Journal of Information and Data Management, 2025, 16:1, doi: 10.5753/jidm.2025.4303

Exploiting Machine Learning Algorithms in the Classification Step of Record Linkage

Milena Macedo Santos [**Universidade Federal do Agreste de Pernambuco** | ***milenasantosmcd@gmail.com***]
Dimas Cassimiro Nascimento [**Universidade Federal do Agreste de Pernambuco** | ***dimas.cassimiro@ufape.edu.br***]

Federal University of Agreste de Pernambuco, Av. Bom Pastor, s/n, Boa Vista, Garanhuns, PE, 55292-270, Brazil.

Received: 22 March 2024 • **Published:** 22 August 2025

Abstract Record linkage is a well-known task that aims to determine duplicate pairs of records in datasets. In this work, we evaluated several Machine Learning-based classification algorithms (*Adaboost, MLP, SVM, Random Forest* and *XGboost*) in the context of record linkage. We conducted experiments which aimed to evaluate the influence of balanced and unbalanced training sets over the efficacy of the record linkage classification step. We also explore the usage of scatterplots to improve the qualitative discussion of the obtained experimental results. According to the obtained experimental results, the *Random Forest* algorithm has generated the highest F-measure considering the evaluated datasets. In addition, the *XGboost* model has also presented competitive results, especially in the context of bibliographic and movie datasets.

Keywords: Record Linkage, Machine Learning, Classification, Deduplication

1 Introduction

With the increase in technological advancements in recent decades, the amount of data generated by humans and systems has been growing exponentially. As a consequence, we often need to integrate a significant amount of data aiming to identify frauds, consolidate data in the demographic census, find duplicate products in e-commerce contexts, among other requirements. For this purpose, a task named Record Linkage (RL) is frequently used, which aims to identify records that represent the same real-world entity in one or more datasets, especially in the context in which these datasets do not present unique identifiers [Christen, 2012]. The RL process is crucial for several domains, such as security, health data integration, and consolidation of bibliographic data [de Souza Silva *et al*., 2017].

The RL workflow is usually divided into the following steps: **i) Pre-processing**, which is responsible for cleaning the data and usually aims to standardize schemes and data types used in attribute values; **ii) Indexing**, which reduces the number of comparisons between records in datasets, focusing on comparing pairs of records that have characteristics in common, e.g., records that share the same attribute value; **iii) Comparison**, which is responsible for comparing the pairs of records that are generated in the indexing step using similarity functions; **iv) Classification**, which aims to classify each record pair compared in the previous step as duplicate or non-duplicate; and **v) Quality assessment of results**, which aims to evaluate the number of pairs of records incorrectly and correctly classified as duplicates ([Christen, 2012]).

The usage of classification algorithms based on Machine Learning (ML) for Entity Resolution is a strategy that seeks to improve the correct classification of pairs of records by exploring intelligent models that make use of a training set. Training sets encompass examples of pairs of duplicate and non-duplicate records. ML techniques hold profound significance due to their capacity to streamline and enhance the accuracy of the RL process. This is because ML algorithms offer the ability to learn from patterns and relationships within the data, enabling automated decision-making in RL tasks. By leveraging ML, organizations can significantly reduce the manual effort required for data integration, while also improving the accuracy and scalability of the RL process. Furthermore, ML models can adapt to evolving data patterns and characteristics, ensuring robustness and efficiency in handling large and diverse datasets. Overall, the adoption of ML in RL not only automates the RL classification step, but also enhances the reliability and effectiveness of data integration efforts.

In the state of the art, we highlight a number of studies that investigate the use of ML models in the context of RL. For example, the work of [Ilangovan, 2019] aims to evaluate the performance of the *SVM* and *Random Forest* models, based on the use of heterogeneous datasets. Another work [Ramezani Foukolayi, 2021] compared the usage of the algorithms *Random Forest*, *Linear SVM*, *Radial SVM*, and *Dense Neural Networks* applied to RL. However, existing works do not focus extensively on investigating the influence of specific characteristics of datasets and training sets over the effectiveness of ML algorithms in the context of RL.

To fill this gap, the purpose of this research is to evaluate the effectiveness of five ML learning algorithms in the classification step of RL, encompassing several models already used in the literature, such as *Random Forest* and *SVM* ([Kaur *et al*., 2020; Ilangovan, 2019]), *MLP, XGboost*, and *AdaBoost*. Furthermore, our goal is to evaluate the influence of class balancing in the training sets and the dispersion of

similarity levels between pairs of records over the effectiveness of ML algorithms in the RL context.

In summary, this article presents three main contributions. First, we explore algorithms (*XGBoost* and *AdaBoost*) based on an ensemble of classifiers which have not been properly explored in the state of the art. Second, we investigate the influence of balanced training sets (regarding the number of duplicated and non-duplicated pairs of records) over the effectiveness of ML algorithms applied to RL. Third, we investigate a strategy based on the scatterplots to analyze the complexity of the datasets evaluated in the RL context and discuss how this graphical analysis relates to the efficacy of the ML algorithms.

From the experimental perspective, we aim to investigate three main research questions: i) Does the use of algorithms based on an ensemble of classifiers (AdaBoost and XGBoost) present improvements when compared to classic ML algorithms (SVM, Random Forests, and MLP) in the context of RL? ii) Does the generation of balanced training sets influence the training and classification steps of ML algorithms in the context of RL? iii) Is it possible to correlate the complexity of datasets in the RL context with the effectiveness of ML-based classifiers using scatterplots that present the distribution of similarities between pairs of records?

This article is an extended version of a previously published work [Santos and Nascimento, 2023], which has been extended in the following directions: i) we present a more comprehensive and detailed discussion of existing related works; ii) we provide more details regarding the conducted experiments; iii) we extended the formalization section, by highlighting the characteristics of the training sets which are investigated in this work; iv) we present more scatterplots, which are useful to enhance the qualitative discussion of the obtained experimental results; and v) we discuss final remarks associated with the obtained experimental results.

The remainder of this article is organized as follows. In Section 2, we present related works. In Section 3, we present the proposed approach, encompassing formalization, data pre-processing, and the process of selecting record pairs to compose training sets. In Section 4, we discuss the experimental evaluation and highlight the final remarks. Finally, in Section 5, we present the main conclusions of this work, as well as perspectives for future works.

2 Related Works

The usage of ML for RL aims to reduce the manual evaluation carried out by a human being to classify pairs of records that correspond to the same real-world entity. In the related literature, there are a significant number of contributions that aim to explore Machine Learning, Deep Learning, and Transformers to enhance the classification step of RL.

The authors of [Köpcke *et al*., 2010] assess various ML algorithms for RL on practical matching issues using real-world datasets. Their work considers factors such as algorithm complexity, scalability, and accuracy. Their findings contribute to a better understanding of the strengths and limitations of existing RL techniques in real-world contexts. In [Jurek-Loughrey and P, 2019], the authors explore methods for classifying record pairs in multi-source data linkage scenarios using semi-supervised and unsupervised approaches. Their approaches aim to improve the classification of record pairs without relying solely on manually labeled training data. Their article evaluates the performance of different algorithms and discusses their applicability in real-world data linkage scenarios. The investigation carried out in [Makri *et al*., 2022] proposes SVM-based approaches to improve the accuracy of RL by addressing issues of fairness, such as bias and discrimination. By refining the SVM-based record linkage process, their article seeks to achieve more reliable and equitable results in data integration tasks. Their findings contribute to improve the effectiveness and fairness of record linkage methods in various domains.

In [Pita *et al*., 2017], the authors aim to reduce the manual review efforts, since it represents a costly and unfeasible task depending on the amount of data to be processed. Their proposal starts by using a probabilistic method in pre-processing to find duplicate and non-duplicate examples. Then, Machine Learning models (*Decision Trees, Naive Bayes, Logistic Regression, Random Forest, Linear Support Vector Machines (SVM)* and *Gradient Boosted Trees*) are employed to classify the remaining pairs of records. In turn, the authors of [Comber and Arribas-Bel, 2019] aim to carry out the RL process using address datasets. To this end, they employ *XGboost, Random Forest* and *Logistic regression* in the context of RL. The authors argue that, when using *ensemble* algorithms, the model tends to generate a better result because the record pairs present in the classification step of RL do not allow separation into a hyperplane with high precision, since they are non-linear. The authors of [Andrzejewski *et al*., 2024] investigate how traditional statistical methods and modern ML algorithms perform in the context of RL. They evaluate factors such as accuracy, efficiency, and scalability of the investigated ML algorithms. Their findings present insights into the suitability of existing ML algorithms for different record linkage scenarios.

The usage of ML for ER is also investigated in the work of [Ramezani Foukolayi, 2021], in which a number of Machine Learning models are evaluated, enabling the reduction of manual review. They evaluate the algorithms *Random Forest, Linear SVM, Radial SVM* and *Dense Neural Networks*. Furthermore, they propose a methodology for transferring previously trained models to other datasets. Another work that presents a similar objective is presented in [Ilangovan, 2019], which seeks to analyze the performance of the algorithms *Random Forest, SVMs* and *Neural Nets* in the context of RL, encompassing the insertion of errors in datasets to generate heterogeneity.

To facilitate the usage of ML for RL, the work proposed in [Wang *et al*., 2021] focuses on streamlining the creation of RL solutions by leveraging ML techniques. They explore methods for automatically selecting and tuning algorithms, preprocessing data, and evaluating model performance. By automating these tasks, their work aims to reduce the time and effort required to develop accurate and scalable record linkage solutions. Their findings highlight the benefits of automation in improving the efficiency and effectiveness of RL model development processes.

Deep learning, with its complex data representation capa-

bilities, emerges as a promising approach to deal with the RL process. Recent works have pioneered the usage of Deep Learning in the context of RL. In [Li *et al*., 2020], the authors proposed an approach to apply pre-trained language models to RL. First, the data is pre-processed to remove noise, normalize variations, and extract relevant features of the records. Then, a pre-trained language model is used to learn latent representations of the data. These representations are then forwarded to a RL algorithm. Their experimental results demonstrate that the use of pre-trained language models significantly improves the accuracy and effectiveness of the RL process when compared to traditional approaches. The authors of [Mudgal *et al*., 2018] also explore the use of deep learning techniques in the context of RL. Regarding record representation, the authors discuss the importance of capturing relevant information, such as structured attributes, contextual information, and text representations. Various deep learning model architectures were explored, such as convolutional neural networks (CNNs), recurrent neural networks (RNNs), and Siamese neural networks. The approach proposed in [Nafa *et al*., 2022] explores the application of active deep learning techniques for RL, focusing on risk sampling strategies. Their study investigates how active learning, combined with deep learning methods, can improve the efficiency and accuracy of RL tasks. They propose risk sampling as a strategy for selecting informative data samples during the training process to enhance model performance. By leveraging active deep learning with risk sampling, the article aims to optimize the RL process, resulting in more accurate and scalable solutions.

More recently, several works have explored the usage of transformers-based approaches [Paganelli *et al*., 2022; Li *et al*., 2023], which have shown remarkable performance in natural language processing tasks, in the context of RL. By leveraging the powerful capabilities of transformers, such as attention mechanisms and contextual embeddings, these works aim to improve the quality of record linkage results while also addressing challenges related to scalability and computational resources. Based on the experimental results presented in [Paganelli *et al*., 2022; Li *et al*., 2023], their findings suggest that transformer-based approaches hold significant promise for advancing the state-of-the-art in RL, offering more effective solutions for data integration and reconciliation.

This work aims to evaluate several algorithms already explored for RL in related works by comparing the algorithms *SVM* and *Random Forest* with a neural network-based algorithm (*MLP*) and *boosting* algorithms, such as *XGboost* and *Adaboost*. Furthermore, this work aims to investigate the influence of the class balancing of the training sets over the effectiveness of the ML algorithms in the context of RL. Finally, we also investigate the usage of scatterplots to improve the qualitative discussion of the experimental results obtained in the context of RL.

3 Proposed Approach

In this section, we present the notation used throughout the article and describe the workflow proposed to select record pairs in order to produce training sets with distinct characteristics.

3.1 Formalization

Given two sets of records $A = \{a_1, a_2, \cdots, a_n\}$ and $B = \{b_1, b_2, \cdots, b_n\}$, the RL goal is to identify all pairs of records $(a, b) \in (A \times B)$, such that a and b represent the same entity in the real world. In this work, we assume that the data sets A and B were previously submitted to a schema alignment process and, consequently, have the same number of attributes.

The use of ML for RL is carried out using mathematical models that are represented by the function $f : X \rightarrow Y$, where X is the domain of the input values and Y is the output set that represents the existing classification classes (for RL, duplicate or non-duplicate).

The training set T is defined as a subset of $(A \times B \times S^m \times L)$, such that $m \leq n$, $S \in [0, 1]$ represents the similarity level of the pair of records in associated with a schema attribute and $L = \{0, 1\}$ is the set of possible classification labels for RL: 0 (not duplicated) or 1 (duplicated). After the similarity functions are applied to the attribute values of each pair of records that compose the training set, a final normalized similarity result is produced (i.e., between $[0, 1]$) associated with each pair of records, as illustrated in Figure 1.

	X				Y
Record pairs	title	year	authors	venue	label
(a_1, b_2)	1	0.8	0.7	0.9	1
(a_4, b_3)	0.3	0	0.8	1	0
(a_7, b_5)	0.8	0.8	0.7	0.9	1
(a_9, b_1)	0.5	1	0.7	1	0

Figure 1. Training set example.

The usage of an ML-based classifier in the RL process works as a function in the form $f : (A \times B \times T) \rightarrow L$, which aims to classify, based on the training set T, an unlabeled pair of records into one of the possible classifications of the set L: 0 or 1.

In this article, we are particularly interested in two characteristics of the training set: training set size and level of balancing. The first characteristic is related to the number of instances that compose the training set. In turn, the second characteristic is defined as the ratio between the number of positive instances (i.e., which are associated with $label = 0$) and negative instances (i.e., which are associated with $label = 1$) in the training set.

Since, in many real-world contexts, the training set labels must be generated manually, this may represent a costly and cumbersome process. Thus, we aim to evaluate the effectiveness of ML algorithms in the context of RL considering different training set sizes. Similarly, we also aim to investigate how the training set balancing influences the efficacy of the ML algorithms. This investigation is important to understand how the size and balance level of training sets should

be tuned for optimizing the performance and fairness of Machine Learning algorithms in record linkage tasks.

3.2 Selection of Record Pairs

In the process of selecting pairs to be used to compose the training and testing sets, we produce three subsets: the subsets A and B store the IDs of the records to be compared and subset C represents the ground truth of truly duplicate pairs of records, containing the IDs referring to records from subsets A and B. The selection of record pairs is carried out differently for duplicate and non-duplicate pairs of records, as shown in Figure 2.

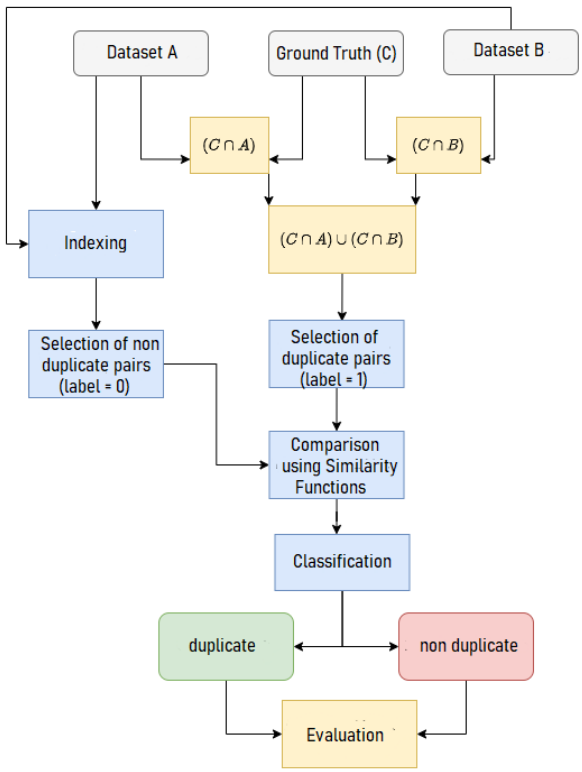

Figure 2. Workflow employed for selecting record pairs.

The workflow presented in Figure 2 uses the set C to ensure that all duplicate record pairs (i.e., label $= 1$) are present in the training or testing set. This is because the RL process deals with a classically unbalanced scenario, that is, the number of duplicate pairs in the datasets is much smaller than the number of non-duplicate pairs. For this reason, all pairs of duplicate records (i.e., record pairs in the set C) are selected for the evaluation process. This process is exemplified in Figure 3.

In turn, the selection of non-duplicate record pairs from the set $(A \times B)$ is carried out using a blocking step. To this end, the blocking technique is used to ensure that the training set is composed by both dissimilar and similar pairs of records, aiming to prevent the training of ML models from being negatively biased. To do this, one (or more) blocking key(s) are chosen to generate the blocks in the indexing phase. After carrying out the blocking process, we remove the duplicate pairs of records. This step is done by comparing the IDs of the pairs of records selected by the blocking technique with

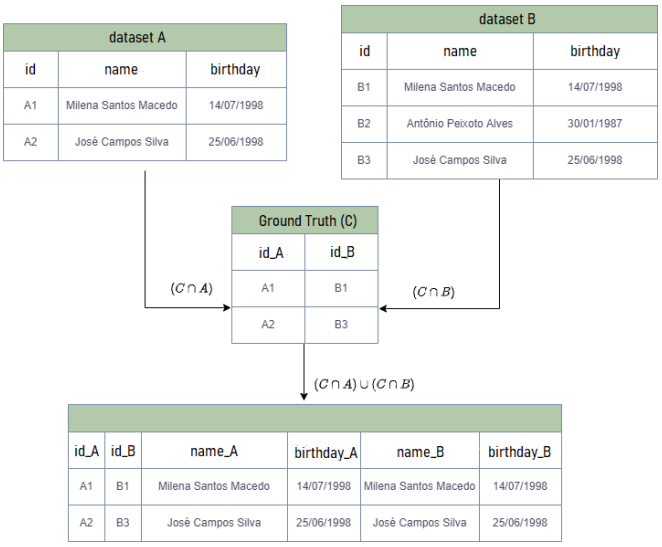

Figure 3. Selection of duplicate pairs of records to compose the training sets.

the IDs stored in the set C, which stores all pairs of duplicate records.

4 Experimental Evaluation

In this work, the objective of the experimental evaluation is to investigate the effectiveness of different ML algorithms in the RL classification stage. Furthermore, we aim to analyze the influence of the characteristics of the training set (balanced/unbalanced and training set size) as well as the similarity levels of the pairs of records to be classified over the effectiveness of the evaluated ML algorithms.

4.1 Datasets

Five pairs of datasets ([Köpcke *et al.*, 2010]) have been selected to conduct the experimental evaluation: two bibliographic datasets, two datasets that store products from the e-commerce context, and a dataset of movie data from IMBD and DBpedia. In Table 1, we report the size of the sets A, B, and C associated with each pair of datasets. These datasets are individually deduplicated, i.e., they represent an RL evaluation scenario called *Clean-Clean* [Papadakis *et al.*, 2013].

Datasets	|A|	|B|	|C|
DBLP-ACM	2294	2616	2224
DBLP-Scholar	2616	64263	5347
Amazon-Google Products	1363	3226	1300
Abt-buy	1081	1092	1097
D10 Movies (IMDB-DBpedia)	23183	27614	22864

Table 1. Statistics of the evaluated datasets.

The bibliographic datasets (*DBPL-ACM* e *DBLP-Scholar*) encompass the following attributes: title, authors, venue, and year. In turn, the product datasets (*Amazon-Google Products* and *Abt-buy*) present the following attributes: name, manufacturer, description, and price.

4.2 Data Pre-Processing

In the pre-processing stage, we performed a number of cleaning tasks over the data stored in the evaluated datasets. The following tasks were performed: i) removal of special characters; and ii) formatting of all strings in lowercase. The main objective of this step was to reduce noise and improve data quality for the RL process.

4.3 Metrics

To evaluate the ML models, we employed the F_1 measure, which represents the harmonic mean between the *precision* and *recall* metrics. The *precision* and *recall* metrics are calculated as follows:

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (3)$$

, such that TP is the number of True Positive record pairs, FP is the number of False Positive record pairs and FN is the number of False Negative record pairs.

4.4 Experimental Design

We designed two experiments to investigate the following research questions: **1**) What is the effectiveness produced by the five machine learning algorithms investigated in the context of RL, considering different levels of class balancing in the training set? and **2**) Based on the analysis of scatterplots, what is the influence of the dispersion of similarity levels between pairs of records over the effectiveness of the evaluated ML algorithms?

In the first experiment, two attributes from each dataset are considered. For the *DBLP-ACM* and *DBLP-Scholar* datasets, we employed the attributes title, venue, and authors. In turn, for the *Amazon-GoogleProducts* and *Abt-buy* datasets, we employed the attributes *product name* and *description*. To compare pairs of records, we used the Jaro-Winkler and Damerau-Levenshtein similarity functions, for each considered attribute. We also employed the Damerau-Levenshtein function to the year attribute. As a result, we produced seven columns of characteristics for the training set.

In turn, for the *Amazon-GoogleProducts* datasets, we employed three attributes: name, description, and price. We applied the Jaro-Winkler and Damerau-Levenshtein similarity functions to the name and description attributes. We also applied the Damerau-Levenshtein function to the price attribute. Thereby, we produced five columns of characteristics for the training set. In turn, for the *Abt-buy* datasets, we used the Jaro-Winkler and Damerau-Levenshtein similarity functions over the name and description attributes, producing four columns of characteristics for the training set. Lastly, for the *IMDB-DBpedia* datasets, we employed the Jaro-Winkler and Damerau-Levenshtein similarity functions over the title and aggregate value attributes.

We produced two different versions of the training set: i) unbalanced, in which the majority of pairs of records are non-duplicates; and ii) balanced, in which pairs of non-duplicate records are randomly eliminated, aiming to balance the ratio between pairs of duplicate and non-duplicate records, as shown in Table 2.

	Unbalanced		**Balanced**	
Datasets	**#Label 0**	**#Label 1**	**#Label 0**	**#Label 1**
DBLP-ACM	13641	5347	5347	5347
DBLP-Scholar	6984	2224	2224	2224
Amazon-GP	2844	1300	1300	1300
Abt-buy	1466	1097	1097	1097
IMDB-DBPedia	45113	14467	14467	14467

Table 2. Level of balancing of the evaluated training sets.

In relation to the labels 0 and 1 in Table 2, they refer to the number of non-duplicate and duplicate pairs of records, respectively. Furthermore, we considered three different sizes for the e-commerce and bibliographic training sets. In turn, for the e-commerce context, we generated training sets containing 10%, 30%, and 50% of the record pairs in the ground truth. For the Bibliographic and Movie datasets, we generated training sets encompassing 5%, 10%, and 25% of the record pairs in the ground truth.

5 Environment and Implementation

The conducted experiments were carried out on Google Colaboratory, known as Colab, a cloud service available as a product from Google. Colab is used to write and execute code in Python using a browser, aiming to facilitate the use of ML for data analysis. For the indexing process, the standard blocking technique was used to select duplicate and non-duplicate pairs (see Figure 2). The indexing implementation used the Python Record Linkage Toolkit library, version 0.15. We used Standard Blocking considering the most discriminative attribute of each pair of datasets. In turn, the execution of the ML algorithms used the scikit-learn Python library. The source code for running the experiments was implemented using jupyter notebook technology.

6 Results

In this section, we present the results obtained from the experimental evaluation. In Figures 4 and 5, we present the results of Experiment 1. The X and Y axes of Figures 4 and 5 represent the percentage of the size of the training set (in relation to the set sizes shown in Table 1) and the results of the F_1 metric reported by the ML algorithms, respectively. In turn, the results of Experiment 2 are shown in Figures 6 and 7.

6.1 Influence of Training Set Balancing

In Experiment 1, we investigate the impact of using balanced and unbalanced training sets over the effectiveness produced by ML algorithms. It is important to highlight that a typical RL scenario is characterized by unbalanced databases, i.e.,

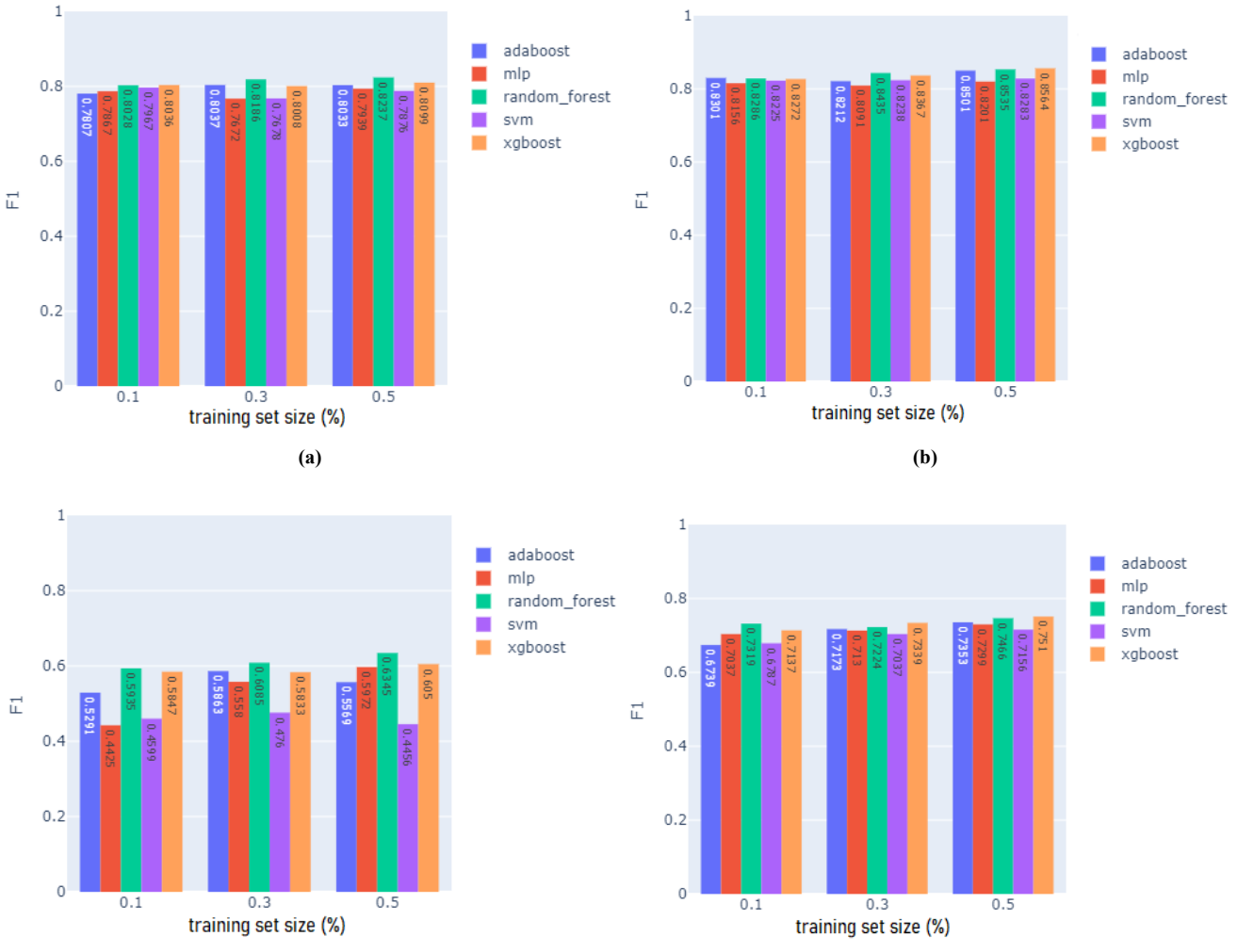

Figure 4. Efficacy results of ML algorithms, considering the following pairs of datasets: (a) *Abt-Buy* with unbalanced training set; (b) *abt-buy* with balanced training set; (c) *Amazon-GoogleProducts*, with unbalanced training set; (d) *Amazon-GoogleProducts*, with balanced training set.

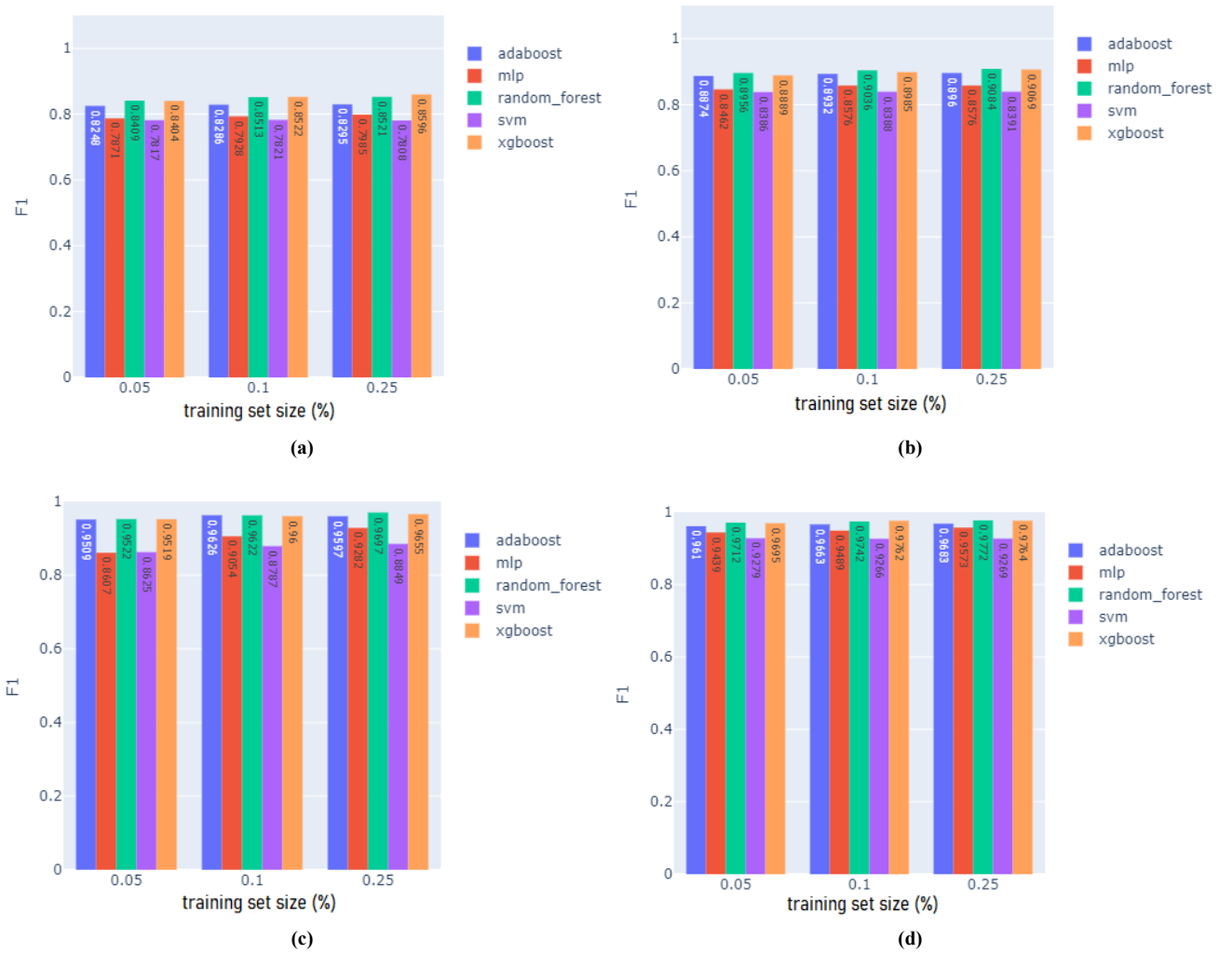

Figure 5. Efficacy results of ML algorithms, considering the following pairs of datasets: (a) *IMDB-DBpedia* with unbalanced training set; (b) *IMDB-DBpedia* with balanced training set; (c) *DBLB-ACM*, with unbalanced training set; (d) *DBLB-ACM*, with balanced training set.

the vast majority of record pairs are non-duplicate. For this reason, it is important to investigate the impact of training set balancing over the effectiveness of RL classification approaches.

Based on the experimental results (Figures 4 and 5), we notice that the use of balanced training sets produced gradual improvements over the results of the F_1 metric. For example, by analyzing the results in Figures 4(a)-(b), regarding the dataset pair *Abt-buy*, we can observe an increase of 5.10^{-2} regarding the F_1 metric when employing a balanced training set. In turn, in the experimental results reported in Figures 4(c)-(d), regarding the *Amazon-GoogleProducts* datasets, an even greater increase is observed (up to 10^{-1}) over the F_1 result, when using the balanced training set. Similarly, regarding the results produced using the Movie datasets (Figures 5(a)-(b)), we also note a moderate increase in F_1 when the algorithms employ a balanced training set. This phenomenon is observed for all evaluated ML algorithms.

We also noted a peculiarity for the *Amazon-GoogleProducts* datasets, as shown in Figures 4(c-d), which report the lowest results for the *F1* metric, especially when compared to the results produced by the pair of bibliographic datasets (Figures 5(c)-(d)). This result is explained by the fact that e-commerce datasets in the context of RL are more challenging, since there are many ways to represent the same product, regarding its description or its name. On the other hand, bibliographic datasets are simpler to be tackled by Machine Learning-based classification algorithms, since the pairs of duplicate records usually have higher similarity, which directly reflects the effectiveness of the classification algorithms.

Based on the results in Figures 4 and 5, we also noticed the influence of the size of the training set over the effectiveness results of the ML algorithms. Considering all evaluated datasets, the increase in the size of the training sets resulted in a moderate improvement in the result of the F_1 metric. In particular, in bibliographic datasets, the evaluated ML algorithms reported high F_1 results, even in the scenario where the smallest training set is used. In practical scenarios, manually generating a training set represents a slow and costly process. In this context, the experimental results indicate that, depending on the complexity of the pairs of records to be classified, we can train an effective classifier based on ML even if a reduced training set is employed. Finally, we notice that, in general, the *Random Forest* and *XGBoost* algorithms have presented promising F_1 results considering all evaluated datasets, obtaining a superior result compared to the other ML algorithms considered in the evaluation.

6.2 Dispersion of Similarity Levels

In Experiment 2, we explore scatterplots in order to explain the different levels of effectiveness produced by the ML-based classifiers. In the first experiment, the effectiveness results reported by the ML algorithms were higher for bibliographic and movie datasets and lower when processing the e-commerce datasets. This result is strongly correlated with the dispersion of similarity levels between pairs of records from the datasets considered in the experiments.

In the scatterplots reported in Figures 6 and 7, we can easily verify the association of characteristics produced by the similarities generated from the Damerau-Levenshtein and Jaro-Winkler distances, according to the attributes considered for each pair of datasets. In Figures 6(a)-(b) and 6(c)-(d), which report the levels of dispersion of similarities for pairs of datasets in the e-commerce context, we can observe a considerable overlap of pairs of duplicated (in red) and non-duplicated (in blue) records in regions of close similarity. This characteristic, evidenced by the scatterplot, makes the classification process carried out by the ML algorithms more complex. On the other hand, when analyzing the results reported in Figures 7(a)-(b) and 7(c)-(d), we can notice that, regarding pairs of bibliographic and movie datasets, there is a clearer separation between the similarity regions that encompass pairs of duplicate and non-duplicate records (when compared to the scatterplots of the e-commerce datasets), which facilitates the classification process carried out by the ML algorithm.

Therefore, the level of dispersion of pairs of records associated with different labels in the scatterplot can be considered to indicate the level of difficulty of the RL classification step. In other words, the presence of pairs of duplicate records that have a greater number of spelling errors (or a greater number of abbreviations or missing words) results in a decrease in similarity between the pairs of duplicate records (see Figure 6), which considerably increases the complexity of the classification stage. This is because the existence of pairs of duplicate records with low similarity increases the number of record pairs in the border [Peeters *et al*., 2023] in the classification stage, which represent pairs of records that are difficult to correctly classify.

Conversely, the presence of pairs of non-duplicate records with high similarity (as can also be seen in Figure 6) also increases the complexity factor associated with an RL classification task. This is because these pairs of records can negatively bias the training process of classification models, as reported by the authors of [Dal Bianco *et al*., 2018].

6.3 Final Remarks

This work aimed to investigate the influence of training set balancing and size over the efficacy of ML-based algorithms in the context of RL. Furthermore, we also analyzed the similarity dispersion of record pairs considering different similarity functions. We observed that the *XGboost* algorithm produced promising results, but *Random Forest* reported superior F_1 results when compared to the remaining algorithms.

We also highlight that the usage of scatterplots to visualize the similarity between pairs of duplicate and non-duplicate records is crucial for qualitatively discussing the experimental results of record linkage tasks based on ML algorithms. Scatterplots provide an intuitive and insightful representation of how well the algorithm distinguishes between duplicate and non-duplicate pairs of records. By plotting similarity scores (or other relevant metrics) for each pair of records, scatterplots may be used to present clustering patterns and discern the effectiveness of the ML algorithms in separating true matches from non-matches. Additionally, scatterplots facilitate the identification of potential outliers or misclassified instances, aiding in the interpretation and refine-

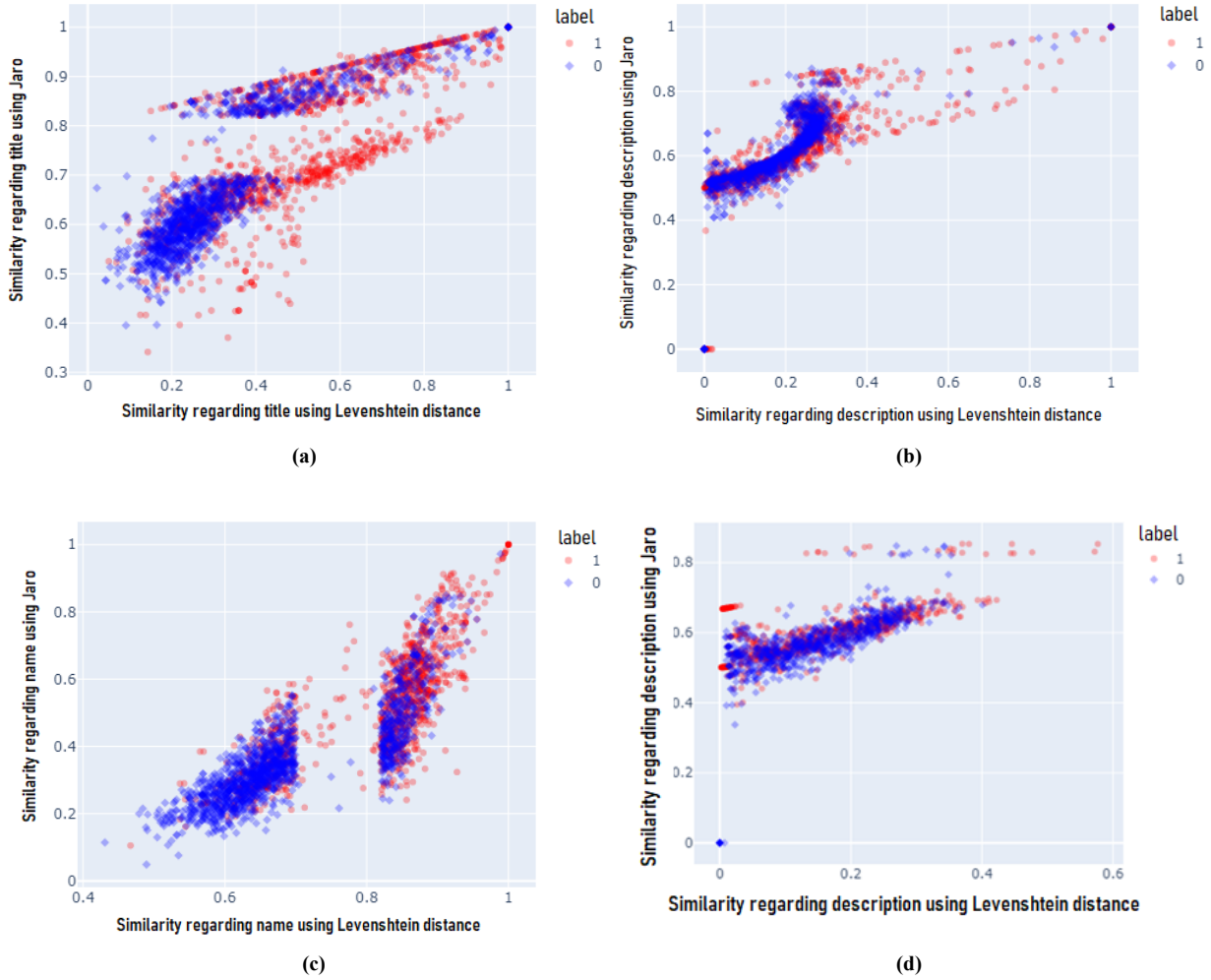

Figure 6. Similarity scatterplot (using the *Damerau-Levenshtein* and *Jaro-Winkler* functions) of pairs of records from the following databases: (a) *Amazon-GoogleProducts*, using the *title* attribute; (b) *Amazon-GoogleProducts*, using the *description* attribute; (c) *Abt-buy*, using the *title* attribute; (d) *Abt-buy*, using the *title* attribute.

ment of the Machine Learning model. This visual analysis enables Machine Learning developers to generate deeper insights into the algorithm's performance, identify opportunities for improvements, and make informed decisions regarding parameter tuning or feature selection to enhance the accuracy and fairness of Record Linkage results.

We also highlight the proximity of the results reported in Experiment 1. The evaluation reported in this article study was carried out considering a limited environment. However, if the investigated classification models are employed in the *Big Data* context, even small differences regarding F_1 percentage can significantly impact the results, i.e., even a small percentage of increase in F_1 may represent a significant number of duplicate pairs.

7 Conclusions and Future Works

In this work, we evaluated five ML algorithms (*Adaboost, MLP, SVM, Random Forest* and *XGboost*) in the classification stage of RL. Initially, we proposed a workflow to select duplicate and non-duplicate pairs of records to compose the training and testing sets. To evaluate the ML algorithms, we designed two experiments. In the first experiment, we investigated the influence of the level of balance in the training set over the effectiveness of the ML models. In the second experiment, we investigated the usage of scatterplots to better understand and discuss qualitatively the results produced by ML algorithms in the context of RL.

The experimental results suggest that the *Random Forest* and *XGBoost* models tend to perform better than the other ML algorithms in the context of the designed experiments. In turn, the *SVM* and *MLP* algorithms reported inferior results. We also concluded that the usage of balanced training sets tends to favor the effectiveness results reported by ML algorithms in the context of RL, producing an increase of more than 10% over the result of the F_1 metric. In turn, based on the analysis of scatterplots, we observe that pairs of records associated with different labels from bibliographic and movie datasets present less overlap in the similarity regions, which facilitates the classification stage. In turn, dataset pairs in the e-commerce context present much more challenging pairs of records, encompassing both pairs of du-

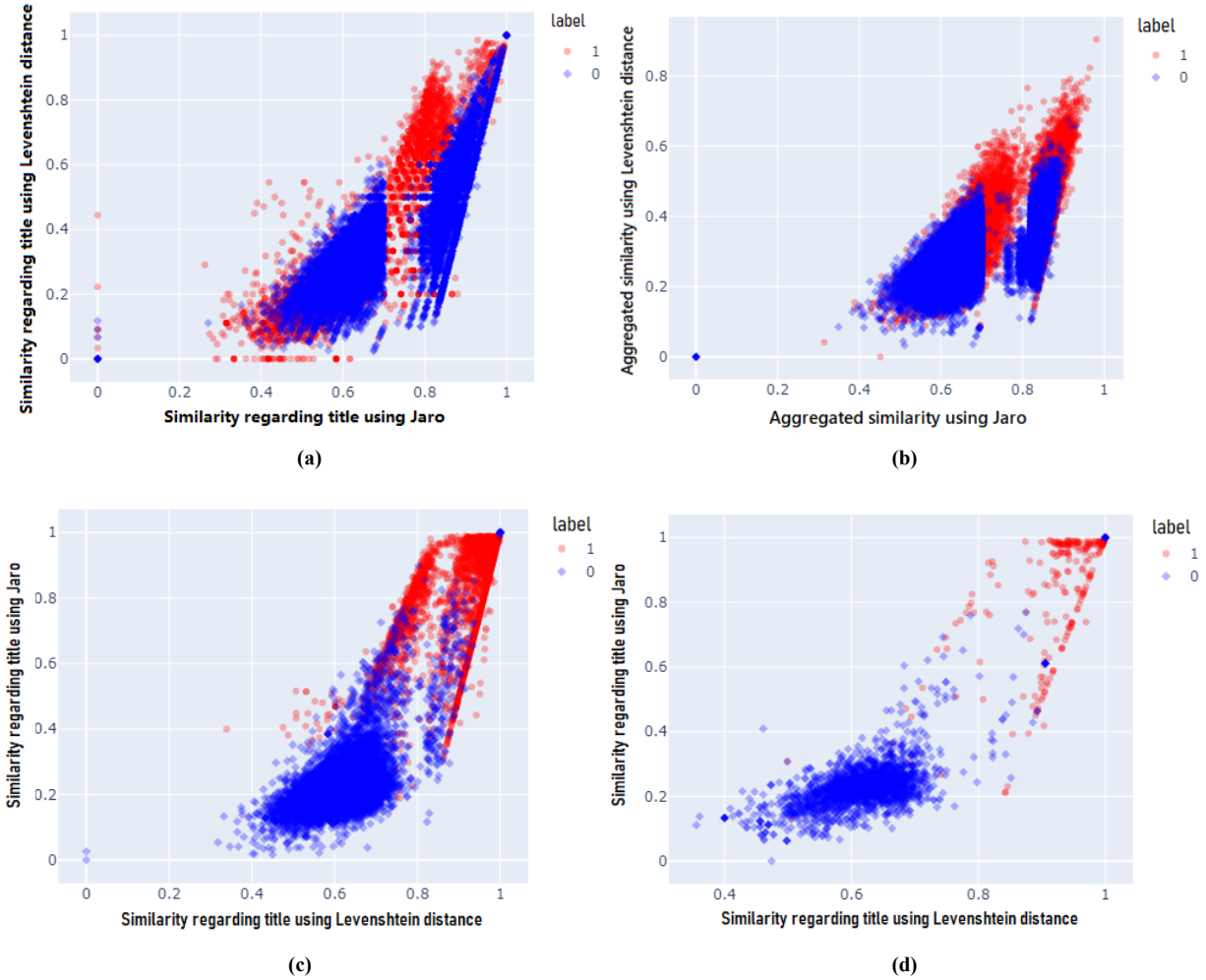

Figure 7. Similarity scatterplot (using the *Damerau-Levenshtein* and *Jaro-Winkler* functions) of pairs of records from the following databases: (a) *IMDB-DBPedia*, using the *title* attribute; (b) *IMDB-DBPedia*, using the *aggregated* similarity of the records pairs; (c) *DBLB-Scholar*, using the *name* attribute; (d) *DBLB-Scholar*, using the *description* attribute

plicate records with low similarity and pairs of non-duplicate records with high similarity, generating an evident overlap of border pairs, which makes it considerably difficult for ML algorithms to correctly classify pairs of records. Therefore, we can explore the analysis of similarity dispersion to estimate the level of complexity of an RL classification step.

For future works, we intend to evaluate the ML algorithms considering other datasets with distinct characteristics. Furthermore, we intend to incorporate recent approaches that consider *Transformers* [Li *et al*., 2023; Paganelli *et al*., 2022] and *Deep learning* [Mudgal *et al*., 2018; Li *et al*., 2020] in the experimental evaluation.

Competing interests

The authors declare that they have the following competing interests.

References

Andrzejewski, W., Bębel, B., Boiński, P., Kowalewska, J., Marszałek, A., and Wrembel, R. (2024). Statistical modeling vs. machine learning for deduplication of customer records (industrial paper).

Christen, P. (2012). *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Springer.

Comber, S. and Arribas-Bel, D. (2019). Machine learning innovations in address matching: A practical comparison of word2vec and crfs. *Transactions in GIS*, 23(2):334–348.

Dal Bianco, G., Gonçalves, M. A., and Duarte, D. (2018). Bloss: Effective meta-blocking with almost no effort. *Information Systems*, 75:75–89.

de Souza Silva, L., Nascimento Filho, D. C., and Moro, M. M. (2017). Uma avaliação de eficiência e eficácia da combinaçao de técnicas para deduplicaçao de dados. In *Anais do XXXII Simpósio Brasileiro de Bancos de Dados*, pages 160–171. SBC.

Ilangovan, G. (2019). Benchmarking the effectiveness and efficiency of machine learning algorithms for record linkage.

Jurek-Loughrey, A. and P, D. (2019). Semi-supervised and unsupervised approaches to record pairs classification in

multi-source data linkage. *Linking and Mining Heterogeneous and Multi-view Data*, pages 55–78.

Kaur, P. *et al*. (2020). A comparison of machine learning classifiers for use on historical record linkage.

Köpcke, H., Thor, A., and Rahm, E. (2010). Evaluation of entity resolution approaches on real-world match problems. *Proceedings of the VLDB Endowment*, 3(1-2):484–493.

Li, Y., Li, J., Suhara, Y., Doan, A., and Tan, W.-C. (2020). Deep entity matching with pre-trained language models. *arXiv preprint arXiv:2004.00584*.

Li, Y., Li, J., Suhara, Y., Doan, A., and Tan, W.-C. (2023). Effective entity matching with transformers. *The VLDB Journal*, 32(6):1215–1235.

Makri, C., Karakasidis, A., and Pitoura, E. (2022). Towards a more accurate and fair svm-based record linkage. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 4691–4699. IEEE.

Mudgal, S., Li, H., Rekatsinas, T., Doan, A., Park, Y., Krishnan, G., Deep, R., Arcaute, E., and Raghavendra, V. (2018). Deep learning for entity matching: A design space exploration. In *Proceedings of the 2018 International Conference on Management of Data*, pages 19–34.

Nafa, Y., Chen, Q., Chen, Z., Lu, X., He, H., Duan, T., and Li, Z. (2022). Active deep learning on entity resolution by risk sampling. *Knowledge-Based Systems*, 236:107729.

Paganelli, M., Del Buono, F., Baraldi, A., Guerra, F., *et al*. (2022). Analyzing how bert performs entity matching. *Proceedings of the VLDB Endowment*, 15(8):1726–1738.

Papadakis, G., Koutrika, G., Palpanas, T., and Nejdl, W. (2013). Meta-blocking: Taking entity resolutionto the next level. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1946–1960.

Peeters, R., Der, R. C., and Bizer, C. (2023). Wdc products: A multi-dimensional entity matching benchmark. *arXiv preprint arXiv:2301.09521*.

Pita, R., Mendonça, E., Reis, S., Barreto, M., and Denaxas, S. (2017). A machine learning trainable model to assess the accuracy of probabilistic record linkage. In *Big Data Analytics and Knowledge Discovery: 19th International Conference, DaWaK 2017, Lyon, France, August 28–31, 2017, Proceedings 19*, pages 214–227. Springer.

Ramezani Foukolayi, M. (2021). Comparison of machine learning algorithms in a human-computer hybrid record linkage system.

Santos, M. M. and Nascimento, D. C. (2023). Avaliando fatores de influência sobre algoritmos de aprendizado de máquina na etapa de classificação da resolução de entidades. In *Anais do XXXVIII Simpósio Brasileiro de Bancos de Dados*, pages 63–75. SBC.

Wang, P., Zheng, W., Wang, J., and Pei, J. (2021). Automating entity matching model development. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, pages 1296–1307. IEEE.