations of this work and further aspects that seem worthwhile studying in future work are discussed.

BENCHMARK DATASETS A central challenge in PPRL research is the absence of realistic benchmark datasets. There are several datasets that are frequently used in the literature, most notably derived from the North Carolina voter registry. While multiple datasets were created in this thesis to study the robustness of PPRL methods, their overall diversity remains limited. Only age and gender-based subgroups were adopted whereas in practice use cases could comprise much more diverse groups, such as with rural and urban residency.

Moreover, the majority of existing work focuses on the western hemisphere, in particular its typical dataset characteristics of name and address formats. This is partly due to the limited availability of real-world datasets. The data generator used to create the German records currently models only a subset of dependencies. For instance, it does not yet account for relationships between age or year of birth and given names. The generation of household structures is promising, but it needs further work and evaluation.

The application domain of PPRL methods is naturally concerned with sensitive data which is often person related. This is the reason why this work and most of the literature focuses on datasets comprising attributes like name, date of birth and address. Less work

is done on linkage including also geographical, transactional or sensor information. One example use case is fraud detection, where the goal is to identify recurring patterns within such a *multi-domain* dataset rather than focusing solely on matching individual personal records. Encodings for geodata or numerical information have been proposed, but not in combination with person-related (string) attributes, similar to record-level encodings.

SCHEMA MATCHING One of the presumptions of this thesis is a common schema of the databases, e.g., by transforming the original data to a given global schema. However, this may induce problems when trying to fit data into a common schema which may not be suitable, e.g., due to different cultural naming practices or from different use case domains. There is recent work on privacy-preserving schema matching [4, 60] and it would be interesting to see if it can be more tightly integrated in the PPRL process.

A related issue involves potential variations in the semantic or format of attributes. An example of this is the place of birth attribute in our benchmark datasets: while in the North Carolina dataset it is a two-digit abbreviation of the *state* of birth, in the German datasets it contains the *city* of birth. This has of course consequences regarding the parametrization, in particular differences in optimal weighting.

ADOPTION OF OTHER ENCODING METHODS This work focuses on Bloom filter based PPRL methods due to their popularity. Nevertheless, a variety of other encoding techniques has been proposed, as outlined in Section 2.3.3. It would be worthwhile to explore how the contributions of this work can be incorporated into protocols and encoding methods that offer stronger resistance to attacks than Bloom filter encodings. The dynamic attribute weighting scheme described in Chapter 4 uses frequency information only but there are other value-specific traits, in particular error likelihood, that could be incorporated.

SECRET MANAGEMENT IN MANY-PARTY LINKAGE An important but often neglected aspect in perturbation-based PPRL protocols concerns the necessary parameter exchange between the data owners, in particular of the secret for the keyed hash functions. The parameters must remain confidential to the data owners; otherwise, the linkage unit could perform a dictionary attack. Although acceptable when only few trusted data owners are involved, this constraint renders these PPRL methods impractical for many-party settings, in particular with free participation using, e.g., apps, due to the likely leakage of the encoding secrets. One option would be to use SMPC techniques to collaboratively compute the encoding function so that the secret is shared, i.e., divided, between the data owners [138].

INCLUSION OF EXTERNAL AND USE CASE SPECIFIC KNOWLEDGE When linking personal records, the criteria that define matches are usually quite similar, so it is generally unnecessary to compute new attribute weights or build the classification model from the ground up. Further work could explore how transfer learning and model reuse can be applied to PPRL [94]. This would also benefit the proposed multi-layer protocol, since the attribute-level model could be a pretrained one rather than a bootstrapped model, that would likely improve performance and reduce the need for manually labeling relatively unambiguous match candidates. In certain situations, automatic labeling could instead be performed by employing a secure oracle that estimates how likely it is that the payload data actually belongs to the same individual.

USE CASE DRIVEN PROTOCOL SELECTION While maximizing the linkage quality is generally desirable, in PPRL it typically comes at the cost of lowering the attack resistance as more information has to be revealed to the classification model. Advanced attacks on PPRL, such as graph-based approaches, often rely on the existence of pair similarities. If the data to be linked does not suffer from certain quality issues, such as typographical errors or variations in names, it may be viable to apply a single hash function to different attribute combinations rather than using similarity-preserving encodings. This approach is employed for instance in the US health research network PCORnet [88]. The match-key method is a promising technique that automatically identifies appropriate attribute combinations [117].

A semi-automatic assessment of the each data owners data acquisition practices is desirable to determine possible data error types in order to build a recommendation system for the most suitable PPRL approach. The linkage quality considered *sufficient* or *adequate* is use case specific and the priority of recall and precision is as well. Further work is needed on how to include such desiderata in the linkage process beyond provision of (calibrated) match probabilities along the payload data to the analysts which enables them to set their custom match threshold. Such research would benefit from the joint work of experts in record linkage methods as well as in secondary health research in order to study approaches in a realistic context.

List of Figures

2.1	Record linkage workflow for matching two databases.	12
2.2	Example of data normalization for record preprocessing.	12
2.3	Example of standard blocking using two strategies for key generation. . .	13
2.4	Overview of common similarity measures for record linkage.	14
2.5	Example of linkage result postprocessing	17
2.6	Confusion matrix for evaluating binary classification results.	18
2.7	Illustration of quality measures for binary classification of similarity distributions with low and high overlap of matches and non-matches . .	20
2.8	Separation principle of identifying information and microdata in sec- ondary research	21
2.9	Example of an attack by aligning encoded and plaintext values by their respective frequencies.	24
2.10	Comparison of attribute- and record-level schemata for encoding an example record	25
2.11	Example for encoding two similar values using Bloom filters	27
2.12	Encoding of an example record using the weighted record-level techniques RBF and CLK-RBF	28
3.1	Architecture of the evaluation framework with linkage input creation components, PPRL execution services, and the result analysis modules. .	40
3.2	Attribute dependency graph of the data generator based on German census data	43
3.3	Example use of the evaluation framework based on the data creation path $D_G \rightarrow D_C$	49
3.4	Example for group-based quality assessment	50
3.5	Example for prediction-based analysis on distributions of attribute fre- quencies and Bloom filter fill ratios.	50
3.6	Overview of linkage problem dimensions and their creation scenarios. . .	51
3.7	Comparison of attribute lengths of the linkage datasets created using the corruption module on base datasets from Germany (GER), Baden- Württemberg (BW) and North Carolina (NC) as well as linkage datasets derived by selection from the NC voter registry.	54

LIST OF FIGURES

3.8	Comparison of attribute missingness of the linkage datasets created using the corruption module on base datasets from Germany (GER), Baden-Württemberg (BW) and North Carolina (NC) as well as linkage datasets derived by selection from the NC voter registry.	54
3.9	Frequency distributions of the most common values of selected attributes in the linkage datasets derived from North Carolina voter registry	56
3.10	Frequency distributions of the most common values of selected attributes in the linkage datasets based on the generator for GER and BW.	57
3.11	Overview of the most common difference patterns of true matches in the linkage datasets derived from the North Carolina voter registry.	57
3.12	Comparison of quality metrics using different blocking strategies using plaintext and HLSH, each with and without ground truth IDs.	61
3.13	Comparison of linkage quality results of different blocking strategies on plaintext (PT) with and without ground truth IDs (GT) and encoding strategies for the dataset NC-D-100L.	62
3.14	Linkage quality with different weight sets based datasets from the origins GER, BW and NC and different corruption scenarios T, D and F.	63
3.15	Linkage quality with different weight sets on subgroups datasets from North Carolina (NC).	63
3.16	Linkage quality differences of PPRL with record-level encodings to a plaintext baseline using the Unicorn deep learning model.	64
4.1	Comparison of quality measures for attribute- and value-specific weighting on CLK-RBF for R	81
4.2	Linkage quality metric differences of value-specific weighting (S_{idf} [0.5,2], CLK-RBF) compared to attribute-specific weights estimated using the Fellegi-Sunter approach.	84
4.3	Histogram of similarity shifts of corresponding record pairs due to value-specific weight adaptation with consistent frequency distributions.	84
4.4	Linkage quality metric differences of value-specific weighting compared to attribute-specific weights for corruption-based linkage problems from GER (top), BW (middle) and NC (bottom).	86
4.5	Histogram of similarity shifts of corresponding record pairs due to value-specific weight adaptation with source-specific frequency distributions.	87
4.6	Linkage quality metric differences of value-specific weighting compared to attribute-specific weights for the subgroup linkage problems from North Carolina (NC).	87

5.1	PPRL protocol with three linkage layers, using record-level encodings, keyed attribute-level encodings and masked clerical review.	94
5.2	Example of a reidentification attack for an attribute $L(\text{astname})$ based on RBF encodings and known attribute similarities	98
5.3	Linkage quality development for dataset E1M with varying budget and error rates.	103
5.4	Linkage quality development for dataset E1S, E1L, E2S and E2M with varying budget and fixed error rates.	104
5.5	Linkage quality development of selected runs with two-fold evaluation per iteration	105
5.6	Linkage quality development for datasets E1S, E1M and E1L with XOR hardened CLK encodings.	105
5.7	Comparison of the privacy measures in L_C using different attribute selection strategies.	106
5.8	Linkage execution interface of SecUREmatch with columnar views for comparison layers.	109
5.9	Illustration of masking methods for clerical review of record pairs	110
5.10	Manual threshold selection based on the predictions with different level of disclosure.	111

List of Tables

2.1	Common data quality issues in record pairs	10
3.1	Examples of tags provided by components of the PPRL framework	42
3.2	Example record cluster of a certain individual from temporal snapshots of a voter registry grouped by voter ID.	45
3.3	Weight sets used in the robustness evaluation.	59
4.1	Example of weighted comparison of records with <i>common</i> first and last name.	70
4.2	Example of weighted comparison of records with <i>rare</i> first and last name.	70
4.3	Example of a variation of a single weight on record-level encodings based on RBF and CLK-RBF	77
4.4	Description of used datasets for evaluation of weight adaptation and application methods.	79
4.5	Attribute properties and encoding parameters for ABF, RBF and CLK-RBF in the evaluation of the weight adaption scheme	79
4.6	Linkage quality comparison of weight adaptation methods on Attribute- level Bloom filter.	81
4.7	Comparison of linkage quality results when using averaging, attribute- specific and value-specific weighting for different weight application methods.	82
4.8	Linage quality comparison of attribute-specific and value-specific weight- ing with and without postprocessing with restriction to 1:1 links.	82
4.9	Linkage quality comparison of weighting methods with limited frequency information based on CLK-RBF.	83
5.1	Example of a record pair comparison using different disclosure levels . .	91
5.2	Overview of datasets for evaluating the multi-layer PPRL protocol	100
5.3	Number of hash functions for the Keyed ABF encodings in L_A	101
5.4	Attribute weights $\mathbf{w}$ used for the baseline linkage solely based on (non- keyed) attribute-level Bloom filters.	102
5.5	Privacy measures in L_A for Keyed Attribute-level Bloom filter (KABF) compared to the baseline approach using only ABF-based linkage.	105

Bibliography

- [1] Mohamed Abdelhamid, Joana Gaia, and George Lawrence Sanders. ‘Putting the Focus Back on the Patient: How Privacy Concerns Affect Personal Health Information Sharing Intentions’. In: *Journal of Medical Internet Research* 19.9 (2017), e169. DOI: [10.2196/jmir.6877](https://doi.org/10.2196/jmir.6877).
- [2] Mohammad Alaggan, Sébastien Gambs, and Anne-Marie Kermarrec. ‘BLIP: non-interactive differentially-private similarity computation on bloom filters’. In: *Symposium on Self-Stabilizing Systems*. Springer. 2012, pp. 202–216. DOI: [10.1007/978-3-642-33536-5_20](https://doi.org/10.1007/978-3-642-33536-5_20).
- [3] Carlo Batini, Cinzia Cappiello, Chiara Francalanci, and Andrea Maurino. ‘Methodologies for Data Quality Assessment and Improvement’. In: *ACM Computing Surveys* 41.3 (2009), pp. 1–52. DOI: [10.1145/1541880.1541883](https://doi.org/10.1145/1541880.1541883).
- [4] Domenico Beneventano, Sonia Bergamaschi, and Carmelo La Corte Corte. ‘A Privacy-Preserving Schema Matching Technique Based on k-Anonymity’. In: *ITADATA2025: The 4th Italian Conference on Big Data and Data Science*. 2025.
- [5] Martin Bialke, Thomas Bahls, Christoph Havemann, Jens Piegsa, K Weitmann, Tim Wegner, and Wolfgang Hoffmann. ‘MOSAIC – A Modular Approach to Data Management in Epidemiological Studies’. In: *Methods of Information in Medicine* (2015), pp. 1–8. DOI: [10.3414/ME14-01-0133](https://doi.org/10.3414/ME14-01-0133).
- [6] Martin Bialke, Peter Penndorf, Tim Wegner, Thomas Bahls, Christoph Havemann, Jens Piegsa, and Wolfgang Hoffmann. ‘A Workflow-Driven Approach to Integrate Generic Software Modules in a Trusted Third Party’. In: *Journal of Translational Medicine* 13.1 (2015), p. 176. DOI: [10.1186/s12967-015-0545-6](https://doi.org/10.1186/s12967-015-0545-6).
- [7] Burton Howard Bloom. ‘Space/Time Trade-Offs in Hash Coding with Allowable Errors’. In: *Communications of the ACM* 13.7 (1970), pp. 422–426. DOI: [10.1145/362686.362692](https://doi.org/10.1145/362686.362692).
- [8] Dan Bogdanov, Sven Laur, and Jan Willemson. ‘Sharemind: A Framework for Fast Privacy-Preserving Computations’. In: *Computer Security - ESORICS 2008*. Vol. 5283. Springer Berlin Heidelberg, 2008, pp. 192–206. DOI: [10.1007/978-3-540-88313-5_13](https://doi.org/10.1007/978-3-540-88313-5_13).
- [9] Christian Borgs. ‘Optimal Parameter Choice for Bloom Filter-based Privacy-Preserving Record Linkage’. PhD thesis. Universität Duisburg-Essen, 2019.

- [10] François Borst, François-André Allaert, and Catherine Quantin. ‘The Swiss Solution for Anonymously Chaining Patient Files’. In: *Studies in health technology and informatics* 84.Pt 2 (2001), pp. 1239–1241. DOI: [10.3233/978-1-60750-928-8-1239](https://doi.org/10.3233/978-1-60750-928-8-1239).
- [11] James H. Boyd, Sean. M. Randall, Adrian P. Brown, Max Maller, Davie Botes, Margo Gillies, and Anna Ferrante. ‘Population Data Centre Profiles: Centre for Data Linkage’. In: *International Journal of Population Data Science* 4.2 (2020), p. 1139. DOI: [10.23889/ijpds.v4i2.1139](https://doi.org/10.23889/ijpds.v4i2.1139).
- [12] Leo Breiman. ‘Random forests’. In: *Machine Learning* 45 (2001), pp. 5–32. DOI: [10.1023/A:1010933404324](https://doi.org/10.1023/A:1010933404324).
- [13] Adrian P. Brown, Sean M. Randall, Anna M. Ferrante, James B. Semmens, and James H. Boyd. ‘Estimating parameters for probabilistic linkage of privacy-preserved datasets’. In: *BMC Medical Research Methodology* 17.95 (2017). DOI: [10.1186/s12874-017-0370-0](https://doi.org/10.1186/s12874-017-0370-0).
- [14] Adrian Calma. ‘Active Learning with Uncertain Annotators: Towards Dedicated Collaborative Interactive Learning’. PhD thesis. kassel University Press, 2020. ISBN: 9783737608749.
- [15] Lidia Ceriani and Paolo Verme. ‘The origins of the Gini index: extracts from Variabilità e Mutabilità (1912) by Corrado Gini’. In: *The Journal of Economic Inequality* 10.3 (2012), pp. 421–443. DOI: [10.1007/s10888-011-9188-x](https://doi.org/10.1007/s10888-011-9188-x).
- [16] Peter Christen. ‘A Survey of Indexing Techniques for Scalable Record Linkage and Deduplication’. In: *IEEE Transactions on Knowledge and Data Engineering* 24.9 (2012), pp. 1537–1555.
- [17] Peter Christen. *Data Matching. Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Data-Centric Systems and Applications. Springer Berlin, Heidelberg, 2012. DOI: [10.1007/978-3-642-31164-2](https://doi.org/10.1007/978-3-642-31164-2).
- [18] Peter Christen. ‘Development and user experiences of an open source data cleaning, deduplication and record linkage system’. In: *ACM SIGKDD Explorations Newsletter* 11.1 (2009), pp. 39–48. DOI: [10.1145/1656274.1656282](https://doi.org/10.1145/1656274.1656282).
- [19] Peter Christen, David J. Hand, and Nishadi Kirielle. ‘A Review of the F-Measure: Its History, Properties, Criticism, and Alternatives’. In: *ACM Computing Surveys* 56.3 (2023). DOI: [10.1145/3606367](https://doi.org/10.1145/3606367).
- [20] Peter Christen, Thilina Ranbaduge, and Rainer Schnell. *Linking Sensitive Data. Methods and Techniques for Practical Privacy-Preserving Information Sharing*. Vol. 1. Springer Cham, 2020. DOI: [10.1007/978-3-030-59706-1](https://doi.org/10.1007/978-3-030-59706-1).

-
- [21] Peter Christen, Thilina Ranbaduge, Dinusha Vatsalan, and Rainer Schnell. ‘Precise and Fast Cryptanalysis for Bloom Filter Based Privacy-Preserving Record Linkage’. In: *IEEE Transactions on Knowledge and Data Engineering* 31.11 (2019), pp. 2164–2177. DOI: [10.1109/TKDE.2018.2874004](https://doi.org/10.1109/TKDE.2018.2874004).
- [22] Peter Christen and Dinusha Vatsalan. ‘Flexible and extensible generation and corruption of personal data’. In: *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*. CIKM ’13. San Francisco, California, USA: Association for Computing Machinery, 2013, pp. 1165–1168. ISBN: 9781450322638. DOI: [10.1145/2505515.2507815](https://doi.org/10.1145/2505515.2507815).
- [23] Peter Christen, Anushka Vidanage, Thilina Ranbaduge, and Rainer Schnell. ‘Pattern-Mining Based Cryptanalysis of Bloom Filters for Privacy-Preserving Record Linkage’. In: *Advances in Knowledge Discovery and Data Mining*. Springer, Cham, 2018, pp. 530–542. DOI: [10.1007/978-3-319-93040-4_42](https://doi.org/10.1007/978-3-319-93040-4_42).
- [24] Victor Christen, Tim Häntschel, Peter Christen, and Erhard Rahm. ‘Privacy-Preserving Record Linkage Using Autoencoders’. In: *International Journal of Data Science and Analytics* 15.4 (2023), pp. 347–357. DOI: [10.1007/s41060-022-00377-2](https://doi.org/10.1007/s41060-022-00377-2).
- [25] Paolo Contiero, Andrea Tittarelli, Giovanna Tagliabue, Anna Maghini, Sabrina Fabiano, Piergiorgio Crosignani, and Roberto Tessandori. ‘The EpiLink Record Linkage Software: Presentation and Results of Linkage Test on Cancer Registry Files’. In: *Methods of Information in Medicine* 44.1 (2005), pp. 66–71. PMID: [15778796](https://pubmed.ncbi.nlm.nih.gov/15778796/).
- [26] Patricia Correll, Anne-Marie Feyer, Phuong-Thao Phan, Barry Drake, Walid Jammal, Katie Irvine, Adrian Power, Sharon Muir, Shahana Ferdousi, Samantha Moubarak, Yalchin Oytam, James Linden, and Louise Fisher. ‘Lumos: a statewide linkage programme in Australia integrating general practice data to guide system redesign’. In: *Integrated Healthcare Journal* 3.1 (2021). DOI: [10.1136/ihj-2021-000074](https://doi.org/10.1136/ihj-2021-000074).
- [27] J De Bruin. *Python Record Linkage Toolkit: A toolkit for record linkage and duplicate detection in Python*. Version v0.14. 2019. DOI: [10.5281/zenodo.3559043](https://doi.org/10.5281/zenodo.3559043).
- [28] Lea Demelius, Karl Kreiner, Dieter Hayn, Michael Nitzlnader, and Günter Schreier. ‘Encoding of Numerical Data for Privacy- Preserving Record Linkage’. In: *Studies in health technology and informatics*. IOS Press 271 (2020), pp. 23–30. DOI: [10.3233/SHIT200070](https://doi.org/10.3233/SHIT200070).
- [29] Lee Raymond Dice. ‘Measures of the Amount of Ecologic Association Between Species’. In: *Ecology* 26.3 (1945), pp. 297–302. DOI: [10.2307/1932409](https://doi.org/10.2307/1932409).

- [30] James Doidge, Peter Christen, and Katie Harron. ‘Quality Assessment in Data Linkage’. In: *Joined up Data in Government: The Future of Data Linking Methods [Internet]*. UK Government Analysis Function and Office for National Statistics, 2021.
- [31] Duc Dung Dao. ‘Konzeption und Realisierung einer Anwendung zur Generierung von personenbezogenen Daten’. Bachelor thesis. Leipzig University, 2022.
- [32] Elizabeth A. Durham, Murat Kantarcioglu, Yuan Xue, Csaba Toth, Mehmet Kuzu, and Bradley Malin. ‘Composite Bloom Filters for Secure Record Linkage’. In: *IEEE Transactions on Knowledge and Data Engineering* 26.12 (2014), pp. 2956–2968. DOI: [10.1109/TKDE.2013.91](https://doi.org/10.1109/TKDE.2013.91).
- [33] Elizabeth Ashley Durham. ‘A framework for accurate, efficient private record linkage’. PhD thesis. Vanderbilt University, 2012.
- [34] Cynthia Dwork. ‘Differential privacy’. In: *International colloquium on automata, languages, and programming*. Springer, Berlin, Heidelberg. 2006, pp. 1–12. DOI: [10.1007/11787006_1](https://doi.org/10.1007/11787006_1).
- [35] European Parliament and Council of the European Union. *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance)*. 2016.
- [36] Ivan P. Fellegi and Alan B. Sunter. ‘A Theory for Record Linkage’. In: *Journal of the American Statistical Association* 64.328 (1969), pp. 1183–1210. DOI: [10.2307/2286061](https://doi.org/10.2307/2286061).
- [37] *FHIR Implementation Guides*. Public Health Informatics Institute. URL: <https://phii.org/module-5/fhir-implementation-guides/> (visited on 01/29/2026).
- [38] Jeremy Foxcroft, Peter Christen, and Luiza Antonie. ‘Class Ratio and Its Implications for Reproducibility and Performance in Record Linkage’. In: *Advances in Knowledge Discovery and Data Mining*, ed. by De-Nian Yang et al. Vol. 14645. Springer, Singapore, 2024, pp. 194–205. DOI: [10.1007/978-981-97-2242-6_16](https://doi.org/10.1007/978-981-97-2242-6_16).
- [39] Eibe Frank, Mark A Hall, and Ian H Witten. *The WEKA workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques"*. 4th. San Francisco, California: Morgan Kaufmann, 2016.
- [40] Martin Franke. ‘Scalable, Accurate and Secure Privacy-Preserving Record Linkage’. [urn:nbn:de:bsz:15-qucosa2-912319](https://nbn-resolving.org/urn:nbn:de:bsz:15-qucosa2-912319). PhD thesis. Leipzig University, Germany, 2024.

-
- [41] Martin Franke, Victor Christen, Peter Christen, **Florens Rohde**, and Erhard Rahm. ‘(Privately) Estimating Linkage Quality for Record Linkage’. In: *Proceedings of the 27th International Conference on Extending Database Technology (EDBT)*. 2024. DOI: [10.48786/EDBT.2024.26](https://doi.org/10.48786/EDBT.2024.26).
- [42] Martin Franke, Marcel Gladbach, Ziad Sehili, **Florens Rohde**, and Erhard Rahm. ‘ScaDS research on scalable privacy-preserving record linkage’. In: *Datenbank-Spektrum* 19 (2019), pp. 31–40. DOI: [10.1007/s13222-019-00305-y](https://doi.org/10.1007/s13222-019-00305-y).
- [43] Martin Franke, Ziad Sehili, Marcel Gladbach, and Erhard Rahm. ‘Post-processing Methods for High Quality Privacy-Preserving Record Linkage’. In: *Data Privacy Management, Cryptocurrencies and Blockchain Technology*. 2018. DOI: [10.1007/978-3-030-00305-0_19](https://doi.org/10.1007/978-3-030-00305-0_19).
- [44] Martin Franke, Ziad Sehili, and Erhard Rahm. ‘Parallel Privacy-Preserving Record Linkage using LSH-based blocking’. In: *International Conference on Internet of Things, Big Data and Security (IoTBDs)*. 2018. DOI: <https://doi.org/10.5220/0006682701950203>.
- [45] Martin Franke, Ziad Sehili, and Erhard Rahm. ‘PRIMAT: a toolbox for fast privacy-preserving matching’. In: *Proc. VLDB Endow.* 12.12 (2019), pp. 1826–1829. DOI: [10.14778/3352063.3352076](https://doi.org/10.14778/3352063.3352076).
- [46] Martin Franke, Ziad Sehili, **Florens Rohde**, and Erhard Rahm. ‘Evaluation of Hardening Techniques for Privacy-Preserving Record Linkage’. In: *Proceedings of the 24th International Conference on Extending Database Technology (EDBT)*. 2021, pp. 289–300. DOI: [10.5441/002/edbt.2021.26](https://doi.org/10.5441/002/edbt.2021.26).
- [47] Klaus Giersiepen, Tobias Bachteler, Tobias Gramlich, Jörg Reiher, B. Schubert, Igor Novopashenny, and Rainer Schnell. ‘Performance of record linkage for cancer registry data linked with mammography screening data’. In: *Bundesgesundheitsblatt, Gesundheitsforschung, Gesundheitsschutz* 53.7 (2010), pp. 740–747. ISSN: 1436-9990. DOI: [10.1007/s00103-010-1084-1](https://doi.org/10.1007/s00103-010-1084-1).
- [48] Aristides Gionis, Piotr Indyk, Rajeev Motwani, et al. ‘Similarity Search in High Dimensions via Hashing’. In: *VLDB’99, Proceedings of 25th International Conference on Very Large Data Bases, September 7-10, 1999, Edinburgh, Scotland, UK*, ed. by Malcolm P. Atkinson et al. Morgan Kaufmann, 1999, pp. 518–529. ISBN: 1-55860-615-7.
- [49] Aris Gkoulalas-Divanis and Grigorios Loukides. ‘Medical Data Privacy Handbook’. In: *Medical Data Privacy Handbook*. Springer, Cham, 2015, pp. 1–832. ISBN: 978-3-319-23633-9. DOI: [10.1007/978-3-319-23633-9](https://doi.org/10.1007/978-3-319-23633-9).

- [50] Aris Gkoulalas-Divanis, Dinusha Vatsalan, Dimitrios Karapiperis, and Murat Kantarcioglu. ‘Modern Privacy-Preserving Record Linkage Techniques: An Overview’. In: *IEEE Transactions on Information Forensics and Security* 16 (2021), pp. 4966–4987. DOI: [10.1109/TIFS.2021.3114026](https://doi.org/10.1109/TIFS.2021.3114026).
- [51] Marcel Gladbach, Ziad Sehili, Thomas Kudrass, Peter Christen, and Erhard Rahm. ‘Distributed Privacy-Preserving Record Linkage Using Pivot-Based Filter Techniques’. In: *2018 IEEE 34th International Conference on Data Engineering Workshops (ICDEW)*. IEEE, 2018, pp. 33–38. DOI: [10.1109/ICDEW.2018.00013](https://doi.org/10.1109/ICDEW.2018.00013).
- [52] Martin Graf, Lukas Laskowski, Florian Papsdorf, Florian Sold, Roland Gremelspacher, Felix Naumann, and Fabian Panse. ‘Frost: a platform for benchmarking and exploring data matching results’. In: *Proceedings of the VLDB Endowment* 15.12 (2022), pp. 3292–3305. DOI: [10.14778/3554821.3554823](https://doi.org/10.14778/3554821.3554823).
- [53] Beatrix Haddock, Alix Pletcher, Nathaniel Blair-Stahn, Os Keyes, Matt Kappel, Steve Bachmeier, Syl Lutze, James Albright, Alison Bowman, Caroline Kinuthia, Zeb Burke-Conte, Rajan Mudambi, and Abraham Flaxman. *Simulated data for census-scale entity resolution research without privacy restrictions: a large-scale dataset generated by individual-based modeling*. 2024. DOI: [10.12688/gatesopenres.15418.1](https://doi.org/10.12688/gatesopenres.15418.1).
- [54] Christopher Hampf, Martin Bialke, Hauke Hund, Christian Fegeler, Stefan Lang, Peter Penndorf, Nico Wöller, Frank-Michael Moser, Arne Blumentritt, Ronny Schuldt, et al. ‘Privacy-preserving record linkage by a federated trusted third party (fTTP)–unlocking medical research potential in Germany’. In: *GMS Med Inform Biom Epidemiol* 21.Doc05 (2025). DOI: [10.3205/mibe000277](https://doi.org/10.3205/mibe000277).
- [55] Christopher Hampf, Martin Bialke, Hauke Hund, Christian Fegeler, Stefan Lang, Peter Penndorf, Nico Wöller, Frank-Michael Moser, Arne Blumentritt, Ronny Schuldt, Florian Seidel, Peter Brunecker, Reto Wettstein, Lukas Arnecke, and Wolfgang Hoffmann. ‘Privacy-Preserving Record Linkage by a Federated Trusted Third Party (fTTP) – Unlocking Medical Research Potential in Germany’. In: *GMS Medizinische Informatik, Biometrie und Epidemiologie* 21 (2025). DOI: [10.3205/MIBE000277](https://doi.org/10.3205/MIBE000277).
- [56] Christopher Hampf, Lars Geidel, Norman Zerbe, Martin Bialke, Dana Stahl, Arne Blumentritt, Thomas Bahls, Peter Hufnagl, and Wolfgang Hoffmann. ‘Assessment of Scalability and Performance of the Record Linkage Tool E-PIX® in Managing Multi-Million Patients in Research Projects at a Large University Hospital in Germany’. In: *Journal of Translational Medicine* 18.86 (1 2020), pp. 1–11. DOI: [10.1186/s12967-020-02257-4](https://doi.org/10.1186/s12967-020-02257-4).

- [57] David Hand and Peter Christen. ‘A Note on Using the F-measure for Evaluating Record Linkage Algorithms’. In: *Statistics and Computing* 28.3 (2018), pp. 539–547. ISSN: 0960-3174, 1573-1375. DOI: [10.1007/s11222-017-9746-6](https://doi.org/10.1007/s11222-017-9746-6).
- [58] David J. Hand, Peter Christen, and Nishadi Kirielle. ‘F*: An Interpretable Transformation of the F-measure’. In: *Machine Learning* 110.3 (2021), pp. 451–456. DOI: [10.1007/s10994-021-05964-1](https://doi.org/10.1007/s10994-021-05964-1).
- [59] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. ‘DeBERTa: Decoding-enhanced BERT with Disentangled Attention’. In: *International Conference on Learning Representations*. 2021. DOI: [10.48550/arXiv.2006.03654](https://doi.org/10.48550/arXiv.2006.03654).
- [60] Jan-Eric Hellenberg, Fabian Dustin Mahling, Lukas Laskowski, Felix Naumann, Matteo Paganelli, and Fabian Panse. ‘PRISMA: A Privacy-Preserving Schema Matcher using Functional Dependencies’. In: *International Conference on Extending Database Technology (EDBT)*. 2025. DOI: [10.48786/EDBT.2025.24](https://doi.org/10.48786/EDBT.2025.24).
- [61] Youzhe Heng, Frederik Armknecht, Yanling Chen, and Rainer Schnell. ‘On the effectiveness of graph matching attacks against privacy-preserving record linkage’. In: *PLOS ONE* 17.9 (2022). DOI: [10.1371/journal.pone.0267893](https://doi.org/10.1371/journal.pone.0267893).
- [62] Youzhe Heng, Rainer Schnell, and Frederik Armknecht. ‘Cryptanalysis of the Record Linkage Protocol used by German Cancer Registries’. In: *Sicherheit*. 2024. DOI: [10.18420/SICHERHEIT2024_004](https://doi.org/10.18420/SICHERHEIT2024_004).
- [63] Thomas N. Herzog, Fritz J. Scheuren, and William E. Winkler. *Data Quality and Record Linkage Techniques*. Vol. 1. Springer New York, NY, 2007. DOI: [10.1007/0-387-69505-2](https://doi.org/10.1007/0-387-69505-2).
- [64] Kai Hildebrandt, Fabian Panse, Niklas Wilcke, and Norbert Ritter. ‘Large-Scale Data Pollution with Apache Spark’. In: *IEEE Transactions on Big Data* 6.2 (2020), pp. 396–411. DOI: [10.1109/TBDATA.2016.2637378](https://doi.org/10.1109/TBDATA.2016.2637378).
- [65] Timm Intemann, Knut Kaulke, Dennis-Kenji Kipker, Vanessa Lettieri, Christoph Stallmann, Carsten O. Schmidt, Martin Bialke, Christopher Hampf, Dana Stahl, Martin Lablans, Klaus Kraywinkel, Sebastian Bartholomäus, Anatol-Fiete Näher, Galina Tremper, Mohamed Lambarki, Stefanie March, Fabian Prasser, Anna Christine Haber, Johannes Drepper, Irene Schlünder, Toralf Kirsten, Iris Pigeot, Ulrich Sax, Benedikt Buchner, Sebastian C. Semler, and Wolfgang Ahrens. ‘Verbesserung des Record Linkage für die Gesundheitsforschung in Deutschland – wie können Defizite behoben werden? [Improving record linkage for health research—how to overcome deficiencies?]’ In: *Bundesgesundheitsblatt - Gesundheitsforschung - Gesundheitsschutz* 68 (2025), pp. 1440–1449. DOI: [10.1007/s00103-025-04153-y](https://doi.org/10.1007/s00103-025-04153-y).

- [66] Maximilian Jugl and Toralf Kirsten. ‘Gecko: A Python library for the generation and mutation of realistic personal identification data at scale’. In: *SoftwareX* 27 (2024), p. 101846. DOI: [10.1016/J.SOFTX.2024.101846](https://doi.org/10.1016/J.SOFTX.2024.101846).
- [67] Maximilian Jugl, Sascha Welten, Yongli Mou, Yeliz Ucer Yediel, Oya Deniz Beyan, Ulrich Sax, and Toralf Kirsten. *Privacy-Preserving Linkage of Distributed Datasets using the Personal Health Train*. 2023. arXiv: [2309.06171](https://arxiv.org/abs/2309.06171) [[cs](#).[DC](#)].
- [68] Michael G. Kahn, Tiffany J. Callahan, Juliana Barnard, Alan E. Bauck, Jeff Brown, Bruce N. Davidson, Hossein Estiri, Carsten Goerg, Erin Holve, Steven G. Johnson, Siaw-Teng Liaw, Marianne Hamilton-Lopez, Daniella Meeker, Toan C. Ong, Patrick Ryan, Ning Shang, Nicole G. Weiskopf, Chunhua Weng, Meredith N. Zozus, and Lisa Schilling. ‘A Harmonized Data Quality Assessment Terminology and Framework for the Secondary Use of Electronic Health Record Data’. In: *eGEMs (Generating Evidence & Methods to improve patient outcomes)* 4.1 (1 2016), p. 18. ISSN: 2327-9214. DOI: [10.13063/2327-9214.1244](https://doi.org/10.13063/2327-9214.1244).
- [69] Dimitrios Karapiperis, Aris Gkoulalas-Divanis, and Vassilios S. Verykios. ‘Distance-Aware Encoding of Numerical Values for Privacy-Preserving Record Linkage’. In: *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*. 2017, pp. 135–138. DOI: [10.1109/ICDE.2017.58](https://doi.org/10.1109/ICDE.2017.58).
- [70] Dimitrios Karapiperis and Vassilios S. Verykios. ‘Distributed Near-Optimal LSH-based Framework for Privacy-Preserving Record Linkage’. In: *Computer Science and Information Systems* 11.2 (2 2014), pp. 745–763. DOI: [10.2298/CSIS140215040K](https://doi.org/10.2298/CSIS140215040K).
- [71] Abel N. Kho, John P. Cashy, Kathryn L. Jackson, Adam R. Pah, Satyender Goel, Jörn Boehnke, John Eric Humphries, Scott Duke Kominers, Bala Hota, Shannon A. Sims, Bradley Malin, Dustin D. French, Theresa L. Walunas, David Meltzer, Erin O. Kaleba, Roderick C. Jones, and William L. Galanter. ‘Design and Implementation of a Privacy Preserving Electronic Health Record Linkage Tool in Chicago’. In: *Journal of the American Medical Informatics Association* 22.5 (5 2015), pp. 1072–1080. DOI: [10.1093/jamia/ocv038](https://doi.org/10.1093/jamia/ocv038).
- [72] Hanna Köpcke, Andreas Thor, and Erhard Rahm. ‘Evaluation of entity resolution approaches on real-world match problems’. In: *Proceedings of the VLDB Endowment* 3.1-2 (2010), pp. 484–493. DOI: [10.14778/1920841.1920904](https://doi.org/10.14778/1920841.1920904).
- [73] Hye-Chung Kum, Ashok Krishnamurthy, Ashwin Machanavajjhala, Michael K Reiter, and Stanley Ahalt. ‘Privacy preserving interactive record linkage (PPIRL)’. In: *Journal of the American Medical Informatics Association* 21.2 (2014), pp. 212–220. DOI: [10.1136/amiajnl-2013-002165](https://doi.org/10.1136/amiajnl-2013-002165).

- [74] Hye-Chung Kum, Eric Ragan, Mahin Ramezani, Theodoros Giannouchos, Gurudev Ilangoan, Qinbo Li, Adam D'Souza, Elmer V. Bernstam, Jeffrey R. Curtis, Alva O. Ferdinand, and Cason Schmit. 'Privacy-by-Design: Case Studies in Interactive Record Linkage Using a Hybrid Human-Computer System'. In: *International Journal of Medical Informatics* 204 (2025), p. 106045. DOI: [10.1016/j.ijmedinf.2025.106045](https://doi.org/10.1016/j.ijmedinf.2025.106045).
- [75] Hye-Chung Kum, Eric D. Ragan, Gurudev Ilangoan, Mahin Ramezani, Qinbo Li, and Cason Schmit. 'Enhancing privacy through an interactive on-demand incremental information disclosure interface: applying privacy-by-design to record linkage'. In: *Proceedings of the Fifteenth USENIX Conference on Usable Privacy and Security (SOUPS)*. DOI:[10.5555/3361476.3361489](https://doi.org/10.5555/3361476.3361489). Santa Clara, CA, USA: USENIX Association, 2019, pp. 175–189.
- [76] Tobias Kussel, Torben Brenner, Galina Tremper, Josef Schepers, Martin Lablans, and Kay Hamacher. 'Record Linkage Based Patient Intersection Cardinality for Rare Disease Studies Using Mainzelliste and Secure Multi-Party Computation'. In: *Journal of Translational Medicine* 8.20(1) (2022), :458. DOI: [10.1186/s12967-022-03671-6](https://doi.org/10.1186/s12967-022-03671-6).
- [77] Martin Lablans, Andreas Borg, and Frank Ückert. 'A RESTful Interface to Pseudonymization Services in Modern Web Applications'. In: *BMC Medical Informatics and Decision Making* 15.1 (1 2015), pp. 1–10. DOI: [10.1186/s12911-014-0123-5](https://doi.org/10.1186/s12911-014-0123-5).
- [78] Peeter Laud and Alisa Pankova. 'Privacy-Preserving Record Linkage in Large Databases Using Secure Multiparty Computation'. In: *BMC Medical Genomics* 11.S4 (S4 2018), p. 84. DOI: [10.1186/s12920-018-0400-8](https://doi.org/10.1186/s12920-018-0400-8).
- [79] Ibrahim Lazrig, Toan C. Ong, Indrajit Ray, Indrakshi Ray, Xiaoqian Jiang, and Jaideep Vaidya. 'Privacy Preserving Probabilistic Record Linkage Without Trusted Third Party'. In: *16th Annual Conference on Privacy, Security and Trust (PST)*. 2018, pp. 1–10. DOI: [10.1109/PST.2018.8514192](https://doi.org/10.1109/PST.2018.8514192).
- [80] Qinbo Li, Adam G. D'Souza, Cason Schmit, and Hye-Chung Kum. *Increasing Transparent and Accountable Use of Data by Quantifying the Actual Privacy Risk in Interactive Record Linkage*. 2019. arXiv: [1906.03345](https://arxiv.org/abs/1906.03345).
- [81] Tiancheng Li and Ninghui Li. 'On the Tradeoff between Privacy and Utility in Data Publishing'. In: *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '09. New York, NY, USA: Association for Computing Machinery, 2009, pp. 517–526. DOI: [10.1145/1557019.1557079](https://doi.org/10.1145/1557019.1557079).
- [82] Shurong Lin and Eric Kolaczyk. 'Data Privacy for Record Linkage and Beyond'. In: *NBER Working Paper* 32940 (2024). DOI: [10.3386/w32940](https://doi.org/10.3386/w32940).

- [83] Robin Linacre, Sam Lindsay, Theodore Manassis, Zoe Slade, Tom Hepworth, Ross Kennedy, and Andrew Bond. ‘Splink: Free software for probabilistic record linkage at scale.’ In: *International Journal of Population Data Science* 7.3 (2022). DOI: [10.23889/ijpds.v7i3.1794](https://doi.org/10.23889/ijpds.v7i3.1794).
- [84] Yehuda Lindell and Benny Pinkas. ‘Secure Multiparty Computation for Privacy-Preserving Data Mining’. In: *Journal of Privacy and Confidentiality* 1.1 (2009), pp. 59–98. DOI: [10.29012/jpc.v1i1.566](https://doi.org/10.29012/jpc.v1i1.566).
- [85] Melissa A. Little, Gloribel Bonilla, Timothy McMurry, Kinsey Pebley, Robert C. Klesges, and G. Wayne Talcott. ‘The Feasibility of Using Self-Generated Identification Codes in Longitudinal Research With Military Personnel’. In: *Evaluation & the Health Professions* 45.4 (2022), pp. 354–361. DOI: [10.1177/01632787211031625](https://doi.org/10.1177/01632787211031625).
- [86] Ronan A. Lyons, Kerina H. Jones, Gareth John, Caroline J. Brooks, Jean-Philippe Verplancke, David V. Ford, Ginevra Brown, and Ken Leake. ‘The SAIL Databank: Linking Multiple Health and Social Care Datasets’. In: *BMC medical informatics and decision making* 9.3 (2009). DOI: [10.1186/1472-6947-9-3](https://doi.org/10.1186/1472-6947-9-3).
- [87] Stefanie March, Silke Andrich, Johannes Drepper, Dirk Horenkamp-Sonntag, Andrea Icks, Peter Ihle, Joachim Kieschke, Bianca Kollhorst, Birga Maier, Ingo Meyer, Gabriele Müller, Christoph Ohlmeier, Dirk Peschke, Adrian Richter, Marie-luise Rosenbusch, Nadine Scholten, Mandy Schulz, Christoph Stallmann, Enno Swart, Stefanie Wobbe-Ribinski, Antke Wolter, Jan Zeidler, and Falk Hoffmann. ‘Gute Praxis Datenlinkage (GPD) [Good Practice Data Linkage]’. In: *Das Gesundheitswesen* 81.08/09 (2019), pp. 636–650. DOI: [10.1055/a-0962-9933](https://doi.org/10.1055/a-0962-9933).
- [88] Keith Marsolo, Daniel Kiernan, Sengwee Toh, Jasmin Phua, Darcy Louzao, Kevin Haynes, Mark Weiner, Francisco Angulo, Charles Bailey, Jiang Bian, Daniel Fort, Shaun Grannis, Ashok Kumar Krishnamurthy, Vinit Nair, Pedro Rivera, Jonathan Silverstein, Maryan Zirkle, and Thomas Carton. ‘Assessing the Impact of Privacy-Preserving Record Linkage on Record Overlap and Patient Demographic and Clinical Characteristics in PCORnet®, the National Patient-Centered Clinical Research Network’. In: *Journal of the American Medical Informatics Association: JAMIA* 30.3 (2023), pp. 447–455. DOI: [10.1093/jamia/ocac229](https://doi.org/10.1093/jamia/ocac229).
- [89] Taskforce Datenschutz der Medizininformatik-Initiative [Data protection task force of the Medical Informatics Initiative Germany]. *Übergreifendes Datenschutzkonzept der Medizininformatik-Initiative [Comprehensive Data Protection Concept of the Medical Informatics Initiative]*. 1.0. 2021.
- [90] Daniel Morales, Isaac Agudo, and Javier Lopez. ‘Private Set Intersection: A Systematic Literature Review’. In: *Computer Science Review* 49 (2023), p. 100567. DOI: [10.1016/j.cosrev.2023.100567](https://doi.org/10.1016/j.cosrev.2023.100567).

-
- [91] NFDI4Health. ‘White Paper - Verbesserung des Record Linkage für die Gesundheitsforschung in Deutschland’. In: *NFDI4Health* (2023). DOI: [10.4126/FRL01-006461895](https://doi.org/10.4126/FRL01-006461895).
- [92] Frank Niedermeyer, Simone Steinmetzer, Martin Kroll, and Rainer Schnell. ‘Cryptanalysis of Basic Bloom Filters Used for Privacy Preserving Record Linkage’. In: *Journal of Network and Computer Applications* 6.2 (2014), pp. 59–79. DOI: [10.29012/jpc.v6i2.640](https://doi.org/10.29012/jpc.v6i2.640).
- [93] Konstantinos Nikoletos, George Papadakis, and Manolis Koubarakis. ‘pyJedAI: a Lightsaber for Link Discovery.’ In: *ISWC (Posters/Demos/Industry)*. 2022. DOI: [10.13140/RG.2.2.12171.16160](https://doi.org/10.13140/RG.2.2.12171.16160).
- [94] Thiago Nóbrega, Carlos Eduardo S. Pires, Dimas Cassimiro Nascimento, and Leandro Balby Marinho. ‘Towards Automatic Privacy-Preserving Record Linkage: A Transfer Learning Based Classification Step’. In: *Data & Knowledge Engineering* 145 (May 2023), p. 102180. DOI: [10.1016/j.datak.2023.102180](https://doi.org/10.1016/j.datak.2023.102180).
- [95] Margaret King Odell and Robert Bruce Russell. ‘The soundex coding system’. Patent 1261167 (US). 1918.
- [96] Quita Olsen, Amalie Dyda, Leanna Woods, Elton Lobo, Rebekah Eden, Michelle A. Krahe, Bernadette Richards, Nalini Pather, Lesley McGee, Clair Sullivan, and Jason D. Pole. ‘Worldwide Willingness to Share Health Data High but Privacy, Consent and Transparency Paramount, a Meta-Analysis’. In: *npj Digital Medicine* 8.1 (2025), p. 540. DOI: [10.1038/s41746-025-01868-9](https://doi.org/10.1038/s41746-025-01868-9).
- [97] Toan C Ong, Andrew Hill, Michael G Kahn, Lauren R Lembcke, Lisa M Schilling, and Shaun J Grannis. ‘Linkability Measures to Assess the Data Characteristics for Record Linkage’. In: *Journal of the American Medical Informatics Association* 31.11 (2024), pp. 2651–2659. DOI: [10.1093/jamia/ocae248](https://doi.org/10.1093/jamia/ocae248).
- [98] Fabian Panse, André Düjon, Wolfram Wingerath, and Benjamin Wollmer. ‘Generating Realistic Test Datasets for Duplicate Detection at Scale Using Historical Voter Data’. In: *Proceedings of the 24th International Conference on Extending Database Technology*. 2021, pp. 580–581. DOI: [10.5441/002/edbt.2021.67](https://doi.org/10.5441/002/edbt.2021.67).
- [99] George Papadakis, Ekaterini Ioannou, Emanouil Thanos, and Themis Palpanas. *The Four Generations of Entity Resolution*. Vol. 16. Springer, Cham, 2021. DOI: [10.1007/978-3-031-01878-7](https://doi.org/10.1007/978-3-031-01878-7).
- [100] George Papadakis, Nishadi Kirielle, Peter Christen, and Themis Palpanas. ‘A Critical Re-evaluation of Record Linkage Benchmarks for Learning-Based Matching Algorithms’. In: *IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, 2024, pp. 3435–3448. DOI: [10.1109/ICDE60146.2024.00265](https://doi.org/10.1109/ICDE60146.2024.00265).

- [101] Klaus Pommerening, Johannes Drepper, Krister Helbing, and Thomas Ganslandt. *Leitfaden zum Datenschutz in medizinischen Forschungsprojekten: Generische Lösungen der TMF 2.0 [Guidelines for data protection in medical research coordinated with the data protection officers of the federal and state governments. Generic solutions of TMF 2.0.]* Vol. 11. 2.0. Medizinisch Wissenschaftliche Verlagsgesellschaft, 2014. ISBN: 978-3-95466-295-1.
- [102] Fabian Prasser, Johanna Eicher, Helmut Spengler, Raffael Bild, and Klaus A. Kuhn. ‘Flexible Data Anonymization Using ARX—Current Status and Challenges Ahead’. In: *Journal of Software: Practice and Experience* 50.7 (2020), pp. 1277–1304. DOI: [10.1002/spe.2812](https://doi.org/10.1002/spe.2812).
- [103] Anna Primpeli and Christian Bizer. ‘Profiling Entity Matching Benchmark Tasks’. In: *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. CIKM ’20. Virtual Event, Ireland: Association for Computing Machinery, 2020, pp. 3101–3108. DOI: [10.1145/3340531.3412781](https://doi.org/10.1145/3340531.3412781).
- [104] Anna Primpeli, Christian Bizer, and Margret Keuper. ‘Unsupervised Bootstrapping of Active Learning for Entity Resolution’. In: *The Semantic Web. ESCW 2020* 12123 (2020), pp. 215–231. DOI: [10.1007/978-3-030-49461-2_13](https://doi.org/10.1007/978-3-030-49461-2_13).
- [105] Hans-Ulrich Prokosch, Thomas Bahls, Martin Bialke, Jürgen Eils, Christian Fegeler, Julian Gruendner, Birger Haarbrandt, Christopher Hampf, Wolfgang Hoffmann, Hauke Hund, Marvin Kampf, Lorenz A. Kapsner, Piotr Kasprzak, Oliver Kohlbacher, Dagmar Krefting, Jonathan M. Mang, Michael Marschollek, Sebastian Mate, Armin Müller, Fabian Prasser, Julian Sass, Sebastian Semler, Holger Stenzhorn, Sylvia Thun, Sven Zenker, and Roland Eils. ‘The COVID-19 Data Exchange Platform of the German University Medicine’. In: *Challenges of Trustable AI and Added-Value on Health*. IOS Press, 2022, pp. 674–678. DOI: [10.3233/SHTI220554](https://doi.org/10.3233/SHTI220554).
- [106] Vartika Puri, Shelly Sachdeva, and Parmeet Kaur. ‘Privacy Preserving Publication of Relational and Transaction Data: Survey on the Anonymization of Patient Data’. In: *Computer Science Review* 32 (2019), pp. 45–61. DOI: [10.1016/j.cosrev.2019.02.001](https://doi.org/10.1016/j.cosrev.2019.02.001).
- [107] Dorian Pyle. *Data Preparation for Data Mining*. Morgan Kaufmann Publishers Inc., San Francisco, CA, United States, 1999. ISBN: 978-1-55860-529-9.
- [108] Xuedi Qinl, Chengliang Chai, Nan Tang, Jian Li, Yuyu Luo, Guoliang Li, and Yaoyu Zhu. ‘Synthesizing Privacy Preserving Entity Resolution Datasets’. In: *IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 2022, pp. 2359–2371. DOI: [10.1109/ICDE53745.2022.00222](https://doi.org/10.1109/ICDE53745.2022.00222).

-
- [109] Eric D. Ragan, Hye-Chung Kum, Gurudev Ilangoan, and Han Wang. ‘Balancing Privacy and Information Disclosure in Interactive Record Linkage with Visual Masking’. In: *CHI ’18: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI ’18. Montreal QC, Canada: Association for Computing Machinery, 2018, pp. 1–12. DOI: [10.1145/3173574.3173900](https://doi.org/10.1145/3173574.3173900).
- [110] Erhard Rahm and Hong Hai Do. ‘Data Cleaning: Problems and Current Approaches.’ In: *IEEE Data Engineering Bulletin* 23 (2000), pp. 3–13.
- [111] Thilina Ranbaduge. ‘A Scalable Blocking Framework for Multidatabase Privacy-preserving Record Linkage’. PhD thesis. Australian National University, Australia, 2018.
- [112] Thilina Ranbaduge and Peter Christen. ‘Privacy-Preserving Temporal Record Linkage’. In: *IEEE International Conference on Data Mining (ICDM)*. IEEE, 2018, pp. 377–386. DOI: [10.1109/ICDM.2018.00053](https://doi.org/10.1109/ICDM.2018.00053).
- [113] Thilina Ranbaduge, Peter Christen, and Rainer Schnell. ‘Secure and accurate two-step hash encoding for privacy-preserving record linkage’. In: *Advances in Knowledge Discovery and Data Mining: 24th Pacific-Asia Conference, PAKDD 2020, Singapore, May 11–14, 2020, Proceedings, Part II* 24. Vol. 12085. Springer, Cham, 2020, pp. 139–151. DOI: [10.1007/978-3-030-47436-2_11](https://doi.org/10.1007/978-3-030-47436-2_11).
- [114] Thilina Ranbaduge, Dinusha Vatsalan, and Ming Ding. ‘Privacy-Preserving Deep Learning Based Record Linkage’. In: *IEEE Transactions on Knowledge and Data Engineering* 36.11 (2024), pp. 6839–6850. DOI: [10.1109/TKDE.2023.3342757](https://doi.org/10.1109/TKDE.2023.3342757).
- [115] Sean Randall, Adrian Brown, Anna Ferrante, James Boyd, and Suzanne Robinson. ‘Implementing Privacy Preserving Record Linkage: Insights from Australian Use Cases’. In: *International Journal of Medical Informatics* 191 (2024), p. 105582. ISSN: 13865056. DOI: [10.1016/j.ijmedinf.2024.105582](https://doi.org/10.1016/j.ijmedinf.2024.105582).
- [116] Sean M Randall, Anna M Ferrante, James H Boyd, and James B Semmens. ‘The Effect of Data Cleaning on Record Linkage Quality’. In: *BMC Medical Informatics and Decision Making* 13.64 (2013). DOI: [10.1186/1472-6947-13-64](https://doi.org/10.1186/1472-6947-13-64).
- [117] Sean M. Randall, Adrian P. Brown, Anna M. Ferrante, and James H. Boyd. ‘Privacy Preserving Linkage Using Multiple Dynamic Match Keys’. In: *International Journal of Population Data Science* 4.1 (2019). DOI: [10.23889/ijpds.v4i1.1094](https://doi.org/10.23889/ijpds.v4i1.1094).
- [118] Gesine Richter, Nourane Trigui, Amke Caliebe, and Michael Krawczak. ‘Attitude towards Consent-Free Research Use of Personal Medical Data in the General German Population’. In: *Heliyon* 10.6 (2024), e27933. DOI: [10.1016/j.heliyon.2024.e27933](https://doi.org/10.1016/j.heliyon.2024.e27933).

- [119] **Florens Rohde**, Victor Christen, Martin Franke, and Erhard Rahm. ‘Multi-Layer Privacy-Preserving Record Linkage with Clerical Review based on gradual information disclosure’. In: *BTW 2025*. Bamberg, Germany: Gesellschaft für Informatik e.V., 2025. DOI: [10.18420/BTW2025-18](https://doi.org/10.18420/BTW2025-18).
- [120] **Florens Rohde**, Victor Christen, and Erhard Rahm. ‘Exploring Privacy-Preserving Record Linkage: A Holistic Framework for Dataset Generation and Detailed Result Analysis’. In: *VLDB 2025 Workshop: 14th International Workshop on Quality in Databases (QDB’25)*. London, 2025.
- [121] **Florens Rohde**, Victor Christen, and Erhard Rahm. ‘SecUREmatch: Integrating Clerical Review in Privacy-Preserving Record Linkage’. In: *SIGMOD-Companion ’25*. Berlin, Germany, 2025. DOI: [10.1145/3722212.3725131](https://doi.org/10.1145/3722212.3725131).
- [122] **Florens Rohde**, Martin Franke, Victor Christen, and Erhard Rahm. ‘Value-specific Weighting for Record-level Encodings in Privacy-Preserving Record Linkage’. In: *BTW 2023*. Dresden, Germany: Gesellschaft für Informatik e.V., 2023. DOI: [10.18420/BTW2023-21](https://doi.org/10.18420/BTW2023-21).
- [123] **Florens Rohde**, Martin Franke, Ziad Sehili, Martin Lablans, and Erhard Rahm. ‘Optimization of the Mainzelliste software for fast privacy-preserving record linkage’. In: *Journal of Translational Medicine* 19.33 (2021). DOI: [10.1186/s12967-020-02678-1](https://doi.org/10.1186/s12967-020-02678-1).
- [124] Alieh Saeedi, Markus Nentwig, Eric Peukert, and Erhard Rahm. ‘Scalable Matching and Clustering of Entities with FAMER’. In: *Complex Systems Informatics and Modeling Quarterly* 16 (16 2018), pp. 61–83. DOI: [10.7250/csimq.2018-16.04](https://doi.org/10.7250/csimq.2018-16.04).
- [125] Alieh Saeedi, Eric Peukert, and Erhard Rahm. ‘Using Link Features for Entity Clustering in Knowledge Graphs’. In: *The Semantic Web*, ed. by Aldo Gangemi et al. Vol. 10843. ESWC. Cham: Springer, Charm, 2018. DOI: [10.1007/978-3-319-93417-4_37](https://doi.org/10.1007/978-3-319-93417-4_37).
- [126] Ravi S. Sandhu and Pierangela Samarati. ‘Access control: principle and practice’. In: *IEEE Communications Magazine* 32.9 (1994), pp. 40–48. DOI: [10.1109/35.312842](https://doi.org/10.1109/35.312842).
- [127] Rishi Saripalle, Christopher Runyan, and Mitchell Russell. ‘Using HL7 FHIR to Achieve Interoperability in Patient Health Record’. In: *Journal of Biomedical Informatics* 94 (2019), p. 103188. DOI: [10.1016/j.jbi.2019.103188](https://doi.org/10.1016/j.jbi.2019.103188).
- [128] Rainer Schnell, Tobias Bachteler, and Jörg Reiher. ‘A Novel Error-Tolerant Anonymous Linking Code’. In: *German RLC Working Paper* (2011). DOI: [10.2139/ssrn.3549247](https://doi.org/10.2139/ssrn.3549247).

- [129] Rainer Schnell, Tobias Bachteler, and Jörg Reiher. ‘Improving the Use of Self-Generated Identification Codes’. In: *Evaluation Review* 34.5 (2010), pp. 391–418. DOI: [10.1177/0193841X10387576](https://doi.org/10.1177/0193841X10387576).
- [130] Rainer Schnell, Tobias Bachteler, and Jörg Reiher. ‘Privacy-preserving record linkage using Bloom filters’. In: *BMC Med. Inf. & Decision Making* 25.9 (2009), p. 41. DOI: [10.1186/1472-6947-9-41](https://doi.org/10.1186/1472-6947-9-41).
- [131] Rainer Schnell and Christian Borgs. ‘Encoding Hierarchical Classification Codes for Privacy-Preserving Record Linkage Using Bloom Filters’. In: *Machine Learning and Knowledge Discovery in Databases*, ed. by Peggy Cellier et al. Vol. 1168. Springer, Cham, 2020, pp. 142–156. DOI: [10.1007/978-3-030-43887-6_12](https://doi.org/10.1007/978-3-030-43887-6_12).
- [132] Rainer Schnell and Christian Borgs. ‘Randomized Response and Balanced Bloom Filters for Privacy Preserving Record Linkage’. In: *IEEE 16th International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2016, pp. 218–224. DOI: [10.1109/ICDMW.2016.0038](https://doi.org/10.1109/ICDMW.2016.0038).
- [133] Rainer Schnell and Christian Borgs. ‘XOR-Folding for Bloom Filter-based Encryptions for Privacy-preserving Record Linkage’. In: *Working Paper WP-GRLC-2016-03, German Record Linkage Center, Nuremberg* WP-GRLC-2016-03 (2016). DOI: [10.2139/ssrn.3527984](https://doi.org/10.2139/ssrn.3527984).
- [134] Rainer Schnell, Dorothea Rukasz, Christian Borgs, Julian Reinhold, Severin Weiland, Stefan Brumme (HMAC, SHA256), William B. Brogden (Metaphone), Tim O’Brien (Metaphone), Stephen Lacy (Double Metaphone), and Apache Software Foundation. *R package PPRL*. 2025. DOI: <https://doi.org/10.32614/CRAN.package.PPRL>. URL: <https://cran.r-project.org/package=PPRL> (visited on 12/17/2025).
- [135] Ziad Sehili, Lars Kolb, Christian Borgs, Rainer Schnell, and Erhard Rahm. ‘Privacy Preserving Record Linkage with PPJoin’. In: *Proceedings of 16. GI-Fachtagung für Datenbanksysteme in Business, Technologie und Web (BTW2015)* (2015).
- [136] Ziad Sehili, **Florens Rohde**, Martin Franke, and Erhard Rahm. *Multi-Party Privacy Preserving Record Linkage in Dynamic Metric Space*. BTW 2021. Streaming, Dresden, Germany, 2021. DOI: [10.18420/btw2021-13](https://doi.org/10.18420/btw2021-13).
- [137] Sebastian Semler, Frank Wissing, and Ralf Heyder. ‘German Medical Informatics Initiative’. In: *Methods of Information in Medicine* 57.S 01 (S 01 2018), e50–e56. DOI: [10.3414/me18-03-0003](https://doi.org/10.3414/me18-03-0003).
- [138] Adi Shamir. ‘How to share a secret’. In: *Communications of the ACM* 22.11 (1979), pp. 612–613.

- [139] Duncan Smith. ‘Secure pseudonymisation for privacy-preserving probabilistic record linkage’. In: *Journal of Information Security and Applications* 34.262608 (2017), pp. 271–279. ISSN: 22142126. DOI: [10.1016/j.jisa.2017.01.002](https://doi.org/10.1016/j.jisa.2017.01.002).
- [140] Chakkrit Snae. ‘A Comparison and Analysis of Name Matching Algorithms’. In: *International Journal of Computer and Information Engineering* 1.1 (2007), pp. 107–112. ISSN: eISSN: 1307-6892.
- [141] Sebastian Stammmler, Tobias Kussel, Phillipp Schoppmann, Florian Stampe, Galina Tremper, Stefan Katzenbeisser, Kay Hamacher, and Martin Lablans. ‘Mainzelliste SecureEpiLinker (MainSEL): Privacy-Preserving Record Linkage using Secure Multi-Party Computation’. In: *Bioinformatics* 38.6 (2020), ed. by Zhiyong Lu, pp. 1657–16668. DOI: [10.1093/bioinformatics/btaa764](https://doi.org/10.1093/bioinformatics/btaa764).
- [142] S. Joshua Swamidass, Matthew Matlock, and Leon Rozenblit. ‘Securely measuring the overlap between private datasets with cryptosets’. In: *PLOS ONE* 10.2 (2015), pp. 1–18. ISSN: 19326203. DOI: [10.1371/journal.pone.0117898](https://doi.org/10.1371/journal.pone.0117898).
- [143] Latanya Sweeney. ‘k-anonymity: A model for protecting privacy’. In: *International journal of uncertainty, fuzziness and knowledge-based systems* 10.05 (2002), pp. 557–570. DOI: <https://doi.org/10.1142/S0218488502001648>.
- [144] Galina Tremper, Torben Brenner, Moanes Ben Amor, Tobias Kussel, and Martin Lablans. ‘Mainzelliste: Ten Years of Pseudonymization, Record Linkage, and Informed Consent Management’. In: *Patterns* 7.1 (2026), p. 101432. DOI: [10.1016/j.patter.2025.101432](https://doi.org/10.1016/j.patter.2025.101432).
- [145] Jianhong Tu, Ju Fan, Nan Tang, Peng Wang, Guoliang Li, Xiaoyong Du, Xiaofeng Jia, and Song Gao. ‘Unicorn: A Unified Multi-tasking Model for Supporting Matching Tasks in Data Integration’. In: *Proceedings of the ACM on Management of Data* 1.1 (2023). DOI: [10.1145/3588938](https://doi.org/10.1145/3588938).
- [146] Dinusha Vatsalan and Peter Christen. ‘An iterative two-party protocol for scalable privacy-preserving record linkage’. In: *Proceedings of the Tenth Australasian Data Mining Conference - Volume 134*. AusDM ’12. Sydney, Australia: Australian Computer Society, Inc., 2012, pp. 127–138. DOI: [10.5555/2525373.2525389](https://doi.org/10.5555/2525373.2525389).
- [147] Dinusha Vatsalan, Peter Christen, Christine M. O’Keefe, and Vassilios S. Verykios. ‘An Evaluation Framework for Privacy-Preserving Record Linkage’. In: *Jorunal for Privacy and Confidentiallity* 6.1 (2014), pp. 35–75. DOI: <https://doi.org/10.29012/jpc.v6i1.636>.
- [148] Dinusha Vatsalan, Peter Christen, and Vassilios S. Verykios. ‘A Taxonomy of Privacy-Preserving Record Linkage Techniques’. In: *Information Systems* 38.6 (2013), pp. 946–969. DOI: [10.1016/j.is.2012.11.005](https://doi.org/10.1016/j.is.2012.11.005).

- [149] Dinusha Vatsalan, Peter Christen, and Vassilios S. Verykios. ‘An efficient two-party protocol for approximate matching in private record linkage’. In: *Proceedings of the Ninth Australasian Data Mining Conference - Volume 121*. AusDM ’11. Ballarat, Australia: Australian Computer Society, Inc., 2011, pp. 125–136. DOI: [10.5555/2483628.2483644](https://doi.org/10.5555/2483628.2483644).
- [150] Dinusha Vatsalan, Ziad Sehili, Peter Christen, and Erhard Rahm. ‘Privacy-Preserving Record Linkage for Big Data: Current Approaches and Research Challenges’. In: *Handbook of Big Data Technologies*. Springer, Cham, 2017, pp. 851–895. DOI: [10.1007/978-3-319-49340-4_25](https://doi.org/10.1007/978-3-319-49340-4_25).
- [151] Anushka Vidanage, Peter Christen, Thilina Ranbaduge, and Rainer Schnell. ‘A Graph Matching Attack on Privacy-Preserving Record Linkage’. In: *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. CIKM ’20. Virtual Event, Ireland: Association for Computing Machinery, 2020, pp. 1485–1494. DOI: [10.1145/3340531.3411931](https://doi.org/10.1145/3340531.3411931).
- [152] Anushka Vidanage, Peter Christen, Thilina Ranbaduge, and Rainer Schnell. ‘A Vulnerability Assessment Framework for Privacy-preserving Record Linkage’. In: *ACM Transactions on Privacy and Security* 26.3 (2023). ISSN: 2471-2566. DOI: [10.1145/3589641](https://doi.org/10.1145/3589641).
- [153] Anushka Vidanage, Thilina Ranbaduge, Peter Christen, and Sean M Randall. ‘A Privacy Attack on Multiple Dynamic Match-key Based Privacy-Preserving Record Linkage’. In: *International Journal of Population Data Science* 5.1 (2020). DOI: [10.23889/ijpds.v5i1.1345](https://doi.org/10.23889/ijpds.v5i1.1345).
- [154] Anushka Vidanage, Thilina Ranbaduge, Peter Christen, and Rainer Schnell. ‘A Taxonomy of Attacks on Privacy-Preserving Record Linkage’. In: *Journal of Privacy and Confidentiality* 12.1 (2022). DOI: [10.29012/jpc.764](https://doi.org/10.29012/jpc.764).
- [155] Nicole Gray Weiskopf and Chunhua Weng. ‘Methods and Dimensions of Electronic Health Record Data Quality Assessment: Enabling Reuse for Clinical Research’. In: *Journal of the American Medical Informatics Association* 20.1 (2013), pp. 144–151. DOI: [10.1136/amiajnl-2011-000681](https://doi.org/10.1136/amiajnl-2011-000681).
- [156] Sascha Welten, Yongli Mou, Laurenz Neumann, Mehrshad Jaberansary, Yeliz Yediel Ucer, Toralf Kirsten, Stefan Decker, and Oya Beyan. ‘A Privacy-Preserving Distributed Analytics Platform for Health Care Data’. In: *Methods of Information in Medicine* 61.S01 (2022), e1–e11. DOI: [10.1055/s-0041-1740564](https://doi.org/10.1055/s-0041-1740564).
- [157] *What Is Privacy-Preserving Record Linkage (PPRL) and Why Does It Matter?* Data-vant. 2023. URL: <https://www.datavant.com/blog/privacy-preserving-record-linkage> (visited on 01/19/2026).

- [158] Rebecca Wilson, Oliver Butters, Demetris Avraam, James Baker, Jonathan Tedds, Andrew Turner, Madeleine Murtagh, and Paul Burton. ‘DataSHIELD – New Directions and Dimensions’. In: *Data Science Journal* 16 (0 2017), p. 21. DOI: [10.5334/dsj-2017-021](https://doi.org/10.5334/dsj-2017-021).
- [159] William E. Winkler and Yves Thibaudeau. *An application of the Fellegi-Sunter model of Record Linkage to the 1990 U.S. decennial census*. Tech. Rep. RR1991/09. Washington, DC: US Bureau of the Census, 1991.
- [160] Huiping Xu, Xiaochun Li, Changyu Shen, Siu L. Hui, and Shaun Grannis. ‘Incorporating Conditional Dependence in Latent Class Models for Probabilistic Record Linkage: Does It Matter?’ In: *The Annals of Applied Statistics* 13.3 (2019). DOI: [10.1214/19-AOAS1256](https://doi.org/10.1214/19-AOAS1256).
- [161] William E Yancey. *Evaluating String Comparator Performance for Record Linkage*. Technical report Statistics 2005/05. U.S. Census Bureau, 2005.
- [162] Yixiang Yao, Weizhao Jin, and Srivatsan Ravi. ‘Labeling without Seeing? Blind Annotation for Privacy-Preserving Entity Resolution’. In: *Transactions on Machine Learning Research* (2025). ISSN: 2835-8856.
- [163] Vivienne J. Zhu, Marc J. Overhage, James Egg, Stephen M. Downs, and Shaun J. Grannis. ‘An Empiric Modification to the Probabilistic Record Linkage Algorithm Using Frequency-Based Weight Scaling’. In: *Journal of the American Medical Informatics Association* 16.5 (2009), pp. 738–745. DOI: [10.1197/jamia.M3186](https://doi.org/10.1197/jamia.M3186).
- [164] Sumayya Ziyad, Peter Christen, Anushka Vidanage, Charini Nanayakkara, and Rainer Schnell. ‘Privacy-Preserving Record Linkage Using Reference Set Based Encoding: A Single Parameter Method’. In: *Information Systems* 133 (2025), p. 102569. DOI: [10.1016/j.is.2025.102569](https://doi.org/10.1016/j.is.2025.102569).

Bibliografische Angaben

Rohde, Florens: Robust and transparent Privacy-Preserving Record Linkage. Dissertation, Universität Leipzig, 117 Seiten, 2026

Selbständigkeitserklärung

Hiermit erkläre ich, die vorliegende Dissertation selbständig und ohne unzulässige fremde Hilfe angefertigt zu haben. Ich habe keine anderen als die angeführten Quellen und Hilfsmittel benutzt und sämtliche Textstellen, die wörtlich oder sinngemäß aus veröffentlichten oder unveröffentlichten Schriften entnommen wurden, und alle Angaben, die auf mündlichen Auskünften beruhen, als solche kenntlich gemacht. Ebenfalls sind alle von anderen Personen bereitgestellten Materialien oder erbrachten Dienstleistungen als solche gekennzeichnet.

Leipzig, 6. Februar 2026

Florens Rohde