(Privately) Estimating Linkage Quality for Record Linkage

Martin Franke
franke@informatik.uni-leipzig.de
Leipzig University & ScaDS.AI
Leipzig, Germany

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

Peter Christen
peter.christen@anu.edu.au
The Australian National University
Canberra, Australia

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

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

ABSTRACT

Record linkage is the task of identifying records from different databases that refer to the same real-world entity. This task is an essential component of data integration to facilitate data analysis in a variety of domains, including healthcare, national security, and e-commerce. To evaluate the quality of record linkage approaches, the performance measures of precision, recall, and F-measure are commonly used. These measures require ground truth data that specifies known matches and non-matches. However, in practical linkage applications there typically is no such ground truth data available. Although linkage quality can be assessed manually by domain experts, such a clerical review process is time- and resource-consuming and generally not feasible when linking databases that are very large or that contain sensitive (personal) data. We review existing and propose improved unsupervised approaches for estimating the quality of linkage results. We evaluate our approaches on multiple datasets from three different domains. This evaluation shows that our approaches outperform existing methods and lead to estimates that are close to the actual linkage quality.

1 INTRODUCTION

Comprehensive data analysis in multi-site research projects requires careful preparation and integration of relevant data from various heterogeneous databases. For instance, in (bio-)medical research or clinical trials, data on patients treated at different health facilities must be consolidated. Since global identifiers are typically absent [7], it is necessary to identify records from the different databases referring to the same real-world entity, such as a patient. This problem is known as record linkage [7] and relies on comparing available quasi-identifiers such as the names, addresses, and dates of birth of patients.

Record linkage is a challenging task due to data quality, scalability, as well as privacy and confidentiality issues [9]. Most importantly, record linkage algorithms must achieve high linkage quality, as this is essential for their practical applicability and utility. Ideally, a record linkage approach should find all matches (pairs of records referring to the same entity), despite possible data quality problems, like erroneous, outdated, or incomplete data, in the source databases [10]. At the same time, false matches (pairs of records classified as matches, but referring to two different entities) should be avoided as much as possible, as otherwise conclusions based on incorrect assumptions may be drawn.

In many record linkage applications, however, there is no ground truth (gold standard) data available that specifies if two records refer to the same entity or not (true match status) [7]. One possibility to acquire ground truth data is to manually generate such data by (smartly) sampling record pairs and manually classifying them as a match or a non-match [7]. Similarly, domain experts can manually assess linkage results by (visually) inspecting classified record pairs in order to confirm or reject match decisions [24]. However, such a manual classification (also known as clerical review) is time- and resource-consuming as well as error-prone, especially for datasets that are large and/or difficult to classify. It can therefore lead to many potential matches, i.e., candidates for which it is unclear if they refer to the same entity or not.

Evaluating linkage quality becomes even more challenging when personal or sensitive data needs to be linked [9]. This problem is addressed by privacy-preserving record linkage (PPRL) techniques [17], where linkage is conducted on encoded data using secure protocols such that no sensitive information is revealed during the linkage process to protect the privacy of individuals [50].

In privacy-constrained scenarios, it is generally not possible to inspect actual (quasi-identifying) attribute values of classified record pairs as these can be sensitive and reveal the identity of an individual. Furthermore, the organizations conducting the linkage are generally not allowed or willing to share ground truth or training data. There is limited work [6, 24] that investigates approaches for manual clerical reviews working on partially (visually) masked quasi-identifying attribute values. Such approaches, however, will again be time- and resource-consuming while making the clerical review process likely to be less accurate than if complete attribute values were available for manual assessment. Unsupervised approaches for estimating linkage quality are therefore required to overcome this lack of ground truth data. So far, however, only a few such approaches have been proposed [21, 30]. As we show in our work, in many scenarios the estimated measures do not correlate well with the actual linkage quality. We therefore propose several extensions to existing approaches, as well as novel heuristics, to improve estimates for linkage quality. In particular, we make the following contributions:

- We adapt existing and propose novel unsupervised methods for estimating linkage quality based on a given similarity graph. Our methods address various data quality issues such as heterogeneity of records and duplicates in the same database. Our methods can be used in practice for both traditional and privacy-preserving record linkage

applications, in particular, to optimize linkage configurations, such as the classification threshold, which is often a challenging task.
- To estimate the overlap between datasets, our methods require a set of attributes where the values for true matching records are mostly the same, while the values for non-matches mostly differ. To achieve this aim, we develop an apriori-like strategy to automatically determine suitable attribute combinations, in particular for heterogeneous datasets where a manual selection of attributes is hard.
- We comprehensively evaluate our methods for estimating linkage quality against two baseline methods proposed in the literature [30, 31] using real-world datasets from three different domains (persons, music, and cameras).

The remainder of this paper is structured as follows. In Sect. 2, we outline the basic record linkage process and explain how linkage quality is assessed. Then, we define the problem of estimating linkage quality (Sect. 3) and discuss related work (sect. 4). In Sect. 5 we present our novel approaches for estimating the linkage quality for both clean and dirty databases. In Sect. 6, we discuss the privacy aspects of using similarity graphs and cryptosets. In Sect. 7, we evaluate our approaches on different datasets to validate their practical applicability. Finally, we conclude our work in Sect. 8.

2 BACKGROUND

The general linkage process consists of multiple steps as shown in Fig. 1 [7]. Without loss of generality, we assume the task of linking two databases, D_A and D_B. However, the process, as well as our methods, can easily be extended to the linkage of multiple (more than two) databases. In the following, we will describe each step in more detail. It is assumed that general information and linkage parameters, such as schema information or attributes used for linkage, are exchanged in advance between the database owners A and B that are providing the databases to be linked [47].

The linkage of databases can be conducted under different protocols depending on the specific use case and (eventual) privacy requirements [9, 17]. Depending on the protocol, the database owners may be involved in all linkage steps, or they only need to pre-process and possibly encode and block their databases before sending them to a linking unit. Such a linkage unit is a special party that participates in the linkage process by conducting the linkage of the databases sent to it. A detailed description of the different linkage protocols is provided in [9].

Pre-processing: This step focuses on resolving data quality issues (data cleaning) and ensuring that all records follow the same structure and formats (data standardization) [7].

Encoding: Linking sensitive databases requires that no private or confidential information is revealed during the linkage [9, 17, 50]. In such scenarios, the database owners have to perform an additional step where records are encoded or encrypted in a way that sensitive attribute values are secured from re-identification. To prevent the database owners from re-identifying each other's records, e.g., using dictionary attacks [51], the linkage process must follow specific protocols that define how data is exchanged between the linkage participants [9]. Most PPRL approaches consider an honest-but-curious security model, which assumes that all parties involved in a linkage project follow the linkage protocol but try to learn the other parties' sensitive data [51].

Blocking / Filtering: The trivial approach to link two databases is to compare every possible pair of records. To overcome this quadratic complexity, blocking or filtering techniques are commonly used to reduce the number of record comparisons [37]. Record pairs that do not meet predefined blocking or filtering criteria are not considered to be a match and hence are not compared in detail.

Comparison: Each candidate record pair is compared using similarity functions that are applied on the records' attributes. A similarity function generally calculates a value between 0 and 1 that quantifies how similar two attribute values are [7]. In general, several record attributes are compared using an appropriate similarity function. Each record comparison therefore results in a similarity vector, where each entry represents the result of a specific similarity function evaluated on a specific pair of attributes.

The output of this step is a set of candidate record pairs together with their similarity vector. This result can be considered as a similarity graph. A graph is a pair $G = (V, E)$ such that $E \subseteq [V]^2$, i.e., elements of E are 2-element subsets of V. More specifically, a similarity graph $SG = (V, E)$ is a graph in which vertices of V represent records and edges of E connect two compared records and hold their resulting similarity vector.

Classification: In this step a classification model is used to assign each candidate record pair (based on its similarity vector) to one of two (or three) classes: matches, non-matches, and (optionally) potential matches [7]. The class of potential matches contains those candidate record pairs where the model was not able to make a clear decision. While different classification techniques have been developed [3, 36], many approaches rely on threshold-based classification. At first, the similarity vector for a pair of records is aggregated into a single similarity score sim_Δ, by calculating, for example, a weighted sum over the vector elements.

Two threshold values $t_\uparrow$ and $t_\downarrow$ can then be defined such that all record pairs with $sim_\Delta \geq t_\uparrow$ are classified as a match, while pairs with $t_\uparrow > sim_\Delta \geq t_\downarrow$ as a potential match and pairs with $sim_\Delta < t_\downarrow$ as a non-match. If $t_\uparrow = t_\downarrow$, then records are classified into two classes only (matches and non-matches). We assume that each edge in the similarity graph has been aggregated into sim_Δ and labeled by the class the connected record pair belongs to (match, non-match, or potential match). Vertices without edges are implicitly considered as non-matches. For the remainder of the paper, we consider that both thresholds are the same ($t_\uparrow = t_\downarrow$).

Post-processing: The output of the classification step is generally not the final outcome of a record linkage process. Certain link constraints must be satisfied depending on the characteristics of the databases and the application that uses the linked dataset [9]. Generally, two cases are distinguished [36]: (1) both databases are *clean* in the sense that they are free of duplicates, i.e., in each database there are no two records that refer to the same entity. More formally, $\nexists a, b \in D : a \equiv b \wedge a \neq b$, where $\equiv$ denotes equivalence (reference to the same entity), while $=$ denotes equality (reference to the same values). For instance, let $a = [\text{Mary}, \text{Smith}, 20-05-1985]$ and $b = [\text{Marie}, \text{Smith}, 20-05-1985]$, then $a \neq b$, while we can assume $a \equiv b$. (2) at least one database is *dirty* in the sense that it contains duplicates, i.e., there are at least two records that refer to the same entity ($\exists a, b \in D : a \equiv b \wedge a \neq b$).

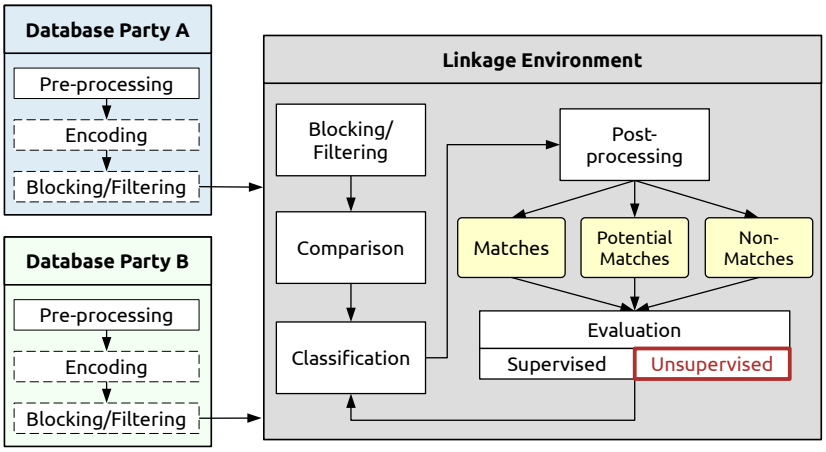

Figure 1: Outline of the general record linkage process. Steps in dashed boxes are optional. The highlighted box consists of our proposed methods.

After the classification step, the similarity graph contains edges (links) of different types that are important for post-processing, as we describe below. In the following, the degree $\deg(a)$ of a graph vertex a is defined as the number of edges that are incident to that vertex [11]. Similarly, we define the degree $\deg(e)$ of a graph edge $e = (a, b)$ as the maximum degree of its endpoints (vertices) a and b, i.e., $\deg(e) = \max(\deg(a), \deg(b))$, where $(a, b) \in E$.

- **One-to-one link**: An edge $e = (a, b)$ between two vertices $a, b \in V$ with $\deg(e) = 1$.
- **Multi-link**: An edge $e = (a, b)$ between two vertices $a, b \in V$ with $\deg(e) > 1$.
- **One-to-many link**: An edge $e = (a, b)$ between two vertices $a, b \in V$ where $\deg(a) = 1$ and $\deg(b) > 1$.
- **Many-to-one link** An edge $e = (a, b)$ between two vertices $a, b \in V$ where $\deg(a) > 1$ and $\deg(b) = 1$.
- **Many-to-many link**: An edge $e = (a, b)$ between two vertices $a, b \in V$ where both $\deg(a) > 1$ and $\deg(b) > 1$.

If the databases are duplicate-free, then records from the same database are usually not compared. As a consequence, the similarity graph forms a bipartite graph [11]. Thus, V allows a division into two partitions, namely V_A and V_B where $V = V_A \cup V_B$, such that every edge has its ends in different partitions, i.e., vertices in the same partition are not adjacent. The partition V_A only consists of records from database D_A and partition V_B only of records from database D_B, respectively. Since we assume duplicate-free databases, any record of D_A can match to at maximum one record of D_B and vice-versa. Thus, the similarity graph needs to be 1-regular and consequently must only contain edges with a degree of 1 (one-to-one links). In this case, post-processing applies a one-to-one link restriction to the match result by resolving all multi-links [14, 35]. This is equivalent to finding a matching over the similarity graph [11] (the term matching here refers to the graph-theoretic terminology). Given a graph $G = (V, E)$, a matching $M \subseteq E$ is a set of edges without common vertices, i.e., all edges are pairwise non-adjacent.

If one database is clean and the other is dirty, then one-to-one links and one-to-many links are permitted. If both databases are dirty, then all types of links listed above can potentially occur. In such a situation, each database owner could first individually run a (intra-source) deduplication process before performing the actual holistic (inter-source) linkage. While each database owner can optimize the deduplication configuration locally and potentially perform a manual assessment of the linking result, this approach has some drawbacks. At first, intra-source duplicates may be fused into a single record (cluster representative), e.g., by selecting attribute values that are more likely to be complete, accurate, and up-to-date. Using this approach, the amount of available information is reduced which potentially leads to more false negatives.

Furthermore, errors in the deduplication process of a source are possible, where two records are considered as match while they actually refer to different entities (intra-source false positive). As a consequence, entities are wrongly fused and this error is propagated through the whole process, which in turn can lead to inter-source false positives [32]. Therefore, it can be beneficial to retain intra-source duplicates. Consequently, records from the same database need to be compared leading to a similarity graph with intra-source links. By definition, this will make the similarity graph no longer bipartite. However, by removing all intra-source links a bipartite subgraph can be obtained.

Besides enforcing link cardinality constraints, post-processing ensures that the linkage result fulfills the transitive closure [7]. For records $a, b, c \in V$, this property guarantees that if both the pair (a, b) and (a, c) are classified as a match, then the pair (b, c) must also be a match. The transitive closure may be violated due to missed true matches, for example, due to blocking [7].

Evaluation: Given a ground truth (gold standard) dataset containing the true match status of a set of record pairs, four classification outcomes are possible for each pair of records:

- **True positive:** A true positive is a record pair that has been classified as a match and the pair is a true match. The two records refer to the same entity.
- **False positive:** A false positive is a record pair that has been classified as a match, but it is not a true match. The two records refer to different entities.
- **True negative:** A true negative is a record pair that has been classified as a non-match and it is a true non-match. The two records refer to different entities.
- **False negative:** A false negative is a record pair that has been classified as a non-match, but it is a true match. The two records refer to the same entity.

For a specific classification configuration, e.g., certain classification threshold, this results in a confusion matrix [18] reporting the total number of true positives (*tp*), true negatives (*tn*), false positives (*fp*) and false negatives (*fn*). Based on the confusion matrix different quality measures can be calculated [19]. The most common measures are recall (R), precision (P), and F-measure (F) which are defined as follows:

$$R = \frac{tp}{tp + fn} \qquad P = \frac{tp}{tp + fp} \qquad F = 2 \times \frac{P \times R}{P + R} \quad (1, 2, 3)$$

The F-measure (F) combines recall and precision into a single number and can be defined as the harmonic mean of both measures [8].

3 PROBLEM DEFINITION

Let D_A and D_B be two databases from database owners A and B, respectively. Let $SG = (V_A, V_B, E)$ be the similarity graph resulting from a record linkage process using a certain linkage configuration. Based on an analysis of the two databases and the given similarity graph SG, we aim to estimate the linkage quality in terms of *tp*, *fp* and *fn*. From these estimates precision and recall, as well as aggregated measures such as the F-measure, can be calculated. We assume that no ground truth data is available that can be used, for example, due to privacy or data protection concerns.

4 RELATED WORK

Existing methods to estimate linkage quality in the context of record linkage can be roughly divided into the following categories.

Manual assessment: The result of a linkage is manually inspected by domain experts in order to assess the linkage quality outcome [7]. The disadvantage of such approaches is that they can be very time- and resource-consuming. To limit this effort, often only a small sample of record pairs is revised, in particular edge cases. These are pairs that are hard to classify and thus have high uncertainty [7]. A simple sampling method is proposed by Boyd et al. [5] where record pairs at different threshold values are sampled and clerically reviewed. The obtained results are then applied to the entire dataset providing estimates for the number of false positives and false negatives. Marchant and Rubinstein proposed OASIS [28], a tool that takes an unlabeled dataset as input and intelligently selects items to be (manually) labeled to provide an estimate of the linkage quality. To minimize the amount of labeling required, OASIS uses an adaptive importance sampling method.

In privacy-preserving settings, however, such manual inspection is even harder to employ. Initial work [24] addresses this problem by visual masking and partly hiding actual attribute values, in order to allow manual link decisions without compromising the privacy of individuals. However, manual decisions based on masked attribute values might also be less accurate compared to reviews based on fully visible attribute values.

Supervised approaches: Linkage quality can be estimated based on ground truth (training) data where the match status of a set of record pairs is known. Such training data need to be of high quality and contain a large diversity of example pairs, especially those that are difficult to classify. Heise et al. [21] proposed a sampling-based approach for duplicity assessment which estimates the number and sizes of duplicate record clusters in a dataset. The main benefit of their approach is that it can efficiently approximate the number of duplicates while only performing a fraction of the candidate comparisons compared to what an actual record linkage process would take. Binette et al. [4] estimate linkage quality from samples by using (partial) ground truth data. Similarly, in [12, 20] partial ground truth data is submitted to the linkage process in the form of positive/negative controls. Positive controls are records that are known to be a match. In contrast, negative controls are records that should definitely not match any other record. In [29], such controls are used for the linkage of prisoner records and a register of deaths. In that specific scenario, for a subset of prisoners, it is known that they died in prison (positive controls) while for another subset of prisoners, it is known that they were alive at the time of the linkage (negative controls). By counting the number of correctly classified control records, the linkage quality can be calculated. In general, the control records can also be artificially created jointly by the database owners and then employed in the linkage process.

Again, in privacy-preserving record linkage scenarios, database owners might not be able or willing to prepare and exchange training data due to privacy constraints [9].

Unsupervised approaches: These approaches do not have access to the characteristics of true matching and non-matching record pairs. Lamiroy and Sun [26] propose an approach to measure recall and precision in the absence of ground truth data. Their method requires access to different competing approaches, such as different classifiers, in order to establish a ranking and find an overall consensus between these approaches. The drawback of this approach is that it is sensitive to collective bias, namely if the competing approaches are consistent in their errors. Similarly, Platanios et al. [39] propose methods for estimating the accuracy of different competing classifiers based on their agreement rates over unlabeled data. The authors show that their approach is able to estimate accuracy if the competing classifiers do not make independent errors.

Other unsupervised approaches rely on dataset characteristics and the similarities between pairs or groups of records. Such approaches are closely related to clustering approaches that can be used for classification, as well as for post-processing [7]. Clustering is the process of partitioning data objects into subsets (called clusters), such that intra-cluster similarity is maximized while inter-cluster similarity is minimized. This means that objects in the same cluster have a high similarity while objects in different clusters have a low similarity to each other [18]. Clustering techniques utilize different heuristics but are generally executed in an unsupervised fashion. In [31], Nikolov et al. propose an unsupervised approach that aims to estimate linkage quality in the absence of labeled data. Therefore, pseudo-precision (*PP*) and pseudo-recall (*PR*) measures are used which are defined as follows:

$$PP = \frac{|\{a \in V_A | \exists b \in V_B : (a, b) \in E\}|}{\sum_{a \in V_A} |\{b \in V_B | (a, b) \in E\}|} \quad (4)$$

$$PR = \frac{|E|}{\min(|V_A|, |V_B|)} \quad (5)$$

Assuming that the databases to be linked are clean, the pseudo-precision measure is based on the following fact: If there are multiple links originating from the same record, at most one can be correct. The other links are necessarily errors. The pseudo-recall measure considers the number of records in the smaller partition (database) as the maximum number of possible matches. However, this is only the case if one database is a subset of the other database. This, in turn, will be rarely the case in most record linkage scenarios [7]. As a consequence, pseudo-recall tends to

(strongly) underestimate the actual recall, if the overlap between the two databases is low. Besides, pseudo-recall can result in values greater than 1, namely if $|E| > \min(|V_A|, |V_B|)$. To overcome this issue, Ngomo and Lyko [30] refined the approach by specifying an alternative pseudo-recall variant which is defined as:

$$PR_{Alt} = \frac{|\{a \in V_A | \exists b \in V_B : (a,b) \in E\}| + |\{b \in V_B | \exists a \in V_A : (a,b) \in E\}|}{|V_A| + |V_B|} \tag{6}$$

This pseudo-recall measure indicates how well the records in both databases are covered by the linkage result. A pseudo-recall value of 1 means that every record of database D_A is linked to at least one record of database D_B and vice versa. While the results in [31] are promising, the authors in [30] achieved varying results with both positive and negative correlations between the estimated and the actual linkage quality. However, the results are hard to interpret as only F-measure values were reported and compared (a weakness of the F-measure reported by others [8]). It is thus difficult to assess if the ambiguous correlations reported are due to the recall or precision estimates.

5 ESTIMATING LINKAGE QUALITY USING SIMILARITY GRAPHS

The key idea of our methods for estimating linkage quality in the absence of ground truth data is to analyze both the input data and the similarity graph generated by a record linkage algorithm. For assessing the quality in terms of recall and precision, the number of true positives (*tp*), false positives (*fp*), and false negatives (*fn*) need to be approximately determined. In the following, we discuss different strategies considering the degree of vertices, similarity of edges, and cryptosets, to determine the relevant counts required for calculating precision and recall.

Because of possibly different data quality levels regarding duplicates in a database, we distinguish our methods as being suitable for deduplicated databases (clean) and databases containing (intra-source) duplicates (dirty). Assuming that the databases to be linked are duplicate-free, the number of possible matches for each record is limited to one (see Sect. 2), and therefore our heuristics need to be more strict. In general, we assume that the similarity graph was generated without applying a one-to-one cardinality restriction as part of a post-processing step. By applying a one-to-one cardinality restriction, the most likely matching record out of a set of candidates would be selected [15]. This would lead to a loss of information about the ambiguity of possible match candidates.

5.1 Deduplicated Databases

The main difference between clean and dirty data sources is that with the former a record $a \in D_A$ can correspond to at maximum one record $b \in D_B$. Otherwise, the database D_B is not duplicate-free if a record $b' \in D_B$ exists where $b' = a$. Due to the transitive closure of a regarding equality [7], record b would be equal to b', which contradicts the assumption of duplicate-free data sources.

We utilize this constraint and the degree of nodes in the similarity graph as indicators for true positives. A one-to-one link implies that there is exactly one match candidate for a record. In contrast, a multi-link implies that there are several match candidates for a record leading to uncertainty regarding the decision of which records to match. While for clean databases each additional match candidate will be a false positive (without a post-processing step), for dirty databases multiple match candidates may form an intra-source duplicate (as we will discuss in Sect. 5.2). In addition to the edge degree, the edge weight (the aggregated similarity sim_Δ or a confidence value) is also an important criterion. The higher the edge weight and the greater the difference to the similarity threshold value is, the more certain a match decision will be.

In the following, we describe the different strategies using the vertex degree and the edge similarity to estimate the number of true positives and false positives, as well as cryptosets to determine the number of false negatives.

5.1.1 Vertex Degree. Due to the constraint for duplicate-free databases, we can approximate the set of true positives by the records of a database D_A that have been linked to at most one record from the other database D_B. We can formalize the set of estimated true positives TP_A regarding a data source A as follows:

$$TP_A = \{a \in V_A \mid \exists b \in V_B : (a,b) \in E\} \tag{7}$$

Using the estimation of the set of true positives, we can approximately determine precision with Eq. (8) where $|E|$ represents the number of edges. In this approximation, the number of true positives is limited by the minimum number of expected true positives regarding the set of records from TP_A and TP_B being linked. To relax the assumption of one-to-one links, Eqs. (9) and (10) considers the average of the number of records from TP_A and TP_B. Eq. (10) limits the number of links by the minimum of $|V_A|$ and $|V_B|$ motivated by the duplicate-free assumption.

$$PP_{1:1} = \frac{\min(|TP_A|, |TP_B|)}{|E|} \qquad PP_{1:n} = \frac{|TP_A| + |TP_B|}{2 \cdot |E|}$$
$$PR_{AltMin} = \frac{|TP_A| + |TP_B|}{2 \cdot \min(|V_A|, |V_B|)} \tag{8, 9, 10}$$

5.1.2 Similarity Scores. In addition to the graph structure, similarity graphs provide information for each edge representing how likely a match between the linked records is. Therefore, we utilize the similarities to calculate for each edge a probability to be a true positive depending on its adjacent edges. As we discuss below, the intuition is that we can select for each record only one edge, and therefore we utilize the similarities of adjacent edges as a probability to select one edge per record. The calculated probability for each edge and the restriction of edges based on the duplicate-free assumption can then be used to calculate the expectation of the number of true positives for the given similarity graph.

For calculating the probability of a true positive given an edge $e = (a,b) \in E$, we determine two probabilities, $p^A_{tp}(e)$ and $p^B_{tp}(e)$ representing how likely e is a true positive considering records $a \in V_A$ and $b \in V_B$. The probability $p^A_{tp}(e)$ defined in Eq. (11) (with $p^B_{tp}(e)$ calculated in a similar way) is based on the similarity of edge e and normalized by the sum of the similarities of its adjacent edges associated with a record a. Here, $N(v)$ denotes the neighborhood of a vertex v which is the set of vertices adjacent to v [11].

$$p^A_{tp}(e) = \mathbb{P}[e = (a,b) \in TP \mid a \in V_A] = \frac{sim_\Delta(e)}{\sum_{b' \in N(a)} sim_\Delta(a,b')} \tag{11}$$

The probabilities $p^A_{tp}(e)$ and $p^B_{tp}(e)$ are used to determine a joint probability indicating how likely it is that e is a true positive. To estimate the number of true positives, we then calculate the expected value of true positives based on the joint probability

of e where $e \in E_{Sel}$. The set E_{Sel} defined in Eq. 12 consists of edges maximizing the similarity for at least one incident vertex a or b regarding the edges being adjacent with the vertices of the neighborhood of a respectively of b. Due to our assumption that the data sources are deduplicated, we assume that for each record the edge with the largest similarity is most likely a true link.

$$E_{sel} = \{(a,b) \in E \mid \max_{b' \in N(A)} \left(sim_\Delta(a,b')\right) = sim_\Delta(a,b) \vee \max_{a' \in N(B)} \left(sim_\Delta(a',b)\right) = sim_\Delta(a,b)\} \quad (12)$$

To calculate the expected value of true positives considering the edges of E_{sel}, we determine the sum of the joint probability over all edges $e \in E_{sel}$ that is formally defined as follows:

$$\mathbb{E}(TP) = \sum_{e=(a,b) \in E_{sel}} p^A_{tp}(e) \cdot p^B_{tp}(e) \quad (13)$$

We can then define our new precision estimate PP_{prob} as:

$$PP_{prob} = \frac{\mathbb{E}(TP)}{|E|} \quad (14)$$

Example: Using the various methods, we can estimate the number of true positives for our example shown in Fig. 2. In this example, $|E| = 10, |V_A| = |V_B| = 6$ as well as $|TP_A| = 6$ and $|TP_B| = 5$. The determined sets are used to calculate $PP_{1:1} = \min(5,6)/10 = 0.5$, $PP_{1:n} = (5+6)/2 \cdot 10 = 0.55$ and $PP = 6/10 = 0.6$. For calculating PP_{prob}, we need to calculate the probabilities $p^A_{tp}(e)$ and $p^B_{tp}(e)$ for each edge $e \in E_{sel} := \{(0,8), (1,6), (2,10), (3,9), (4,7), (5,9)\}$ being aggregated by $\mathbb{E}(TP)$. For instance, the probabilities $p^A_{tp}((3,9))$ and $p^B_{tp}((3,9))$ for edge $(3,9)$ are $p^A_{tp}((3,9)) = 0.68/(0.68+0.41+0.43) \approx 0.45$ and $p^B_{tp}((3,9)) = 0.68/(0.68+0.73) \approx 0.48$, respectively. Overall, the expected number of true positives is $\mathbb{E}(TP) = 2.662$ resulting in $PP_{prob} = 0.266$.

5.1.3 Cryptosets. To approximate recall, we need to determine the number of overlapping records. In order to guarantee the privacy of sensitive personal information we utilize cryptosets [46]. The main idea of using cryptosets is the analysis of histograms consisting of record-depending information from both databases. An example of how cryptosets are generated is illustrated in Fig. 3.

For each record, a private identifier is constructed by applying specific functions on a set of attributes. These private identifiers do not need to be unique, but the number of records with the same private identifier should be kept small. On the one hand, the more unique the private identifiers are, the more accurate the cryptoset estimate of the overlap between the two databases will be. On the other hand, the construction of the private identifiers should be error-tolerant. Records that refer to the same entity but contain errors or inconsistencies, such as typos or missing values, should ideally produce the same private identifier otherwise the overlap will be underestimated. In our example shown in Fig. 3, the private identifiers are generated by concatenating the first three characters of the first name and last name and the last two digits of the year of birth.

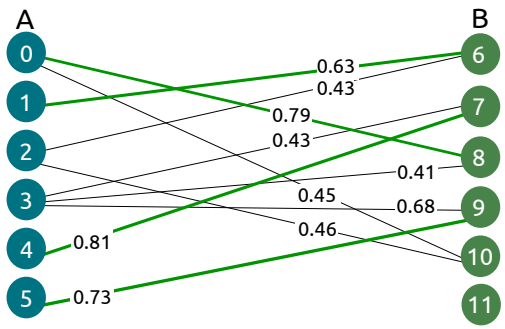

Figure 2: Example similarity graph of records from databases A and B. True positive links are shown with thick green lines.

The resulting private identifier id_{priv} is transformed to a public identifier id_{pub} in the range $[0, L-1]$ using a one-way cryptographic hash function h, i.e., $id_{pub} = h(id_{priv})$ [9]. Then, each database owner initializes a histogram of length L and increments for each record the count at the position id_{pub} mod L corresponding to the public ID of the record (bottom of Fig. 3).

Cryptosets have a trade-off between estimation error and security risk [46]. This trade-off is controlled by the cryptoset length L. Longer cryptosets result in fewer collisions as fewer public identifiers (records) are mapped to the same position. While this makes the estimates more accurate, the cryptosets become less secure.

Overlap Estimation. Assuming two (sensitive) databases D_A and D_B for which cryptosets C_A and C_B have been constructed using the same protocol, then the overlap of records $CE(C_A, C_B)$ (**c**rypto**s**et **e**stimation) in these two databases, $|D_A \cap D_B|$, can be estimated as follows:

$$CE(C_A, C_B) = pc(C_A, C_B) \cdot \sqrt{\frac{\max(|D_A|, |D_B|)}{\min(|D_A|, |D_B|)}} \quad (15)$$

where $pc(\cdot,\cdot)$ is the Pearson correlation coefficient. Note that $|D_A| = \sum_{i=0}^{L-1} C^A_i$ and $|D_B| = \sum_{i=0}^{L-1} C^B_i$. The Pearson correlation coefficient is defined in Eq. 16 based on the covariance between the cryptosets of C_A and C_B normalized by the product of the standard deviations of C_A and C_B, where $\overline{C_A}$ and $\overline{C_B}$ are the means of frequencies of the id_{pub} distribution of C_A and C_B, respectively.

$$pc(C_A, C_B) = \frac{\sum_{i=0}^{L-1}(C_A[i] - \overline{C_A})(C_B[i] - \overline{C_B})}{\sqrt{\sum_{i=0}^{L-1}(C_A[i] - \overline{C_A})^2} \cdot \sqrt{\sum_{i=0}^{L-1}(C_B[i] - \overline{C_B})^2}} \quad (16)$$

We can now determine recall by utilizing the cryptoset-based approximation of the overlap from Eq. (15) and the approximation of the number of true positives based on $|TP_A|$ and $|TP_B|$, or $\mathbb{E}(TP)$ as calculated in Eqs. 7 and 13 for the databases D_A and D_B.

$$PR_{CE1:1} = \frac{\min(|TP_A|, |TP_B|)}{CE} \qquad PR_{CE1:n} = \frac{|TP_A| + |TP_B|}{2 \cdot CE}$$
$$PR_{CEprob} = \frac{\mathbb{E}(TP)}{CE} \quad (17, 18, 19)$$

Generation of Private Identifiers. Due to the importance of private identifiers for estimating the overlap, automatic approaches are required if the databases consist of heterogeneous or sensitive data which makes manual selection infeasible. We propose a method, as outlined in Algorithm 1, that automatically selects a subset of attributes to generate the identifiers based on the attribute characteristics such as uniqueness as well as value distribution. The selected attributes representing the private identifier influence the estimated overlap. A high number of records with

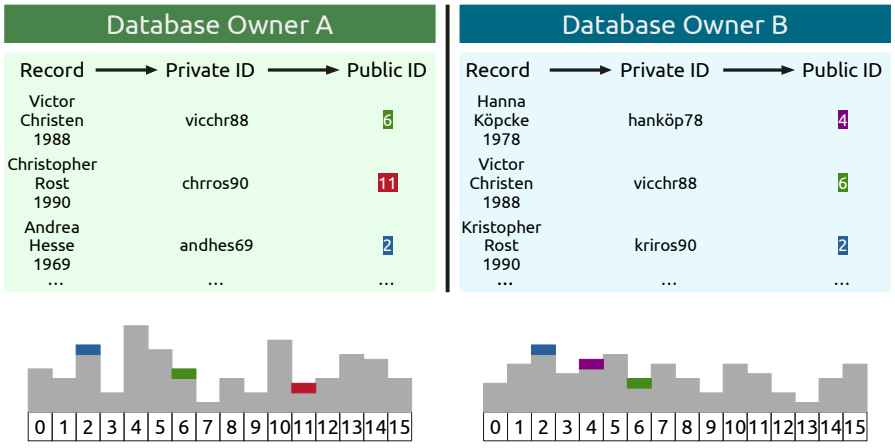

Figure 3: Illustration of the cryptoset approach to estimate the overlap of two (private) datasets (adapted from [46]).

ALGORITHM 1: Apriori-like approach to determine attributes for generating meaningful private identifiers

Input: **D**: dataset from a certain party, **A**: set of attributes
mr: threshold for the ratio of missing attribute values
t_{info}: threshold to filter uninformative attribute combinations
Output: **AC** set of attribute combinations to generate private ids

1 $\mathbf{AC} \leftarrow \emptyset$
2 $\mathbf{A}_{valid} \leftarrow filterAttributes(D, A, mr)$
3 $tempAttCombs \leftarrow apriori(A_{valid})$
4 **do**
5 $\quad filteredCombs \leftarrow \emptyset$
6 $\quad$ **for** $ac \in tempAttCombs$ **do**
7 $\quad\quad u \leftarrow computeUniqueness(\mathbf{D}, ac)$
8 $\quad\quad s \leftarrow computeWeightedUniformitySim(\mathbf{D}, ac, u)$
9 $\quad\quad info \leftarrow 2 \cdot {}^{(s \cdot u)}/_{(s+u)}$
10 $\quad\quad$ **if** $info \geq t_{info}$ **then**
11 $\quad\quad\quad filteredCombs \leftarrow filteredCombs \cup \{ac\}$
12 $\quad\quad\quad \mathbf{AC} \leftarrow \mathbf{AC} \cup \{ac\}$
13 $\quad tempAttCombs \leftarrow apriori(filteredCombs)$
14 **while** $tempAttCombs \neq \emptyset$
15 **return AC**

the same identifier results in an overestimated overlap whereas a small number leads probably to underestimation since the private identifiers are too unique so the intersection of the resulting histograms is small.

Therefore, we propose an automatic approach for selecting a subset of attributes satisfying different criteria so that the resulting identifiers enable an effective estimation [40]. The approach follows an apriori-like strategy [2], where we start with attribute sets of size one and combine them. An attribute combination is added to the final result set **AC** if the harmonic mean based on the uniqueness (u) and the weighted similarity (s) regarding a uniform distribution is above a threshold t_{info} (Algorithm 1 line 5-13). The attribute combination is also added to the candidate set *filteredCombs* to generate larger attribute combinations *tempAttCombs* being validated in the next iteration. The generation process stops if we cannot derive larger attribute sets satisfying the defined criteria in terms of uniqueness and similarity to a uniform distribution.

We define uniqueness (u) as the ratio of distinct values regarding an attribute combination and the number of records. Moreover, the similarity s is determined by computing the histogram intersection between the value distribution regarding a certain attribute combination and a uniform distribution. To avoid a high impact of combinations leading to a high uniqueness, we weigh the similarity by the uniqueness of an attribute combination with ${}^{u \cdot (1-u)}/_{0.25}$ mitigating the impact of combinations with a high ($u \approx 1$) or small uniqueness ($u \approx 0$). To reduce the number of attribute combinations, we filter the possible attributes based on the number of existing values at first (line 2). Our assumption here is that attributes or combinations with a high number of missing values result in ineffective identifiers for representing the underlying records.

5.2 Dirty Databases

The proposed methods in the previous section assume one-to-one links between the two databases. Consequently, if we use these methods for databases with duplicates, we would underestimate the number of true positives since multi-links are possible.

For estimating the number of true positives, we rely on the assumption that records being the same entity are similar to each other, which is also reflected in the similarity graph. As a result of the linkage process, records representing the same entity are elements of one connected component. Moreover, the records of a component should be similar to each other which is explicitly represented by the computed similarities. Nevertheless, the similarities can be different due to quality issues or edges missing due to the specified threshold. In this case, we cannot utilize the similarities directly to quantify the number of true positives. We reformulate the assumption that each record is similar to the other records using the personalized PageRank [33]. In the context of the personalized PageRank considering a certain record, each record of a connected component should be reachable with roughly the same probability. Otherwise, a record is more (dis)similar to a subset of records indicating that not all records refer to the same entity.

To quantify the number of true positives, we introduce a true positive score $tp_{score}(a, b)$ for each edge $e = (a, b)$ based on the personalized PageRank $pp(a, b)$ of the adjacent records a and b as well as the similarity sim_{Δ}. Ideally, the probability of reaching

a record b starting from a is equal to the probability by randomly selecting a record b' from the connected component CC of a since each record should be similar to the other records. The probability of randomly selecting a record of a connected component is $p_{uni} = \frac{1}{|CC|}$. Using the probabilities, we define the true positive score of an edge as:

$$tp_{score}\,(e = (a,b)) = (1 - |pp(a,b) - p_{uni}|) \cdot (1 - |pp(b,a) - p_{uni}|) \cdot sim_{norm}(a,b)$$

The first factor and the second factor represent the probability difference reaching node b starting from a and reaching node a starting from b, respectively. The third factor weighs the two differences using the min-max normalized similarity $sim_{norm}(a,b)$ between a and b. The smaller the differences and higher the similarity, the higher the tp_{score} for the edge $e = (a,b)$. The total number of true positives TP_{score} is estimated by the sum of $tp_{score}(e)$ overall identified matches $e \in E$. The resulting estimation is used to compute the precision PP_{dup} as follows:

$$PP_{dup} = \frac{TP_{score}}{|E|} \tag{20}$$

To measure recall, we use the estimated number of true positives TP_{score} compared to the estimated overlap CE by using cryptosets.

6 DISCUSSION OF PRIVACY ASPECTS

Our methods for estimating linkage quality rely on analyzing similarity graphs as well as cryptosets of the databases to be linked.

In the context of PPRL, there are only a few works that propose attacks on similarity graphs [49]. Such attacks aim to determine a mapping between the encoded data and publicly available plaintext data by using graph features (such as weighted node degrees). Our estimation methods, however, utilize existing similarity graphs which are typically generated by the linkage unit in PPRL scenarios. Therefore, our approaches do not add any privacy flaws but rather rely on the security of the method that was used to generate the similarity graph.

Cryptosets can be seen as a summary of the databases' contents that can be shared in public, untrustworthy environments to measure the overlap between private databases. In the literature, cryptosets are considered as information-theoretic secure [46] as it is not possible to determine which records are in a private database based on its cryptoset. For the overlap estimation, the cryptosets of the databases to be linked need to be shared with the linkage unit or between the database owners. Each cryptoset is a vector of length L containing the counts of public identifiers. Those public identifiers are determined by mapping the private identifiers of records representing (parts of) attribute values to an integer value in the range $[0, L-1]$ using a cryptographic one-way hash function. Setting $L \ll \min(|D_A|, |D_B|)$ results in a many-to-one relationship between private and public identifiers where the number of records being mapped to the same public identifier is typically large and thus impeding the alignment of specific records [46]. Consequently, even if an adversary knows the encoding function, dictionary-based attacks are not feasible.

In addition to this theoretical argument, information gain [9] can be used as a measure to quantify how much information is exposed by a cryptoset C_A compared to a theoretically optimal cryptoset C_U consisting of all possible values in the domain of private identifiers. Due to the large number of possible values in a domain, the public identifiers in C_U are approximately uniformly distributed so that each position in C_U is set with a probability of $1/L$. We can calculate the information gain $I\left(C_A \,\|\, C_U\right)$ using both entropies of C_A and C_U as shown in Eq. (21), where smaller information gain values represent higher privacy. If information gain is high, the frequency distribution of a cryptoset can potentially be used in a cryptanalysis attack to align it to a public value distribution such as telephone books or census data for names. However, no such attack has so far been developed.

$$I\left(C_A \,\|\, C_U\right) = \left(-\sum_{i=0}^{L-1} \frac{1}{L} \cdot log_2 \frac{1}{L}\right) - \left(-\sum_{i=0}^{L-1} \frac{C_A[i]}{|D_A|} \cdot log_2 \frac{C_A[i]}{|D_A|}\right) \tag{21}$$

Securely computing the intersection of private databases is an intensively studied problem with various approaches showing different security and complexity properties [25]. In general, two parties want to compute the intersection of their private sets without revealing anything to the other party other than the (number of) elements in the intersection. For a detailed discussion of different private set intersection protocols, we refer to [38]. Many approaches focus only on two parties, compute only the exact overlap (not considering errors or inconsistencies between matching records), or do not account for duplicate elements (multiset intersection) [1, 13, 16]. Therefore, we employ cryptosets as a specific solution to the private set intersection cardinality problem that meets our requirements. However, our approaches for estimating linkage quality are not strictly limited to cryptosets.

7 EXPERIMENTAL EVALUATION

In this section, we evaluate the proposed approaches for linkage quality assessment using datasets from three distinct domains with different characteristics. In the following, we describe the datasets we used as well as the methods and parameter settings for generating the similarity graphs.

7.1 Datasets

We use datasets from three different domains: voter records (personal information), records about music albums, and records about consumer products (cameras). In contrast to the voter datasets, the music and camera datasets are more heterogeneous (have different attribute structures) and show diverse types of errors. The voter and music datasets are clean (duplicate-free), while the product dataset is dirty and contains intra-source duplicates. We use the voter dataset to estimate the linkage quality in a PPRL scenario. The music and product datasets, in contrast, are used for estimating the quality in a non-privacy-oriented linkage context.

7.1.1 NCVR. We first consider a dataset provided by Panse et al. [34] that is based on the North Carolina Voter Registration (NCVR) database (https://www.ncsbe.gov/). This dataset contains over 120 million historic voter records with person-related attributes such as first name (FN), middle name (MN), last name (LN), year of birth (YOB), place of birth (POB), city, ZIP code, and sex. Compared to the other two datasets, MusicBrainz and Dexter as described next, it represents a homogeneous dataset in terms of the number of attributes and the characteristics of attribute values such as length distribution and amount of missing values. From this dataset, we extracted subsets A and B with $|A| = |B| =$ 200,000 and varying degrees of overlap (number of matches):

Table 1: Dataset overview and linking configuration of MusicBrainz and Dexter.

Dataset	Attributes	#Records	#Matches	Blocking Key	Similarity Function
Music Brainz	Artist, title, album, year, length, language, number	20,000	16,250	preLen1(album)	Trigram(title)
Dexter	Heterog. key-value pairs	21,023	185,839	mfr. name, model number	Trigram(model names, product code, sensor type), Euclid(opt./digital zoom, camera dim., price, weight, resolution)

- $NCVR_H$ (high overlap) where $|A \cap B| = 160{,}000$.
- $NCVR_{MH}$ (medium-high overlap) where $|A \cap B| = 120{,}000$.
- $NCVR_M$ (medium overlap) where $|A \cap B| = 100{,}000$.
- $NCVR_{LM}$ (low-medium overlap) where $|A \cap B| = 80{,}000$.
- $NCVR_L$ (low overlap) where $|A \cap B| = 40{,}000$.

Each singleton record is drawn from the NCVR snapshot of '2021-01-01'. Each duplicate pair (a, b) consists of records $a \in A$ and $b \in B$ where record a is drawn from a snapshot between '2008-01-01'(inclusive) and '2021-01-01'(exclusive), while record b is from snapshot '2021-01-01'. Moreover, there is a difference or error in at least one attribute that is not the year of birth: $\forall a, b : (YOB(a) = YOB(b)) \wedge \exists attr \in \{FN, MN, LN, POB, SEX\} : attr(a) \neq attr(b)$.

As we use this dataset to estimate the linkage quality in a PPRL scenario, we utilize Bloom filters as proposed by Schnell et al. [45] as an encoding technique. Bloom-filter-based encodings have become the quasi-standard for recent PPRL approaches in both research and real applications [9, 48]. We use record-level Bloom filters with a length of $m = 1024$, trigrams, and attribute weighting. To overcome the quadratic complexity of linkage, we use LSH-based blocking based on the Hamming distance as in previous work [14, 15]. To determine the similarity of candidate record pairs we use the Jaccard coefficient [9].

7.1.2 MusicBrainz. The MusicBrainz dataset is a synthetically generated dataset from the MusicBrainz (https://musicbrainz.org/) database. The dataset is corrupted by [22] consisting of five sources with duplicates for 50% of the original records. Each data source is duplicate-free but the records are heterogeneous regarding the characteristics of attribute values such as the number of missing values, length of values, and ratio of errors. The similarity graphs we used in our evaluation have been utilized in several previous studies [27, 41–43]. The linkage configuration is shown in Tab. 1.

7.1.3 Dexter. This dataset is derived from the camera dataset of the ACM SIGMOD 2020 Programming Contest (http://www.inf.uniroma3.it/db/sigmod2020contest/index.html). The dataset consists of 23 sources with ≈21,000 records and intra-source duplicates. Each data source consists of source-specific attributes. We used the same linkage configuration as in previous studies [41, 43] (see Tab. 1).

Table 2: Averaged information gain $I(C_A||C_U)$ and $I(C_B||C_U)$.

Config. private ID	$NCVR_L$	$NCVR_M$	$NCVR_H$
`[2FN, 2LN, YOB]`	0.0497	0.0506	0.0522
`[3FN, 3LN, YOB]`	0.0317	0.0320	0.0316
`[SD_FN, SD_LN, YOB]`	0.0302	0.0314	0.0317

7.2 Results

In the following, we evaluate our proposed methods for estimating the linkage quality on both clean (deduplicated) and heterogeneous/dirty databases. For each dataset, we analyze the recall, precision, and the resulting F-measure estimates and compare them with the actual results as calculated from ground truth data.

7.2.1 Clean Databases. An essential part of estimating the linkage quality is the estimation of the overlap of two databases utilizing the cryptoset method we described in Sect. 5.1.3. Due to independence regarding various similarity graphs, we evaluate the approach considering different manual-defined private identifiers on the NCVR datasets representing a homogeneous dataset. We consider three different private identifier configurations: `[2FN,2LN,YOB]`, `[3FN,3LN,YOB]`, and `[SD_FN,SD_LN,YOB]`, where $2A/3A$ extract the first 2/3 letters from the value of attribute A. Similarly, `SD_`A computes the Soundex [23] from the value of attribute A. We empirically set the cryptoset length to $L = 8192$ as this results in more accurate estimates with a lower standard deviation.

The results, as depicted in Fig. 4 (green bars), show that the cryptoset approach is able to estimate the overlap for different private identifier generation configurations. The estimates

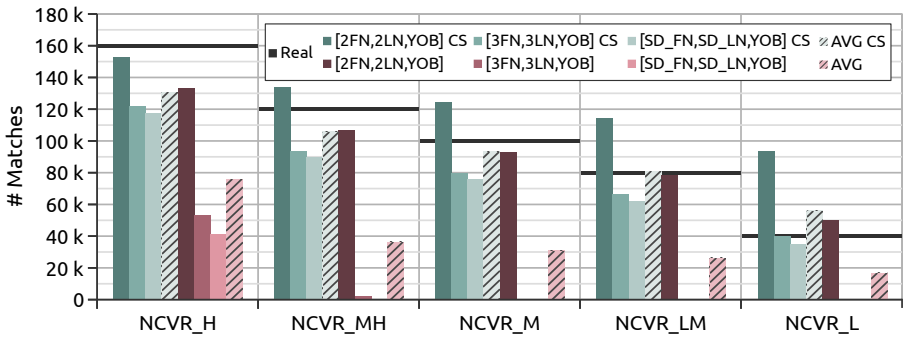

Figure 4: Evaluation of cryptoset approach to measure recall. Results based on ground truth are shown as horizontal lines.

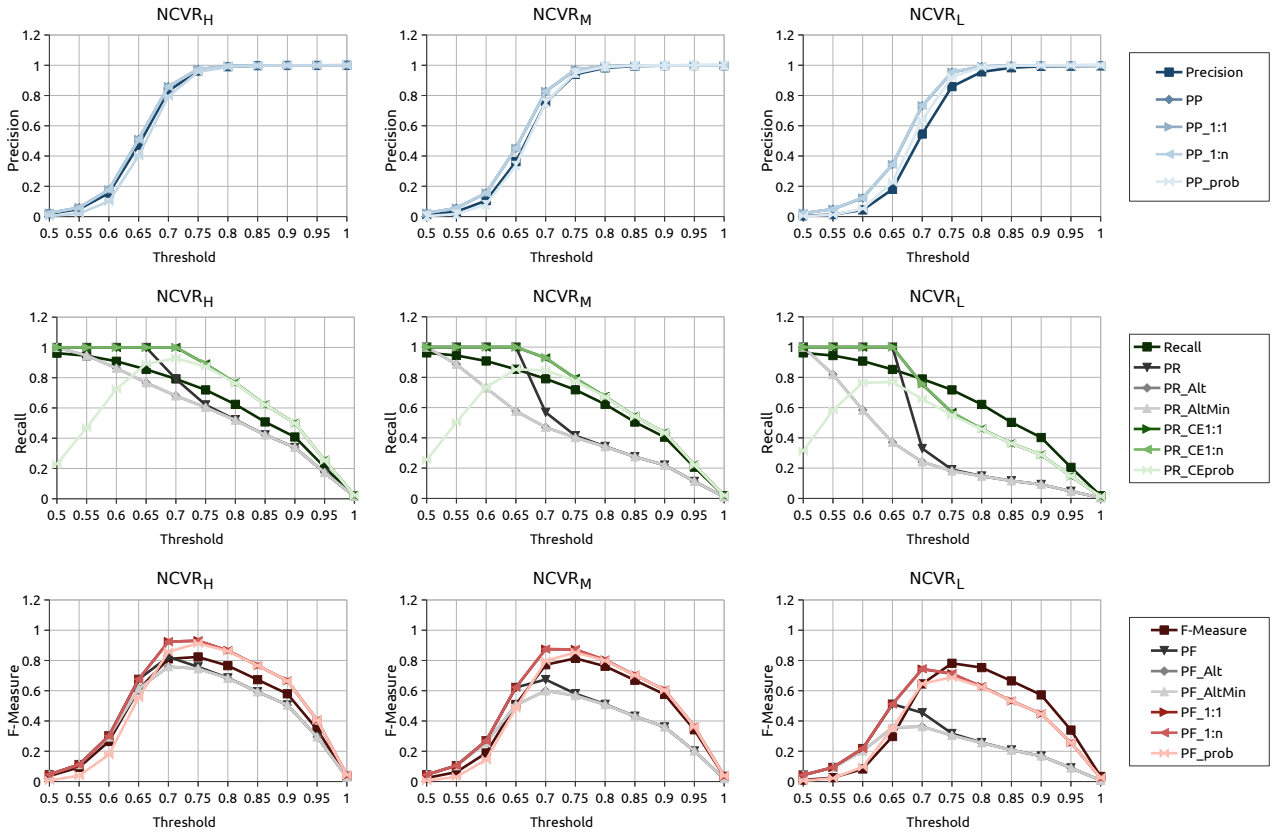

Figure 5: Results on NCVR datasets with different overlaps considering different thresholds to generate the similarity graph

.

for NCVR$_H$ (with 160,000 matches) range from roughly 117,000 (73.1%) to 153,000 (95.6%), for NCVR$_M$ (100,000 matches) from 76,000 to 124,000 and for NCVR$_L$ (40,000 matches) from 35,000 (87.5%) to 94,000 (235%). The configuration `[3FN, 3LN, YOB]` provides the best estimate with an average absolute difference of around 19,660 matches between the estimate and the actual number of matches over all datasets. To reduce the impact of different configurations to construct the private identifiers, we calculate the average estimated overlap over a set of configurations. Using this average leads to the best estimate with an average difference of only around 13,400 matches to the actual number of matches. We therefore use the average over the estimated overlaps as default in the following experiments.

We also compared the cryptoset approach against estimating the overlap using the private identifiers directly. The results (red bars) show that only the configuration `[2FN, 2LN, YOB]` leads to similar results compared to the cryptoset estimate. The non-private overlap estimation is more sensitive regarding the used configuration. The overhead of the encryption is negligibly small with an average runtime of 1.9s compared to 0.9s using the non-private estimation considering all configurations and datasets.

The results of the different quality estimation approaches for the NCVR dataset are shown in Fig. 5. The precision estimates for $NCVR_H$ (high overlap) are very close to each other and also to the real precision. With decreasing overlap the estimates PP, $PP_{1:1}$, and $PP_{1:n}$ are increasingly overestimating the actual precision. In such cases, PP_{prob} is providing better estimates. In terms of recall, the PR, PR_{Alt} and PR_{AltMin} are underestimating the actual recall, in particular for the datasets with medium and low overlap. Due to the same size of both data sources PR_{Alt} and PR_{AltMin} are equal.

The cryptoset approach, in contrast, provides estimates that are much closer to the actual recall, especially for NCVR$_M$ and NCVR$_L$. The recall estimates for PR_{CEprob} are dropping below a certain threshold which is caused by the small number of expected true positives for small thresholds using the probability-based estimation method. Using low thresholds results in graphs with low similarities and a high number of edges for each record. Thus, the number of true positives is underestimated if the threshold is too low and the difference to the optimal threshold is too high because of the high ambiguity in terms of the similarities of correct and incorrect matches. To avoid the effect of dropping recall estimates considering thresholds $t_1 < t_2$, the estimated recall for threshold t_1 can be bounded to the recall value obtained by threshold t_2 assuming that lower thresholds will not result in fewer true positives.

For dataset $NCVR_H$, all approaches provide estimates that are relatively close to the actual F-measure. Here, the approaches PF, PF_{Alt} and PF_{AltMin} (slightly) underestimate the actual F-measure, while the other approaches (slightly to moderately) overestimate the actual F-measure. The highest F-measure of 0.823 is reached for $t = 0.75$, followed closely by an F-measure of 0.811 for $t = 0.7$ and 0.766 for $t = 0.8$. Using the estimations for the threshold selection, the estimation methods $PF_{1:1}$, $PF_{1:n}$, and PF_{prob} lead to the

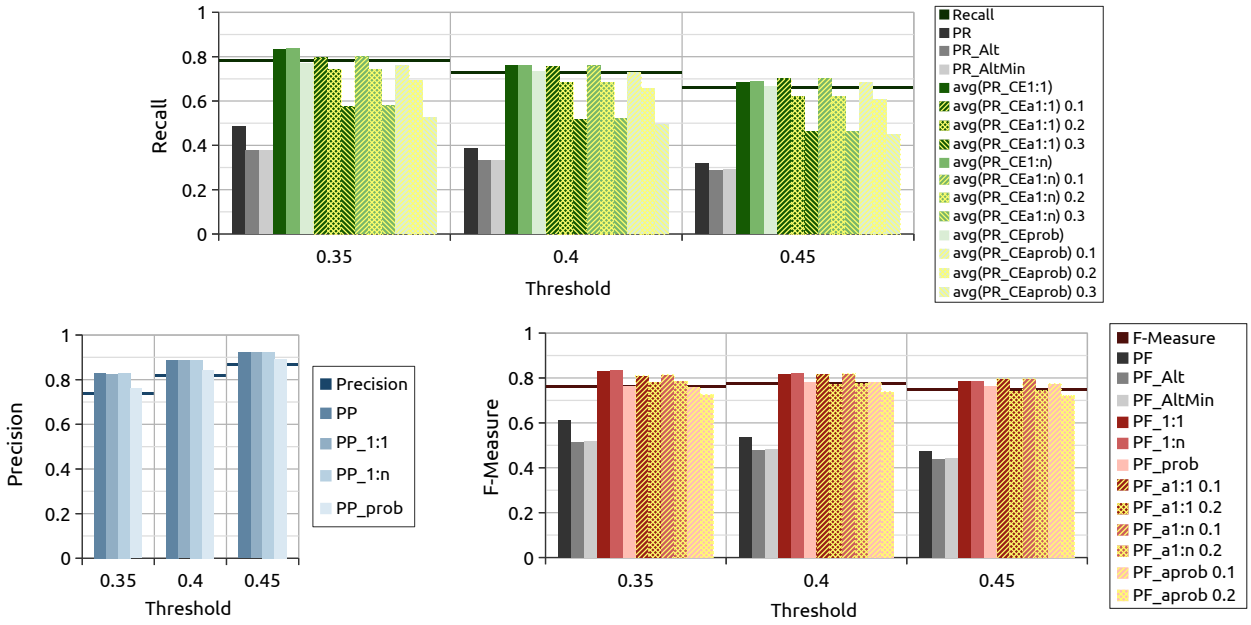

Figure 6: Results on MusicBrainz dataset. Results based on ground truth are shown as horizontal lines.

optimal threshold configuration. In contrast, the approaches PF, PF_{Alt} and PF_{AltMin} reach their maximum estimated F-measure at $t = 0.7$.

For $NCVR_M$, the estimates of PF, PF_{Alt} and PF_{AltMin} begin to diverge more from the actual F-measure. The F-measure is heavily underestimated for thresholds $t > 0.65$, with a maximum at $t = 0.7$, while the optimal threshold is at $t = 0.75$. This trend continues for dataset $NCVR_L$ where the estimates of PF, PF_{Alt} and PF_{AltMin} are even worse. For $NCVR_M$, PF_{prob} achieves the best estimates where the predicted F-measure slightly differs from the actual F-measure by at most 0.06. Considering the threshold selection, the estimation results in selecting the optimal threshold of $t = 0.75$.

In addition to the quality estimation, we analyzed the privacy of the cryptosets for the person datasets $NCVR_L$, $NCVR_M$ and $NCVR_H$. Each entry of a cryptoset is set by on average 24 elements (with a standard deviation ranging from 4.98 to 6.55). We also calculated information gain as described in Sect. 6 using the proposed identifier configurations and $L = 8192$. The information gain values are small, similar to the original work [46], and range from around 0.03 to 0.052 as shown in Tab. 2. Moreover, the more specific the private identifier is, the more evenly the public identifiers are distributed in the cryptoset resulting in a smaller information gain. We also observe that the information gain increases with a higher overlap, which is because the private identifiers are generated from a finite set of values (such as first/last names). Therefore, the number of identifiers mapped to one public identifier increases with the number of records in the overlap while the number of identifiers of non-overlapping records remains constant.

7.2.2 Heterogeneous and Dirty Databases. In contrast to the voter datasets, MusicBrainz and Dexter contain heterogeneous records regarding the characteristics of values. Therefore, we utilize our proposed automatic selection method to determine the private identifiers being utilized to estimate the overlap. To generate the private identifiers, we extract the first three characters from the value of an attribute. For the MusicBrainz dataset, we use all available attributes as candidates. For the Dexter dataset, we utilize a subset of the available attributes, such as product name, brand, and model. Both datasets consist of more than two data sources, we therefore calculate the macro precision and recall (the average of pairwise precision and recall values) [44].

The results for the MusicBrainz dataset are shown in Fig. 6. The recall estimates based on cryptosets in combination with the automatic generation of private identifiers are abbreviated with *'..a..'*. To determine the estimated overlap, we follow the same aggregation strategy as for the NCVR dataset, where we averaged the estimations regarding the generated private identifiers. Moreover, we compare the automatic selection method of the private identifiers with manually selected identifiers.

The cryptoset method in combination with the private identifier generation approach results in recall estimates where the average difference between the true and estimated recall values over all thresholds is below 0.03 for $t_{info} = 0.1$. In contrast to the cryptoset estimation, the baseline estimations PR, PR_{Alt} and PR_{AltMin} lead to an average difference ranging from around 0.33 (PR) up to around 0.4 (PR_{Alt}, PR_{AltMin}). t_{info} highly influences the quality of the cryptoset estimation resulting in different overlaps so that the recall differs up to 0.2 using $t_{info} = 0.3$ compared to $t_{info} = 0.1$. Comparing the automatic approach with the manually defined private identifiers, the method achieves comparable results using $t_{info} = 0.1$. However, due to the manual effort of selecting appropriate attribute combinations, we suggest using an automatic method.

Considering the estimates of precision, the probabilistic method PP_{prob} achieves the best results for the applied thresholds with differences below 0.018. In contrast to the probabilistic method, the baseline approach PP as well as the adapted $PP_{1:1}$ and $PP_{1:n}$ lead to similar estimates being far away from the real precision with a difference ranging from around 0.04 to around 0.09.

The combination of PP_{prob} and $PR_{CEaprob}$ to estimate precision and recall leads to the best F-measure approximation PF_{aprob} with an average difference of around 0.01 to the actual value. However,

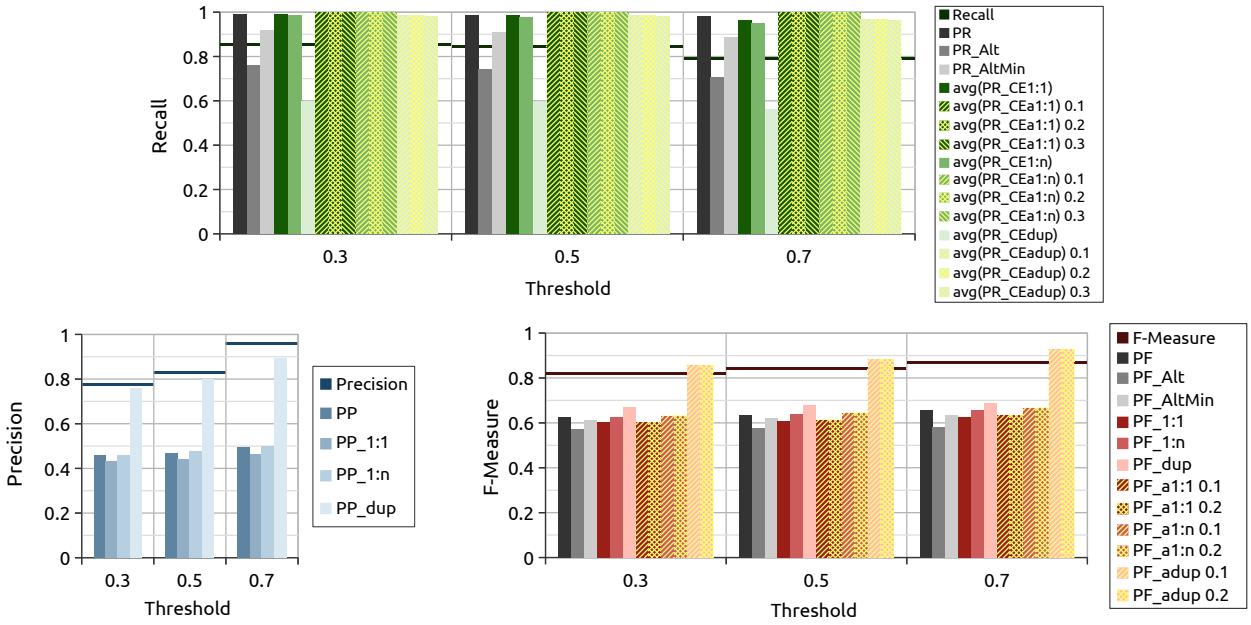

Figure 7: Results on Dexter dataset. Results based on ground truth are shown as horizontal lines.

due to the harmonic mean, $PP_{1:1}$ and $PR_{CEa1:1}$ with $t_{info} = 0.2$ achieve similar results regarding the F-measure with a difference of 0.011 because of the neutralization effect of an overestimated precision and an underestimated recall.

The Dexter dataset consists of heterogeneous data sources containing intra-source duplicates. Consequently, the previous methods for estimating the true positives will lead to inaccurate estimates since the number of true positives for each record is limited to one. Therefore, we apply our method based on the personalized PageRank to determine the precision estimate (PP_{dup}) that incorporates intra-source similarities as well. Due to the heterogeneity regarding various attributes and different types of quality issues, the manual selection of attributes for computing the private identifiers is a challenging task. Nevertheless, we use the brand and name attribute to determine the private identifiers resulting in reasonable results regarding a small manual effort. The results for the Dexter dataset are shown in Fig. 7.

Considering the estimations of recall using the cryptoset method in combination with the automatic private identifier generation, recall values are highly overestimated due to the underestimation of the overlap between the data sources. In contrast, the manually defined cryptoset methods highly underestimate recall values due to an overestimated overlap. The overestimation indicates that the private identifiers are not specific enough to represent records in this dataset. As expected, the baseline and the modified precision estimations utilizing the one-to-one assumption result in poor estimates being almost half of the actual precision. The precision estimate PP_{dup} achieves values that differ only slightly by at most 0.03 compared to the real precision for thresholds from 0.3 and 0.5. The accuracy of PP_{dup} is also reflected by the achieved F-measure estimates differing by at most 0.06.

Overall, our results show that the probabilistic and the personalized PageRank-based methods accurately estimate the number of true positives for clean and dirty databases. Moreover, the cryptoset-based approach improves the recall estimations significantly compared to the baseline approaches. The automatic generation of private identifiers for the cryptoset-based estimation leads to comparable results as the application of manually defined rules. However, for very heterogeneous datasets such as the Dexter dataset, our method does not always lead to accurate results showing the need for further work.

8 CONCLUSION

Typically, quality measures for record linkage results, such as precision and recall, are calculated based on ground truth data. However, in most real-world linkage scenarios such ground truth data is not available. A manual inspection of linkage results is also often not feasible, in particular, due to privacy constraints when linking sensitive data. In this paper, we presented different approaches for estimating the quality of a linkage result given in the form of a similarity graph. We showed that our methods outperform existing approaches and lead to accurate estimates on different datasets. These estimates can be used in practical applications to identify suitable linkage methods and to optimize their parameters, such as the classification threshold. In future work, we plan to investigate clustering-based approaches for estimating the linkage quality in deduplication scenarios.

AVAILABILITY

Reference code and datasets are available from our repository at https://git.informatik.uni-leipzig.de/dbs/pprl/primat and https://cloud.scadsai.uni-leipzig.de/index.php/s/NxQxaT4o9Z4TMDf.

ACKNOWLEDGMENTS

This work was supported by the German Federal Ministry of Education and Research by funding the "Center for Scalable Data Analytics and Artificial Intelligence Dresden/Leipzig" (ScaDS.AI). Peter Christen likes to acknowledge the support of the University of Leipzig and ScaDS.AI, Germany, where parts of this research were conducted while he was funded by the Leibniz Visiting Professorship.

REFERENCES

[1] R. Agrawal, A. V. Evfimievski, and R. Srikant. 2003. Information Sharing Across Private Databases. In *ACM SIGMOD International Conference on Management of Data*. ACM, 86–97. https://doi.org/10.1145/872757.872771

[2] Rakesh Agrawal and Ramakrishnan Srikant. 1994. Fast Algorithms for Mining Association Rules in Large Databases. (1994), 487–499. https://doi.org/10.5555/645920.672836

[3] Olivier Binette and Rebecca C. Steorts. 2022. (Almost) All of Entity Resolution. *Science Advances* 8, 12 (2022). https://doi.org/10.1126/sciadv.abi8021

[4] O. Binette, S. A. York, E. Hickerson, Y. Baek, S. Madhavan, and C. Jones. 2022. Estimating the Performance of Entity Resolution Algorithms: Lessons Learned Through PatentsView.org. https://doi.org/10.48550/ARXIV.2210.01230

[5] James H. Boyd, Tenniel Guiver, Sean M. Randall, Anna M. Ferrante, J. B. Semmens, Phil Anderson, and Teresa Dickinson. 2016. A simple sampling method for estimating the accuracy of large scale record linkage projects. *Methods of Information in Medicine* 55, 03 (2016), 276–283. https://doi.org/10.3414/ME15-01-0152

[6] P. Charalampopoulos, H. Chen, P. Christen, G. Loukides, N. Pisanti, S. P. Pissis, and J. Radoszewski. 2021. Pattern Masking for Dictionary Matching. In *Symposium on Algorithms and Computation ISAAC*. 1–19. https://doi.org/10.4230/LIPIcs.ISAAC.2021.65

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

[8] Peter Christen, David J. Hand, and Nishadi Kirielle. 2023. A Review of the F-Measure: Its History, Properties, Criticism, and Alternatives. *ACM Comput. Surv.* (2023). https://doi.org/10.1145/3606367

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

[10] Peter Christen and Rainer Schnell. 2023. Thirty-three myths and misconceptions about population data: From data capture and processing to linkage. *Int. Journal of Population Data Science* 8, 1 (2023). https://doi.org/10.23889/ijpds.v8i1.2115

[11] Reinhard Diestel. 2017. *Graph Theory* (5. ed.). Springer. https://doi.org/10.1007/978-3-662-53622-3

[12] James C. Doidge and Katie L. Harron. 2019. Reflections on modern methods: linkage error bias. *Int. J. Epidemiol* 48, 6 (2019), 2050–2060. https://doi.org/10.1093/ije/dyz203

[13] Rolf Egert, Marc Fischlin, David Gens, Sven Jacob, Matthias Senker, and Jörn Tillmanns. 2015. Privately Computing Set-Union and Set-Intersection Cardinality via Bloom Filters. In *Information Security and Privacy*. Vol. 9144. 413–430. https://doi.org/10.1007/978-3-319-19962-7_24

[14] Martin Franke, Ziad Sehili, Marcel Gladbach, and Erhard Rahm. 2018. Post-processing Methods for High Quality Privacy-Preserving Record Linkage. In *Data Privacy Management, Cryptocurrencies and Blockchain Technology*. Springer, 263–278. https://doi.org/10.1007/978-3-030-00305-0_19

[15] Martin Franke, Ziad Sehili, Marcel Gladbach, and Erhard Rahm. 2018. Post-Processing Methods for High Quality Privacy-Preserving Record Linkage. In *Data Privacy Management, Cryptocurrencies and Blockchain Technology*. Springer, 263–278. https://doi.org/10.1007/978-3-030-00305-0_19

[16] Michael J. Freedman, Kobbi Nissim, and Benny Pinkas. 2004. Efficient Private Matching and Set Intersection. In *Advances in Cryptology - EUROCRYPT 2004*. Vol. 3027. Springer, 1–19. https://doi.org/10.1007/978-3-540-24676-3_1

[17] A. Gkoulalas-Divanis, D. Vatsalan, D. Karapiperis, and M. Kantarcioglu. 2021. Modern Privacy-Preserving Record Linkage Techniques: An Overview. *IEEE TIFS* 16 (2021), 4966–4987. https://doi.org/10.1109/TIFS.2021.3114026

[18] Jiawei Han, Micheline Kamber, and Jian Pei. 2012. *Data Mining: Concepts and Techniques* (3. ed.). Morgan Kaufmann. https://doi.org/10.1016/C2009-0-61819-5

[19] David Hand and Peter Christen. 2018. A Note on Using the F-Measure for Evaluating Record Linkage Algorithms. *Statistics and Computing* 28, 3 (2018), 539–547. https://doi.org/10.1007/s11222-017-9746-6

[20] Katie Harron, James C. Doidge, and Harvey Goldstein. 2020. Assessing data linkage quality in cohort studies. *Annals of Human Biology* 47, 2 (2020), 218–226. https://doi.org/10.1080/03014460.2020.1742379

[21] Arvid Heise, Gjergji Kasneci, and Felix Naumann. 2014. Estimating the number and sizes of fuzzy-duplicate clusters. In *CIKM*. https://doi.org/10.1145/2661829.2661885

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

[23] David Holmes and M. Catherine McCabe. 2002. Improving Precision and Recall for Soundex Retrieval. In *Information Technology: Coding and Computing*. IEEE, 22–26. https://doi.org/10.1109/ITCC.2002.1000354

[24] Hye-Chung Kum, Ashok Krishnamurthy, Ashwin Machanavajjhala, Michael K Reiter, and Stanley Ahalt. 2014. Privacy preserving interactive record linkage (PPIRL). *Journal of the American Medical Informatics Association* 21, 2 (2014), 212–220. https://doi.org/10.1136/amiajnl-2013-002165

[25] Sumit Kumar Debnath, Pantelimon Stănică, Nibedita Kundu, and Tanmay Choudhury. 2021. Secure and Efficient Multiparty Private Set Intersection Cardinality. *Adv. Math. Commun.* 15, 2 (2021), 365–386. https://doi.org/10.3934/amc.2020071

[26] Bart Lamiroy and Tao Sun. 2011. Computing precision and recall with missing or uncertain ground truth. In *International Workshop on Graphics Recognition*. Springer, 149–162. https://doi.org/10.1007/978-3-642-36824-0_15

[27] Stefan Lerm, Alieh Saeedi, and Erhard Rahm. 2021. Extended Affinity Propagation Clustering for Multi-source Entity Resolution. In *Datenbanksysteme für Business, Technologie und Web (BTW)*. 217–236. https://doi.org/10.18420/btw2021-11

[28] Neil G. Marchant and Benjamin I. P. Rubinstein. 2017. In Search of an Entity Resolution OASIS: Optimal Asymptotic Sequential Importance Sampling. In *VLDB (11)*, Vol. 10. 12. https://doi.org/10.14778/3137628.3137642

[29] Cecilia L. Moore, Janaki Amin, Heather F. Gidding, and Matthew G. Law. 2014. A New Method for Assessing How Sensitivity and Specificity of Linkage Studies Affects Estimation. *PLOS ONE* 9, 7 (2014), 1–6. https://doi.org/10.1371/journal.pone.0103690

[30] Axel-Cyrille Ngonga Ngomo and Klaus Lyko. 2013. Unsupervised Learning of Link Specifications: Deterministic vs. Non-Deterministic. In *Proceedings of the Ontology Matching Workshop*. 12. https://doi.org/10.5555/2874493.2874496

[31] Andriy Nikolov, Mathieu d'Aquin, and Enrico Motta. 2012. Unsupervised Learning of Link Discovery Configuration. In *ESWC*. 119—-133. https://doi.org/10.1007/978-3-642-30284-8_15

[32] Daniel Obraczka, Alieh Saeedi, and Erhard Rahm. 2019. Knowledge Graph Completion with FAMER. *DI2KG Workshop at ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD)*. http://ceur-ws.org/Vol-2512/paper1.pdf

[33] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The PageRank Citation Ranking : Bringing Order to the Web. In *The Web Conference*.

[34] Fabian Panse, André Düjon, Wolfram Wingerath, and Benjamin Wollmer. 2021. Generating Realistic Test Datasets for Duplicate Detection at Scale Using Historical Voter Data. In *EDBT*. 570–581. https://doi.org/10.5441/002/edbt.2021.67

[35] George Papadakis, Vasilis Efthymiou, Emmanouil Thanos, and Oktie Hassanzadeh. 2022. Bipartite Graph Matching Algorithms for Clean-Clean Entity Resolution: An Empirical Evaluation. In *EDBT*. https://doi.org/10.48786/edbt.2022.41

[36] George Papadakis, Ekaterini Ioannou, Emanouil Thanos, and Themis Palpanas. 2021. The Four Generations of Entity Resolution. *Synthesis Lectures on Data Management* 16, 2 (2021), 1–170.

[37] George Papadakis, Dimitrios Skoutas, Emmanouil Thanos, and Themis Palpanas. 2020. Blocking and filtering techniques for entity resolution: A survey. *ACM Computing Surveys (CSUR)* 53, 2 (2020), 1–42. https://doi.org/10.1145/3377455

[38] Benny Pinkas, Mike Rosulek, Ni Trieu, and Avishay Yanai. 2019. SpOT-Light: Lightweight Private Set Intersection from Sparse OT Extension. In *Advances in Cryptology*. Vol. 11694. Springer, 401–431. https://doi.org/10.1007/978-3-030-26954-8_13

[39] Emmanouil Antonios Platanios, Avrim Blum, and Tom Mitchell. 2014. Estimating accuracy from unlabeled data. (2014). https://doi.org/10.1184/R1/6605273.v1

[40] Thilina Ranbaduge, Peter Christen, and Rainer Schnell. 2021. Large Scale Record Linkage in the Presence of Missing Data. *arXiv* (2021). https://doi.org/10.48550/arXiv.2104.09677

[41] Alieh Saeedi, Lucie David, and Erhard Rahm. 2021. Matching Entities from Multiple Sources with Hierarchical Agglomerative Clustering. In *IC3K*. SCITEPRESS, 40–50. https://doi.org/10.5220/0010649600003064

[42] Alieh Saeedi, Eric Peukert, and Erhard Rahm. 2018. Using Link Features for Entity Clustering in Knowledge Graphs. In *ESWC*, Vol. 10843. 576–592. https://doi.org/10.1007/978-3-319-93417-4_37

[43] Alieh Saeedi, Eric Peukert, and Erhard Rahm. 2020. Incremental Multi-source Entity Resolution for Knowledge Graph Completion. In *ESWC*, Vol. 12123. Springer, 393–408. https://doi.org/10.1007/978-3-030-49461-2_23

[44] Araken M Santos, Anne M P Canuto, and Antonino Feitosa Neto. 2010. Evaluating classification methods applied to multi-label tasks in different domains. In *HIS*. 61–66. https://doi.org/10.1109/HIS.2010.5600014

[45] Rainer Schnell, Tobias Bachteler, and Jörg Reiher. 2011. A Novel Error-Tolerant Anonymous Linking Code. *German Record Linkage Center, No. WP-GRLC-2011-02* (2011). https://doi.org/10.2139/ssrn.3549247

[46] S. Joshua Swamidass, Matthew Matlock, and Leon Rozenblit. 2015. Securely Measuring the Overlap between Private Datasets with Cryptosets. *PLOS ONE* 10, 2 (2015). https://doi.org/10.1371/journal.pone.0117898

[47] Dinusha Vatsalan, Peter Christen, and Vassilios S. Verykios. 2013. A Taxonomy of Privacy-Preserving Record Linkage Techniques. *Information Systems* 38, 6 (2013), 946–969. https://doi.org/10.1016/j.is.2012.11.005

[48] Dinusha Vatsalan, Ziad Sehili, Peter Christen, and Erhard Rahm. 2017. Privacy-Preserving Record Linkage for Big Data: Current Approaches and Research Challenges. In *Handbook of Big Data Technologies*. Springer, 851–895. https://doi.org/10.1007/978-3-319-49340-4_25

[49] Anushka Vidanage, Peter Christen, Thilina Ranbaduge, and Rainer Schnell. 2020. A Graph Matching Attack on Privacy-Preserving Record Linkage. In *CIKM*. 1485–1494. https://doi.org/10.1145/3340531.3411931

[50] Anushka Vidanage, Peter Christen, Thilina Ranbaduge, and Rainer Schnell. 2023. A Vulnerability Assessment Framework for Privacy-Preserving Record Linkage. *ACM Transactions on Privacy and Security* (2023). https://doi.org/10.1145/3589641

[51] 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