Exploring Privacy-Preserving Record Linkage: A Holistic Framework for Dataset Generation and Detailed Result Analysis

Florens Rohde
ScaDS.AI Dresden/Leipzig, Leipzig University
Leipzig, Germany
rohde@informatik.uni-leipzig.de

Victor Christen
ScaDS.AI Dresden/Leipzig, Leipzig University
Leipzig, Germany
christen@informatik.uni-leipzig.de

Erhard Rahm
ScaDS.AI Dresden/Leipzig, Leipzig University
Leipzig, Germany
rahm@informatik.uni-leipzig.de

ABSTRACT

Privacy-preserving record linkage (PPRL) methods facilitate integration of sensitive data without disclosing plaintext information among data owners or to third parties. However, PPRL techniques are notably affected by problems related to data quality. Their typically rigid matching strategy can deter data custodians from employing them in practical applications due to potential linkage quality issues. In this work, we present a framework for studying PPRL algorithms with respect to their robustness against dataset variation in order to guide data custodians in selecting suitable methods. Our framework offers multiple possibilities to create test datasets for linkage tasks depending on the available input data. Furthermore, the implementation includes a new synthetic data generator for creating realistic population records including common household structures for Germany. At the heart of our contribution lies the creation and tracking of descriptive tags that outline the characteristics of datasets across various levels of granularity. We describe an approach for exploring linkage quality outcomes based on those record (pair) features which enables researchers to better comprehend their linkage results and assess those of others.

VLDB Workshop Reference Format:
Florens Rohde, Victor Christen, and Erhard Rahm. Exploring Privacy-Preserving Record Linkage: A Holistic Framework for Dataset Generation and Detailed Result Analysis. VLDB 2025 Workshop: 14th International Workshop on Quality in Databases (QDB'25).

VLDB Workshop Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/floroh/pprl-goodall.

1 INTRODUCTION

Record linkage, or entity resolution, identifies different representations of the same real-world entity, like a person. It is essential in many data integration tasks with multiple sources, enabling better data analysis or creation of high-quality machine learning training datasets. Unique record identifiers are usually unavailable for join operations, so records are compared pairwise based on their identifying attributes like first name, last name, and birth date, then classified as match or non-match.

Record linkage can impair individual privacy by merging data that might be misused, leading to legal and organizational restrictions [3]. Privacy-preserving record linkage (PPRL) methods facilitate linking without disclosing sensitive plaintext information among data owners or to third parties. To protect the identifying attributes, data owners encode this data before sending it to a semi-trusted linkage unit for matching. Similarity-preserving encodings allow for matching records approximately despite errors or inconsistencies like typographical errors or outdated data. Multiple encoding techniques have been proposed but Bloom filters are particularly popular in both research and applications due to their simplicity and scalability [3, 12].

Attribute-level encodings create exploitable frequency patterns in encoded data, facilitating rather easy reidentification by aligning the most common plaintext values with the most common Bloom filters [28]. Consequently, these encodings are appropriate for data linkages that require a flexible matching strategy to achieve high-quality outcomes and privacy risks are limited. For improved attack resilience, attributes are combined into a single record-level encoding. However, this approach strongly limits the flexibility of the matching algorithm. The linkage unit may only use 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 [8], 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 striving to identify a suitable linkage strategy tailored to their specific use case may encounter a lack of insight for choosing a satisfactory 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.

Proceedings of the VLDB Endowment. ISSN 2150-8097.

Usually, entity resolution algorithms are evaluated based on benchmark datasets from various domains. Nevertheless, considering their particular application involving sensitive information, assessing 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 [18]. 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 [3]. 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.

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 fine-grained 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 work, we present a framework for comprehensive evaluation and quality analysis of privacy-preserving record linkage 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, facilitating a comprehensive comparison and refinement of methods.
- We demonstrate the analytical abilities of our framework using *tag*-based process descriptors on illustrative use cases.

Related work

Over the past few decades, numerous techniques for PPRL have been introduced [3, 12]. Protocols utilizing secure multiparty computation offer formal security guarantees, but they generally have significant computational demands, especially for fuzzy comparisons. Consequently, perturbation-based methodologies are frequently employed in scenarios involving large-scale data linkage. The initial proposal for utilizing Bloom filter encodings in PPRL dates back to 2009 [24]. However, the deterministic encoding leads to frequent bit positions for common plaintext patterns, enabling frequency attacks, especially in attribute-level Bloom filter (ABF) where each attribute is represented by a separate bit vector. Frequent plaintext attribute values often match with frequent Bloom filters which allows to (partially) reconstruct the original records despite the irreversible hash functions. Various hardening methods have been proposed to disguise these patterns [11], including autoencoders [5], balancing [26], and XOR folding [27]. Encodings combining quasi-identifying attributes into a single record-level representation [7, 25] are highly recommended by the literature [28]. By modifying the number of hash functions k, attribute weights can be integrated taking into account their discriminatory power and error rate [22].

There are several tools that employ perturbation-based PPRL methods for either general purposes [10] or specialized medical applications [15, 23]. 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.

On the one hand, a number of data generators for entity resolution tasks exist. Tools like FEBRL [1] create records entirely from scratch. The newly introduced pseudopeople package facilitates the generation of extensive datasets through individual-based modeling of the US population, which also encompasses household structures [14]. Others create a corrupted variant with a variety of error types based on an existing (real-world) input dataset [16]. Another group of data creation tools, such as GeCo [4] and Gecko [17], integrate both the generative and corruptive aspects.

On the other hand, further tools exist to benchmark matching solutions based on profiling test datasets such as [20]. One of the few tools available for understanding record linkage results is Frost/Snowman [13], 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.

2 METHOD

2.1 System overview

The overall architecture of our proposed framework is depicted in Fig. 1. We use the notation $\mathbf{R}$ for datasets containing records of multiple sources. Within a linkage scenario involving I sources, each dataset $\mathbf{R}_i$ denotes a database owned by i so that $\mathbf{R} = \bigcup_{i \in I} \mathbf{R_i}$. This paper 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 prefinal 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.

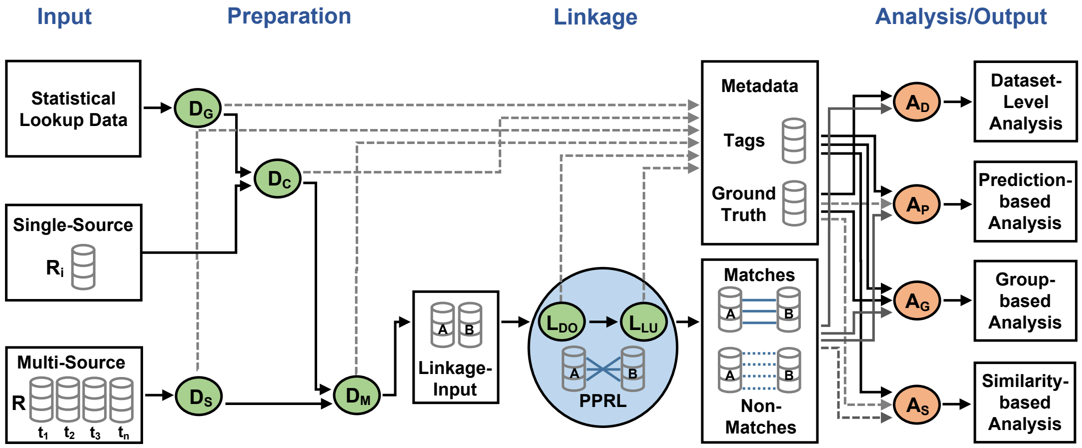


Figure 1: Architecture of our proposed evaluation framework with linkage input creation components on the left side (D_G: Synthetic data generator, D_C: Data corrupter, D_S: Selector from multi-source record clusters, D_M: Linkage problem modifier), the PPRL services in the center (L_{DO}: Encoding, L_{LU}: Matching) and the result analysis components on the right side (A_D: Dataset-level evaluation, A_P: Prediction analysis, A_G: Group analysis, A_S: Similarity analysis).

All those components serve as potential sources of descriptive tags which are gathered in a tag store 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 data-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 matching's blocking stage. 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.

2.2 Tag concept

Tags describe linkage problems based on dataset and outcome characteristics across the dimensions of **G**ranularity, **R**epresentation, **O**rigin, and **T**ype. We denote tags as $\tau_G^R[O](\mathit{LABEL,[string],[number]})$ 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.

Table 1: Examples of tags provided by different components

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

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.

Tab. 1 lists the tag scope and examples from the components in our framework. Our 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.

2.3 Linkage dataset creation

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

2.3.1 D_G: Synthetic dataset generation. Synthesizing realistic data is challenging, as evidenced by ongoing development of new tools. 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. We propose a novel generator tailored for Germany, particularly aimed at tackling the issue of aligning individuals residing within household structures in record linkage. 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.

Individuals generation The generator utilizes German Census data[1] 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, e.g., gender and year of birth influencing the choice of first name. Attributes like name and birthdate, and address are generated from probability distributions.

Household modeling Household structures are generated in four phases, each focusing on a specific household type.

Family Households are generated 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* are generated based on age and partnership type distributions. The generator ensures compatibility between partners by matching demographic attributes. *Single-Person Households* are created using age-based distributions, ensuring alignment with real-world data. *Shared Housings* such as student apartments, are based on grouping individuals with similar demographic attributes. Gaussian distributions are used to model age similarity among housemates.

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.

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

The implementation offers a versatile framework for applying different types of corruption. 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.

2.3.3 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 [3]. Based on the *NCVR* data, Panse et al. [19] provide a database for large-scale record linkage tasks, containing clusters of historical records sharing the same voter ID. We developed a dataset generator enabling the selection of subsets considering the degree of duplicate *dirtyness* or the snapshot time (t) span within these record clusters. Table 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.

Table 2: Example record cluster of a certain individual from temporal snapshots of a voter registry grouped by voter ID.

VoterID	Snapshot t	First name	Middle n.	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

2.3.4 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. The component is part of our implementations of D_C and D_S. However, it may also be utilized separately in combination with an external linkage input creation tool.

[1] https://ergebnisse.zensus2022.de/datenbank/online/

2.4 Linkage protocol 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.

2.4.1 L_{DO}: Encoding at data owners.

The original plaintext is irreversibly converted using cryptographic hash functions. Similarity-preserving encodings such as Bloom filters allow for matching records with data quality issues. The plaintext values are parsed to (overlapping) substrings (q-grams) which are each mapped to positions in a bit vector of fixed size using k keyed hash functions. Identical q-grams map to the same bit positions, so a high overlap of q-grams results in similar Bloom filters and thus enables fuzzy comparison.

Our implementation comprises a variety of such encoding techniques which allows to test the effect on the linkage outcome (see also section 3.3). Furthermore, our 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.

2.4.2 L_{LU}: Privacy-preserving matching.

Encoded records are pairwise compared and classified. Blocking or filtering techniques mitigate the quadratic complexity of comparing every record between sources [2]. Bloom filter similarities are measured using normalized set similarity measures such as the Dice or the Jaccard coefficient. A similarity score above the threshold classifies the record pair as *match*, or as *non-match* otherwise. Due to this rather simple matching approach in PPRL, insights on influencing factors of the encoding technique on the similarity score are very valuable for method development. Thus, apart from tags describing the encoded dataset, $\tau^{E}_{rp}(SIM)$ is provided, encompassing the record pair similarity which is crucial for certain analysis types as outlined below.

The implementation of the components described in the previous sections is based on extensions of the service-oriented PPRL architecture with L_{DO} and L_{LU} in [21]. D_G and D_S are implemented as dedicated web services, enabling configuration-file-based creation and export of linkage datasets as CSV files or JSON objects. These generated datasets or datasets from external sources can be imported into the data owner services in both formats as well.

D_C and D_M are integrated into the data owner service to facilitate in-place experiments based on different data quality scenarios using a single data source as reference. The dataflow between the services is orchestrated by a protocol manager service which enables flexible automated workflows of dataset creation, PPRL protocol execution and result analysis.

2.5 Linkage result analyzer

This section outlines the analytical capabilities of our framework which are implemented as a Python-based Streamlit application using the frameworks' RESTful interfaces for retrieval of the linkage outcome and its descriptive tags. Additionally, CSV export is supported, facilitating customized analysis utilizing external tools.

2.5.1 A_D: Dataset-level measures.

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 [9]. 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.

2.5.2 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 more effectively determine 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 Sec. 3.1.

2.5.3 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. By categorizing elements within the confusion matrix, such as exploring prevalent trends among FP or FN pairs, we can pinpoint systematic deficiencies in the linkage method [6]. Furthermore, we can categorize by match predictions to examine 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.

2.5.4 A_S: Similarity analysis.

This component enables an in-depth examination of the outcomes grounded in τ^{E}_{rp}(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.

3 USE CASES

In this section, we provide illustrative examples of how our framework can be used for studying PPRL methods.

Our framework enables evaluation of PPRL algorithms based on workflow definitions including dataset creation and linkage protocol orchestration. Figure 2 shows an example of such a workflow. First, a synthetic dataset is created and subsequently modified to produce record variants that are either supposed to be *matches* (here: ID 44) or *non-matches* (here: ID 43 and ID 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.

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.

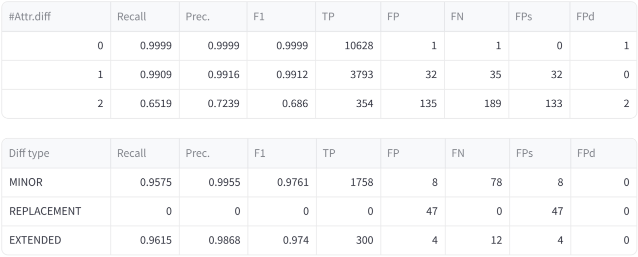
Tag store collects tags created during steps 1-4 (here: only tags related to record 42).

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 2: Application of the experimental framework using the data creation path D_G->D_C: The steps for generating the linkage input datasets, executing the PPRL protocol and subsequent result analysis are illustrated on the left using example records. The table on the right 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.

3.1 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 Fig. 3). 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.



#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: Group-based quality assessment using $\tau^P_{rp}[L_{DO}]$ for the number of differing input attributes (top) and $\tau^P_{ap}[L_{DO}]$ for descriptions of fine-grained corruption types (bottom).

3.2 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 Fig. 4. 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.

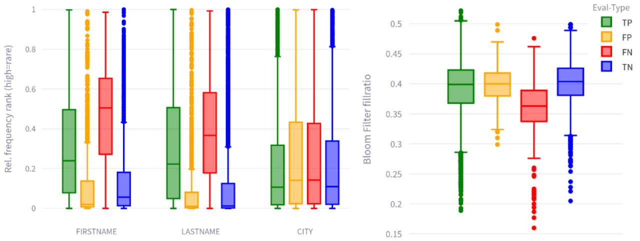


Figure 4: Prediction-based analysis on distributions of attribute frequencies (left, $\tau^P_a[L_{DO}]$) and Bloom filter fill ratios (right, $\tau^E_r[L_{LU}]$) indicates correlation of common names with false positives and of low fill ratios with false negatives.

3.3 Effect of encoding technique on similarity

In perturbation-based PPRL, the encoding choice is crucial for separability of *matches* and *non-matches* based on the record similarity. For studying the suitability of an encoding, we apply A_S on τ^P_{rp} for two data quality scenarios *DIRTY* and *TIME* created by different configurations of D_C, see Fig. 5. The histograms display similarity distributions for various high-level modification types. Results from the *DIRTY* scenario indicate initial separability of *matches* and *non-matches*. The distribution of pairs with exchanged first and last name significantly overlaps with pairs having a similar but typo-free name and a different address. Therefore, a threshold-based classifier predicting *non-match* for these pairs will struggle to detect *matches* with swapped name fields. Data custodians should choose a modified encoding method with relaxed attribute salting to address this issue. The *TIME* scenario output suggests an overly high last name attribute weight, causing false negatives for some pairs with replaced last names.

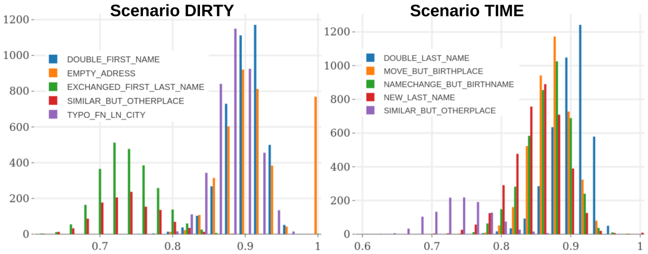


Figure 5: Similarity analysis for different data quality scenarios using $\tau_{rp}^{P}[D_C]$: On the left (*DIRTY*), true matches suffer from missing and erroneous data while on the right (*TIME*), true matches are affected by outdated data.

In Fig. 6, we compare different encoding configurations using histograms of attribute-level Bloom filter similarities, using $\tau_a^P(\text{LEN})$, with and without hardening techniques. Such analysis allows to study possible side-effects of encoding variations which aim primarily at improving the attack resilience. A single threshold as classification model is only feasible if there is low overlap and a common decision boundary between non-matching and matching values across input variations like length.

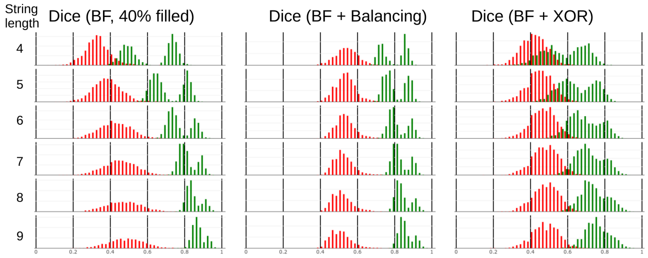


Figure 6: Similarity analysis for comparing encoding techniques using $\tau_a^P[L_{DO}]$: Usage of hardening techniques may lead to improved (middle) or worse (right) separability of non-matches (red) from matches (green) for short input values when compared to plain Bloom Filter encodings (left).

4 DISCUSSION

The variety of dataset creation methods enables choosing the best approach for each use case. Generally, using real-world datasets **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, such as the case of 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.

Our 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 linkage results to the linkage unit module, potentially accompanied by descriptive tags.

The analytical components of our framework 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. Consequently, we seek to augment existing tools for data generation and corruption by incorporating descriptive tags in their outputs, thereby facilitating their integration within our framework. Enhancing transparency on potential sources of linkage bias in PPRL constitutes an initial step towards achieving a more cohesive integration between the data linkage process and the subsequent data analysis. While our work focuses on analyzing the linkage quality, it is worthwhile to extend the framework to study privacy aspects of encodings as well including assessment of risks related to reidentification attacks.

5 CONCLUSION

We introduced a novel 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. We anticipate that gaining a clearer understanding of their possibilities and limitations will enhance confidence in these methods.

In future work, we intend to integrate further linkage benchmark datasets in our selection component for more general use. In addition, we will employ our framework to study adaptive encoding techniques that align the decision boundaries of various groups of pairs.

ACKNOWLEDGMENTS

The authors acknowledge the financial support by the Federal Ministry of Education and Research of Germany and by the Sächsische Staatsministerium für Wissenschaft, Kultur und Tourismus for ScaDS.AI. The German population data generator is based on work by Duc Dung Dao.

REFERENCES

[1] Peter Christen. 2009. Development and user experiences of an open source data cleaning, deduplication and record linkage system. *ACM SIGKDD Explorations Newsletter* 11, 1 (2009), 39–48.

[2] Peter Christen. 2012. *Data Matching*. Springer. https://doi.org/10.1007/978-3-642-31164-2

[3] Peter Christen, Thilina Ranbaduge, and Rainer Schnell. 2020. *Linking Sensitive Data*. Springer, Cham. https://doi.org/10.1007/978-3-030-59706-1

[4] Peter Christen and Dinusha Vatsalan. 2013. Flexible and extensible generation and corruption of personal data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 1165–1168.

[5] Victor Christen, Tim Hantschel, Peter Christen, and Erhard Rahm. 2023. Privacy-Preserving Record Linkage Using Autoencoders. 15, 4 (2023), 347–357. https://doi.org/10.1007/s41060-022-00377-2

[6] James Doidge, Peter Christen, and Katie Harron. 2020. Quality Assessment in Data Linkage. In *Joined up Data in Government: The Future of Data Linking Methods*. UK Government Analysis Function and Office for National Statistics.

[7] Elizabeth A. Durham, Murat Kantarcioglu, Yuan Xue, Csaba Toth, Mehmet Kuzu, and Bradley Malin. 2014. Composite Bloom Filters for Secure Record Linkage. *IEEE Transactions on Knowledge and Data Engineering* 26, 12 (2014), 2956–2968. https://doi.org/10.1109/TKDE.2013.91

[8] Ivan P. Fellegi and Alan B. Sunter. 1969. A Theory for Record Linkage. *J. Amer. Statist. Assoc.* 64, 328 (1969), 1183–1210. https://doi.org/10.1080/01621459.1969.10501049

[9] Martin Franke, Victor Christen, Peter Christen, Florens Rohde, and Erhard Rahm. 2024. (Privately) Estimating Linkage Quality for Record Linkage. In *Proceedings of the 27th International Conference on Extending Database Technology (EDBT)*. https://doi.org/10.48786/EDBT.2024.26

[10] Martin Franke, Ziad Sehili, and Erhard Rahm. 2019. PRIMAT. In *Proceedings of the VLDB Endowment*, Vol. 12. 1826–1829. https://doi.org/10.14778/3352063.3352076

[11] Martin Franke, Ziad Sehili, Florens Rohde, and Erhard Rahm. 2021. Evaluation of Hardening Techniques for Privacy-Preserving Record Linkage. In *Proceedings of the 24th International Conference on Extending Database Technology*. 289–300. https://doi.org/10.5441/002/edbt.2021.26

[12] Aris Gkoulalas-Divanis, Dinusha Vatsalan, Dimitrios Karapiperis, and Murat Kantarcioglu. 2021. Modern Privacy-Preserving Record Linkage Techniques: An Overview. 16 (2021), 4966–4987. https://doi.org/10.1109/TIFS.2021.3114026

[13] Martin Graf, Lukas Laskowski, Florian Papsdorf, Florian Sold, Roland Gremmelspacher, Felix Naumann, and Fabian Panse. 2022. Frost: a platform for benchmarking and exploring data matching results. *Proc. VLDB Endow.* 15, 12 (Aug 2022), 3292–3305. https://doi.org/10.14778/3554821.3554823

[14] 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. 2024. Simulated data for census-scale entity resolution research without privacy restrictions: a large-scale dataset generated by individual-based modeling. https://doi.org/10.12688/gatesopenres.15418.1

[15] Christopher Hampf, Martin Bialke, Hauke Hund, Christian Fegeler, Stefan Lang, Peter Penndorf, Nico Wöller, Frank-Michael Moser, Arne Blumentritt, Ronny Schuldt, et al. 2025. Privacy-preserving record linkage by a federated trusted third party (fTTP)–unlocking medical research potential in Germany. *GMS Med Inform Biom Epidemiol* 21, Doc05 (2025). https://doi.org/10.3205/mibe000277

[16] Kai Hildebrandt, Fabian Panse, Niklas Wilcke, and Norbert Ritter. 2020. Large-Scale Data Pollution with Apache Spark. *IEEE Transactions on Big Data* 6, 2 (06 2020), 396–411. https://doi.org/10.1109/TBDATA.2016.2637378

[17] Maximilian Jugl and Toralf Kirsten. 2024. Gecko: A Python library for the generation and mutation of realistic personal identification data at scale. *SoftwareX* 27 (2024), 101846. https://doi.org/10.1016/J.SOFTX.2024.101846

[18] Hanna Köpcke, Andreas Thor, and Erhard Rahm. 2010. Evaluation of entity resolution approaches on real-world match problems. *Proceedings of the VLDB Endowment* 3, 1-2 (2010), 484–493.

[19] Fabian Panse, André Düjon, Wolfram Wingerath, and Benjamin Wollmer. 2021. Generating Realistic Test Datasets for Duplicate Detection at Scale Using Historical Voter Data. In *Proceedings of the 24th International Conference on Extending Database Technology*. https://doi.org/10.5441/002/edbt.2021.67

[20] Anna Primpeli and Christian Bizer. 2020. Profiling entity matching benchmark tasks. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 3101–3108.

[21] Florens Rohde, Victor Christen, and Erhard Rahm. 2025. SecUREmatch: Integrating Clerical Review in Privacy-Preserving Record Linkage. In *SIGMOD-Companion '25*. Berlin. https://doi.org/10.1145/3722212.3725131

[22] Florens Rohde, Martin Franke, Victor Christen, and Erhard Rahm. 2023. Value-specific Weighting for Record-level Encodings in Privacy-Preserving Record Linkage. In *BTW 2023*. Gesellschaft für Informatik e.V., Dresden. https://doi.org/10.18420/BTW2023-21

[23] Florens Rohde, Martin Franke, Ziad Sehili, Martin Lablans, and Erhard Rahm. 2021. Optimization of the Mainzelliste software for fast privacy-preserving record linkage. *Journal of Translational Medicine* 19, 33 (2021). https://doi.org/10.1186/s12967-020-02678-1

[24] Rainer Schnell, T. Bachteler, and J. Reiher. 2009. Privacy-preserving record linkage using Bloom filters. *BMC Med. Inf. & Decision Making* 9 (2009), 41.

[25] Rainer Schnell, Tobias Bachteler, and Jörg Reiher. 2011. A Novel Error-Tolerant Anonymous Linking Code. *German RLC Working Paper* (2011).

[26] Rainer Schnell and Christian Borgs. 2016. Randomized response and balanced Bloom filters for privacy preserving record linkage. In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*. IEEE, 218–224.

[27] Rainer Schnell and Christian Borgs. 2016. XOR-Folding for Bloom Filter-based Encryptions for Privacy-preserving Record Linkage. *Working Paper WP-GRLC-2016-03, German Record Linkage Center, Nuremberg* (2016). https://doi.org/10.2139/ssrn.3527984

[28] Anushka Vidanage, Thilina Ranbaduge, Peter Christen, and Rainer Schnell. 2022. A Taxonomy of Attacks on Privacy-Preserving Record Linkage. *Journal of Privacy and Confidentiality* 12, 1 (2022). https://doi.org/10.29012/jpc.764