Informativeness-Based Active Learning for Entity Resolution

Victor Christen[1], Peter Christen[2], and Erhard Rahm[1]

[1] University of Leipzig, Leipzig, Germany.
{christen,rahm}@informatik.uni-leipzig.de
[2] Research School of Computer Science, The Australian National University, Canberra, Australia. peter.christen@anu.edu.au

Abstract. Entity Resolution is a crucial task to integrate data from different sources to identify records that represent the same entity. Entity resolution commonly employs supervised learning techniques based on training data of matching and non-matching pairs of records and their attribute similarities as represented by similarity vectors. To reduce the amount of manual labelling to generate suitable training data, we propose a novel active learning approach that does not require any prior knowledge about true matches and that is independent of the learning method used. Our approach successively identifies new training examples based on an informativeness measure for similarity vectors by considering their relationship to already classified vectors and the uncertainty in the similarity vector space covered by the current training set. Experiments on several data sets show that even for a small labelling effort our approach achieves comparable results to fully supervised approaches and it can outperform previous active learning approaches for entity resolution.

Keywords: Record linkage; entropy; uncertainty; interactive labelling.

1 Introduction

Entity Resolution (ER) is the task of identifying pairs of records from different data sources that refer to the same real-world entities [4]. ER is a crucial step for different application domains such as census analysis, national security, and the health, life, and social sciences. The quality and usefulness of any data analysis based on linked data highly depends upon how accurate ER was conducted.

To identify pairs of records that refer to the same entity, the attributes of records are generally compared using similarity functions such as approximate string comparators [4]. A crucial part of ER is the classification of two records as a *match* (same entity) or *non-match* (different entities) based on the calculated similarities between them. Machine learning approaches [13,23] can learn a classifier over sets of known matching and non-matching record pairs based on the similarities of their attributes as represented by a *similarity* or *weight vector*. For example, comparing first name, last name, street address, city and zipcode leads to a five-dimensional similarity vector per compared record pair [4].

To generate a classification model, labelled pairs of records are necessary. This however might require significant manual labelling efforts [26]. Moreover, the number of true matches (record pairs that refer to the same entity) is generally very small compared to the number of non-matching pairs because of the quadratic nature of the comparison space [4], and therefore the selection of labelled pairs is challenging if one wants to learn an unbiased classifier [6]. Active learning techniques promise to minimise the labelling effort as well as to select representative pairs that result in a good classifier.

Previous work in active learning for ER [1,2,19,26] has focused on selecting pairs based on a certain classification model and the resulting decision boundary of the learned classifier. In this paper, we propose a novel active learning approach for ER that considers the covered similarity vector space and the relationships between similarity vectors.

The main idea of our approach is to search for new unlabelled similarity vectors around *informative* similarity vectors that already are classified as matches or non-matches. In this process, we introduce an informativeness measure for a similarity vector based on the current training data set. The most informative vectors are then used to define a search space where new vectors are selected. We specifically make the following contributions:

- We propose an active learning technique for ER that iteratively selects new similarity vectors for manual classification by an oracle independent of any classifier using an informativeness measure. This measure is based on information entropy to characterise the relationship between vectors labelled as matches as well as non-matches. Moreover, the measure considers uncertainty so that new areas in the similarity vector space are queried.
- Our active learning technique is able to generate training data using a budget-limited human oracle [26], and it does not require any prior knowledge about true matches and non-matches.
- We evaluate our active learning technique on three data sets from different application domains. Our results show that our proposed approach outperforms a previous budget-limited active learning approach for ER [26] and achieves classification quality comparable to fully supervised approaches.

In the following we discuss work related to our approach. In Sect. 3 we formalise the problem that we aim to solve with our approach, which we describe in detail in Sect. 4. In Sect. 5 we then experimentally evaluate our approach and compare it with existing active learning as well as supervised methods for ER.

2 Related Work

ER is an essential part of data integration in various domains such as e-commerce, health and social science research, or national security. As a result, ER has been intensively studied [4,11,17,18]. One challenge of ER is the quality of the data sources and their heterogeneity [20]. In order to overcome this problem, supervised as well as unsupervised approaches have been proposed [3,13,23]. Unsupervised approaches utilise clustering methods to identify groups of similar records

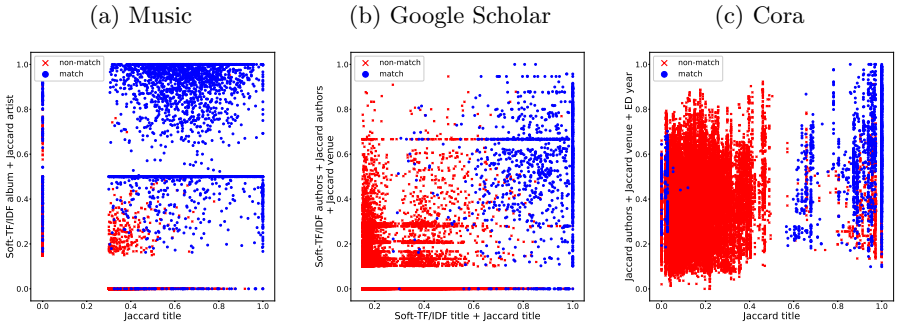


Fig. 1: Examples of similarity vectors where the monotonicity assumption does not hold. The three plots show similarity vectors of the data sets we use in our evaluation in Sect. 5. If an axis represents more than one similarity, they are summed and normalised into [0,1].

that refer to the same entity. In contrast, supervised ER approaches require and use a training data set consisting of verified true matches and true non-matches to build a classifier. In general, unsupervised methods perform worse than supervised approaches as shown by extensive studies [12], where supervised approaches are able to achieve high ER quality for different domains such as consumer products, bibliographic records, and census data.

A crucial part of supervised approaches is the amount and quality of data available for training, because a non-informative or not representative training data set can result in biased, over-fitted, or inaccurate classifiers.

To overcome such issues, active learning techniques [1,2,19,26] have been applied to minimise the labelling effort and to select representative record pairs for manual classification. An active learning approach is an iterative process [5] where in each iteration a number of informative and unlabelled training instances are selected that are then manually classified by a human oracle. Many active learning approaches determine informative instances using the distance between instances [25] or their entropy [21] according to a certain classification model.

Previous work in active learning for ER [1,2] allows to specify a minimum required precision threshold, where the aim of these approaches is to then maximise the recall of the resulting classifier based on the selected record pairs. However, these approaches have the underlying assumption of monotonicity of precision which implies that a record pair with higher similarity is more likely to be a match than a pair with a lower similarity.

Recent work by Wang et al. [26] however has shown that the assumption of monotonicity does not generally hold. We validate this in Fig. 1 which shows the distribution of true matches and non-matches for three data sets according to their similarities. As can be seen, in each data set there are clear examples that violate the monotonicity assumption. Therefore, Wang et al. proposed a cluster based active learning approach that iteratively selects record pairs from a cluster. In each iteration, a cluster is processed by selecting a set of record pairs

to be labelled by a human oracle. The labelled vectors are then added to the final training data set if the purity of the current cluster is above a user defined threshold. Otherwise, the cluster is split into two by classifying the unlabelled vectors of the current cluster based on the current classifier. The authors showed that their approach requires less examples than earlier active learning approaches for ER while achieving similar classification accuracy.

In comparison to our proposed approach, the selected examples by Wang et al. [26], and thus the resulting training data set, depend upon the applied classification model, and therefore the resulting ER quality can vary depending upon the classifier employed in this active learning approach.

Ngonga-Ngomo et al. [19] proposed a generation method of link specifications representing a complex match rule using genetic programming by iteratively improving a set of determined link specifications representing match rules. In each iteration, new examples are selected based on the disagreement according to the current link specification (for example, if 5 of 10 specifications classify a match for a record pair the disagreement is high). A disadvantage of this approach is that the generation of link specifications is not deterministic.

Related to active learning is crowd-sourced based ER [8,16,24,27], where ambiguous or controversial matches are resolved by evaluating votes from a crowd of human evaluators. Mozafari et al. [16] proposed two such approaches, named *Uncertainty* and *MinExpError*, being applicable for applications beyond ER. The main idea of these approaches is to use non-parametric bootstraping to estimate the uncertainty of classifiers. However, crowd-sourcing techniques that rely on a large number of human resources (often non-experts) cannot be used for sensitive data, such as personal health, financial, crime, or government records, where only a small number of experts have access to the data.

In contrast to previous work, our approach is independent of the classification model used to determine informative examples, because we characterise the informativeness of similarity vectors by considering the relationships between vectors within the vector space, as well as the relationships between unlabelled and already labelled vectors. Moreover, our work does not rely upon the monotonicity assumption that does not hold for many ER problems [26].

3 Problem Definition

Active learning approaches aim to reduce the manual efforts required for selecting training data, while keeping the quality of ER classification at a high level [1,2,26]. In general, the goal of ER is to identify matches $m_i \in \mathbf{M}$ for a set of records $\mathbf{R}$ from one or multiple data sources, where each $m_i = (r_x, r_y)$, with $r_x, r_y \in \mathbf{R}$ and $r_x \neq r_y$. To determine a match for a record pair (r_x, r_y), the set of attributes $\mathbf{A} = \{A_1, ..., A_n\}$ characterising these records is used to calculate similarities $s_1, ..., s_n$ between attribute values. Similarity functions $f_j(r_x.A_j, r_y.A_j)$, with $1 \leq j \leq n$, are used to measure how similar the values in attribute A_j are. We assume each similarity function f_j maps into $[0, 1]$, where 1 means two attribute values are the same and 0 means they are completely different [4].

A similarity or weight vector $\mathbf{w} \in [0,1]^n$ consists of the calculated n similarities between the attributes in $\mathbf{A}$. For example, the two records r_1 and r_2 characterised by the attributes $\mathbf{A} = \{surname, address\}$ with $r_1.surname$="*ashworth*", $r_1.address$="*fern hill*" and $r_2.surname$="*ashwort*", $r_2.address$="*fearn hill*" might results in a similarity vector $\mathbf{w} = \langle 0.74, 0.78 \rangle$ when using approximate string comparison functions such as edit distance [4].

The goal of an active learning approach is to identify a set of classified similarity vectors $\mathbf{T} \subset \mathbf{W}$ for a given set of unclassified vectors $\mathbf{W}$, where $\mathbf{T}$ consists of *matches* and *non-matches* and is used as training data to learn a classifier. Our approach considers a predefined budget b of the total number of similarity vectors that can be labelled by a human oracle. The approach selects in each iteration a predefined number k of vectors where the selection depends on the informativeness of each vector in $\mathbf{T}$ and the vector space covered by $\mathbf{T}$.

As detailed below, to measure the informativeness $info(\mathbf{w}_i, \mathbf{T})$, of a vector $\mathbf{w}_i$, we consider the relationship of $\mathbf{w}_i$ to vectors $\mathbf{w}_k \in \mathbf{T} \backslash \{\mathbf{w}_i\}$, where we calculate the similarity between two vectors $\mathbf{w}_i$ and $\mathbf{w}_k$ using the Cosine similarity defined as $sim(\mathbf{w}_i, \mathbf{w}_k) = \frac{\mathbf{w}_i \cdot \mathbf{w}_k}{||\mathbf{w}_i|| \cdot ||\mathbf{w}_k||}$. We assume that the area around a vector $\mathbf{w}_i$ consists of more informative vectors than for a vector $\mathbf{w}_k$, if $info(\mathbf{w}_i, \mathbf{T}) > info(\mathbf{w}_k, \mathbf{T})$. The area $S(\mathbf{w}_i)$ around $\mathbf{w}_i$ represents the search space for selecting new unclassified vectors, where $S(\mathbf{w}_i)$ consists of similarity vectors $\mathbf{w} \in \mathbf{W}$ and where the similarity $sim(\mathbf{w}_i, \mathbf{w})$ is above a certain threshold that is dynamically calculated according to the current training data set $\mathbf{T}$.

4 Informativeness-Aware Active Learning

In this section, we describe our active learning approach beginning with a high-level description. Algorithm 1 describes our informativeness-aware active learning approach for generating a training data set $\mathbf{T}$. This training data set is generated by selecting a number of similarity vectors from the set of all similarity vectors $\mathbf{W}$, where a total budget b is available for manual labelling of selected similarity vectors. The set of all (unlabelled) vectors $\mathbf{W}$ is generated by comparing record pairs based on the set of attributes $\mathbf{A}$ and appropriate similarity functions [4]. Initially, we select a number of similarity vectors $k > 1$ from $\mathbf{W}$ based on selection strategies such as *stratified sampling* or *farthest first* (line 1).

Throughout the learning process, we identify in each iteration a set of informative vectors $\mathbf{I} \subseteq \mathbf{T}$ according to the current training data set $\mathbf{T}$. The vectors in $\mathbf{I}$ are used to determine a search space for selecting k new vectors from $\mathbf{W}$ that are to be labelled by the oracle in the current iteration.

To identify the set $\mathbf{I}$, we characterise the informativeness of a vector considering its relationship to all vectors already in $\mathbf{T}$ (line 4). In particular, the informativeness $info(\mathbf{w}, \mathbf{T})$ of a vector $\mathbf{w} \in \mathbf{T}$ is calculated using an entropy-based measure considering the similarities to vectors of both the same and the other class. Moreover, $info(\mathbf{w}, \mathbf{T})$ considers the potential search space around $\mathbf{w}$ with respect to the labelled vectors from $\mathbf{T}$. We describe the calculation of informativeness for similarity vectors and their selection in Sect. 4.2 below.

Algorithm 1: Informativeness-Aware Active Learning Approach

Input:
- $\mathbf{W}$: Unlabelled similarity vectors
- b: Total manual labelling budget
- k: Number of similarity vectors to select in each iteration

Output:
- $\mathbf{T}$: Training data set in the form of labelled similarity vectors

```
1  T ←initialSelect (W, k)    // Select initial training data set
2  while |T| < b do
3      // Identify informative similarity vectors of the current training data set
4      I ←identifyInformativeVectors (T)
5      // Select unlabelled similarity vectors around informative vectors
6      W_o ←selectVectors (I, W, k, T)
7      T' ←manualClassify (W_o)    // Use oracle to classify selected vectors
8      T ← T ∪ T' // Add newly classified vectors to the overall training data set
9      W ← W \ W_o   // Remove classified vectors from set of unlabelled vectors
10 return T
```

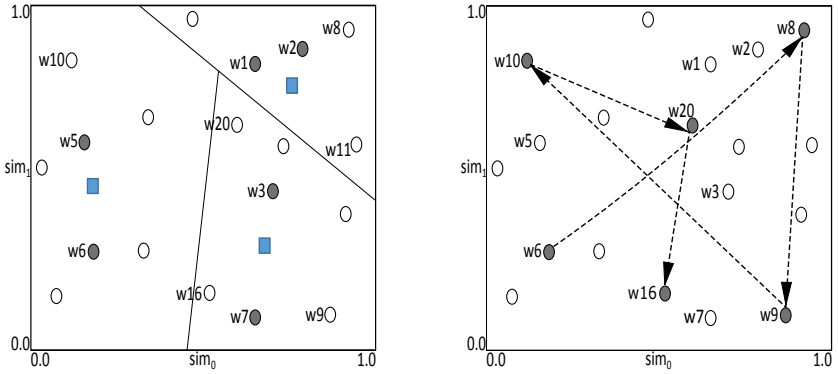
For each similarity vector in $\mathbf{I}$, we determine a search space based on its location in the similarity vector space and the location of the closest similarity vector in the opposite class as determined by the Cosine similarity. We consider each unlabelled vector contained in the search space as a candidate (line 6). The idea of the selection process is to identify similarity vectors in uncertain areas that are close to the boundary of matches and non-matches. The identified set of similarity vectors $\mathbf{W}_o$ is then manually classified by the oracle and added as $\mathbf{T}'$ to the total training data set $\mathbf{T}$ (lines 7 and 8). The approach terminates once the number of classified similarity vectors reaches the total budget b. In the following, we describe the initial selection strategies, the computation of informativeness, and the identification of new training vectors in more detail.

4.1 Initial Selection

Initially, we select a set of similarity vectors from the set of all unclassified vectors $\mathbf{W}$. We propose two strategies: *stratified sampling* and *farthest first* [26].

Stratified sampling splits the set of similarity vectors $\mathbf{W}$ into several partitions $\{\mathbf{P}_1, .., \mathbf{P}_x\}$. To determine an appropriate number of partitions, x, we apply canopy clustering [15] on the unlabelled similarity vectors $\mathbf{W}$. The generated partitions are used to determine the set of k initial similarity vectors. We iteratively select similarity vectors over the x partitions, where in each iteration we select the vector $\mathbf{w}_i$ of partition $\mathbf{P}_i$ that is the closest vector to its cluster centroid, and add $\mathbf{w}_i$ to $\mathbf{T}$. After that, we remove $\mathbf{w}_i$ from partition $\mathbf{P}_i$. The process terminates once the number of selected similarity vectors is k.

On the other hand, the farthest first method [26] initially selects a similarity vector at random from $\mathbf{W}$ and adds it to $\mathbf{T}$. After that, we iteratively add



(a) Stratified sampling with $x = 3$ (b) Farthest first

Fig. 2: Examples of initial selection strategies for $k = 6$. The grey circles represent the selected similarity vectors while squares show the centroids of each partition.

another similarity vector to $\mathbf{T}$ that has the maximum distance to all vectors already in $\mathbf{T}$. We repeat this process until $\mathbf{T}$ contains k similarity vectors.

For example, in Fig. 2a, stratified sampling selects the similarity vectors $w1$, $w2$, $w3$, $w5$, $w6$ and $w7$. The vector space is initially split into $x = 3$ partitions. After that, for each centroid (blue squares) of a partition we select the closest two similarity vectors. In Fig. 2b, the farthest first approach randomly selects, for example, $w6$ as the first similarity vector and adds it to $\mathbf{T}$. After that, $w8$ is selected since it is the vector farthest away from $w6$. The next selected vectors are $w9$, $w10$, $w20$, and $w16$, following the same process.

4.2 Informativeness of Similarity Vectors

In order to generate a representative training data set, we propose a selection approach that considers the informativeness of similarity vectors $\mathbf{w} \in \mathbf{T}$. The goal is to determine informative classified vectors that can be used to select unclassified vectors from $\mathbf{W}$. We describe the informativeness of a similarity vector by considering its location with respect to the vectors of the same as well as vectors from the other class in the vector space. The intuition is that we look for new vectors in the areas of classified vectors that are not outliers (i.e. are not surrounded only by vectors from the other class) but are also not easy to classify vectors (i.e. are not surrounded only by vectors from the same class).

To determine informative vectors of the current training data set $\mathbf{T}$, we define the following measure $info(\mathbf{w}_j, \mathbf{T})$, as shown in Eqn. (1), for a classified vector $\mathbf{w}_j \in \mathbf{T}$, where sim is the Cosine similarity as described in Sect. 3. This measure is based on the entropy of a vector $\mathbf{w}_j$ according to all vectors in $\mathbf{T}$ and the uncertainty of a vector $\mathbf{w}_j$ Entropy and uncertainty are equally weighted when $\alpha = 0.5$.

$$info(\mathbf{w}_j, \mathbf{T}) = \alpha \cdot entropy(\mathbf{w}_j, \mathbf{T}) + (1 - \alpha) \cdot uncertainty(\mathbf{w}_j, \mathbf{T}) \qquad (1)$$

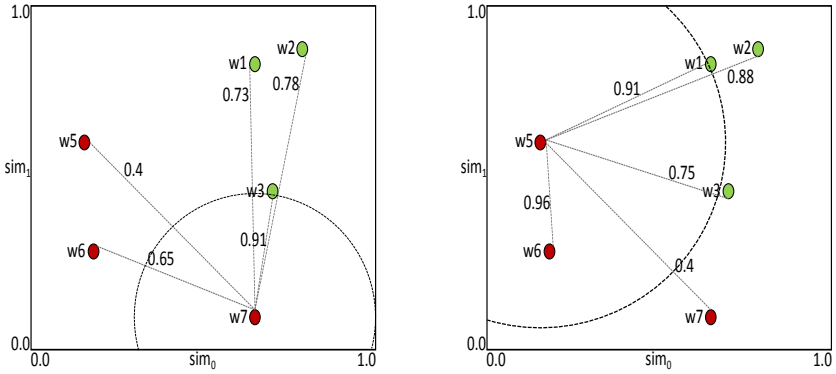


Fig. 3: Two examples for determining the informativeness of similarity vectors $w5$ and $w7$ of $\mathbf{T}$=$\{w1, w2, w3, w5, w6, w7\}$, based on the location in the vector space and the search spaces $S(w5)$ and $S(w7)$ for $w5$ and $w7$, as represented by the circles. Red coloured circles represent classified non-match similarity vectors while green coloured circles represent classified match vectors.

Information entropy [22] can be used to describe how balanced a data set is. In our case, the entropy of a vector $\mathbf{w}_j$ is high if it is close to vectors representing both matches as well as non matches. To determine the entropy of $\mathbf{w}_j$, we compute the aggregated similarities between $\mathbf{w}_j$ and each vector $\mathbf{w}_k$ of $\mathbf{T}_S^{w_j}$ and $\mathbf{T}_O^{w_j}$, where $\mathbf{T}_S^{w_j}$ and $\mathbf{T}_O^{w_j}$ consist of vectors that are assigned to the same class and the other class, respectively, according to $\mathbf{w}_j$, as shown in Eqn. (2).

$$entropy(\mathbf{w}_j, \mathbf{T}) = -\Big[\frac{\sum_{\mathbf{w}_k \in \mathbf{T}_S^{w_j}} sim(\mathbf{w}_j, \mathbf{w}_k)}{|\mathbf{T}|-1} \cdot log(\frac{\sum_{\mathbf{w}_k \in \mathbf{T}_S^{w_j}} sim(\mathbf{w}_j, \mathbf{w}_k)}{|\mathbf{T}|-1}) + \frac{\sum_{\mathbf{w}_k \in \mathbf{T}_O^{w_j}} sim(\mathbf{w}_j, \mathbf{w}_k)}{|\mathbf{T}|} \cdot log(\frac{\sum_{\mathbf{w}_k \in \mathbf{T}_O^{w_j}} sim(\mathbf{w}_j, \mathbf{w}_k)}{|\mathbf{T}|})\Big] \quad (2)$$

The uncertainty of a vector $\mathbf{w_j}$ is determined by the reciprocal of the intersection between the current training data set $\mathbf{T}$ and the search space determined as the area between $\mathbf{w}_j$ and the closest vector of the opposite class as shown in Eqn. (3).

$$uncertainty(\mathbf{w}_j, \mathbf{T}) = \frac{1}{1 + |\mathbf{T} \cap S(\mathbf{w}_j)|} \quad (3)$$

For example, the entropy of $w7$ in Fig. 3 is 0.68 calculated by Eqn. (2) utilising the aggregated similarity to vectors of the same class ($w6$ and $w5$) as $0.65 + 0.4 = 1.05$, as well as to vectors of the other class ($w1$, $w3$ and $w2$) as $0.73 + 0.91 + 0.78 = 2.42$. The intersection between the search space $S(w7)$ and the current training data set $\mathbf{T}$ is empty and therefore $uncertainty(w7) = 1$. Consequently, $info(w7)$ is equal to $0.5 \cdot 0.68 + 0.5 \cdot 1 = 0.84$. The informativeness for $w5$ is calculated similarly where its entropy is 0.697 and its uncertainty is 0.5 since $S(\mathbf{w_5}) \cap T = \{w_6\}$, and therefore $info(w5, \mathbf{T}) = 0.6$.

Algorithm 2: Selection Method of New Similarity Vectors

Input:
- **I**: Set of informative similarity vectors
- **T** Current classified training data set
- **W**: Set of unlabelled similarity vectors
- k: Number of similarity vectors to be selected

Output:
- $\mathbf{W}_o$: Similarity vectors selected for manual classification by oracle

```
1  C = ∅    // Initialise empty set of candidates
2  foreach w_j ∈ I do
3  |  // Determine vector being closest to w_j from the opposite class
4  |  w_c ← getClosest (w_j, T)
5  |  δ ← sim(w_j, w_c) // Calculate threshold representing the search space of w_j
6  |  foreach w_u ∈ W do
7  |  |  // Add unlabelled vector if its similarity is above the threshold δ
8  |  |  if sim(w_u, w_j) > δ then
9  |  |  |  C ← C ∪ {w_u}
10 // Identify the k most diverse vectors from candidate set
11 W_o ← farthestFirstSelection (C, k)
12 return W_o
```

We add a vector $\mathbf{w}_j$ to $\mathbf{I}$ if $info(\mathbf{w}_j, \mathbf{T})$ is above the mean according to the $info$ measure for the vectors of the current training data set $\mathbf{T}$. In our running example, the mean of $info$ according to the current training data set is 0.61, and so we add $w7$ ($info = 0.84$) to $\mathbf{I}$, but not $w5$. The set $\mathbf{I}$ of informative vectors is then used to select vectors of $\mathbf{W}$ to be manually classified and added to $\mathbf{T}$.

4.3 Training Data Selection

The selection method shown in Algo. 2 determines for each similarity vector of $\mathbf{I}$ a set of unlabelled vectors from $\mathbf{W}$. For this, we identify for each vector $\mathbf{w}_j \in \mathbf{I}$ a search space $S(\mathbf{w}_j)$ determined by the closest vector $\mathbf{w}_c$ from the opposite class. For example, in Fig. 4 the closest vector from the other class for $w7$ is $w3$.

The objective is to identify new vectors in uncertain areas so that in each iteration an increasingly more representative training data set $\mathbf{T}$ is generated. A vector $\mathbf{w}_u \in \mathbf{W}$ is added to the set $\mathbf{C}$ of candidates if it is contained in the search space $S(\mathbf{w}_j)$ consisting of vectors $\mathbf{w}_u$ where the similarity $sim(\mathbf{w}_j, \mathbf{w}_u)$ is larger than $sim(\mathbf{w}_j, \mathbf{w}_c)$ (line 9). At the end of the selection method, we determine the most k-diverse vectors of $\mathbf{C}$ by applying a farthest first approach (line 11).

Fig. 4 shows an example for selecting vectors based on w_3 and w_7. The selection method selects all vectors as candidates into $\mathbf{C}$ that are in the search spaces $S(w7)$ and $S(w3)$, shown as circles around $w3$ and $w7$. Consequently, the combined candidate set, $\mathbf{C}$, based on $w7$ and $w3$ consists of the similarity vectors $w9$, $w11$, $w16$, $w18$, $w19$ and $w20$.

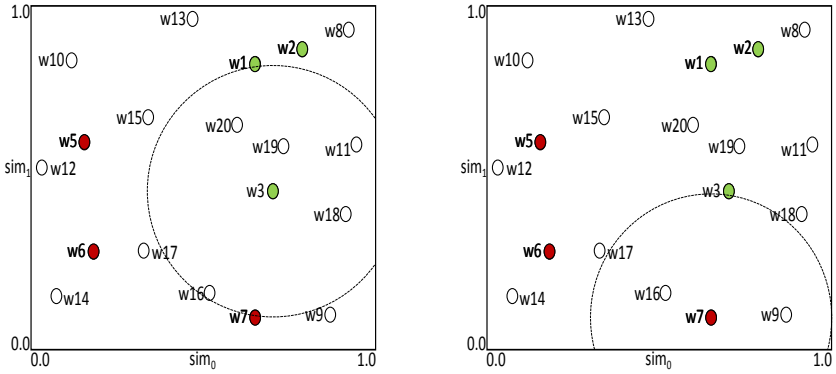


Fig. 4: Two examples of selecting new similarity vectors according to the search spaces $S(w3)$ and $S(w7)$ represented as circles, where $w3$ and $w7$ are the informative vectors. Red and green coloured circles represent classified vectors.

The identified set of similarity vectors $\mathbf{W}_o$ are then manually classified by an oracle and added to $\mathbf{T}$ (Algo. 1, line 8). The updated training data set is used in the next iteration to identify a new set of informative vectors. This loop ends once the number of manually classified similarity vectors reaches the budget b.

4.4 Complexity Analysis

We now briefly discuss the complexity of our proposed approach. Because of the independence of our approach with regard to the actual classification model used, its complexity only depends upon the number of unlabelled similarity vectors, $\mathbf{W}$, the total budget $\mathbf{b}$, and the number k of similarity vectors to be selected in each iteration. In each iteration, we compute the similarities between all pairs of vectors in the current training data set, $\mathbf{T}$, resulting in a complexity of $O(|\mathbf{T}|^2)$. Moreover, we identify for each informative similarity vector of $\mathbf{I}$ the closest unlabelled similarity vectors in $\mathbf{W}$, a process which requires $|\mathbf{W}| \cdot |\mathbf{I}|$ comparisons where $|\mathbf{I}| \leq |\mathbf{T}|$ holds. At the end of each iteration, we determine the k most diverse similarity vectors of $\mathbf{C}$, where $|\mathbf{C}| \leq |\mathbf{W}|$, resulting in a complexity $O(k \cdot |\mathbf{C}|)$. Overall, the complexity to determine similarity vectors for one iteration is $O(|\mathbf{T}|^2 + |\mathbf{W}| \cdot |\mathbf{I}| + k \cdot |\mathbf{C}|)$, with $|\mathbf{I}| \leq |\mathbf{T}|$ and $|\mathbf{C}| \leq |\mathbf{W}|$. The number of iterations is bound by k and b as b/k.

5 Experiments and Results

We evaluated our active learning approach using three data sets as summarised in Table 1. The Cora and Google Scholar (GS) [12] data sets contain publication records that are to be linked, where the GS data set consists of matches

Table 1: Overview of evaluated data sets.

Data set	Number of records	$\|\mathbf{W}\|$	Match:Non-match	Attributes	$n = \|\mathbf{w}\|$
Cora	1,295	286,141	1:16	Title, authors, year, venue	4
Google Scholar	2,616 / 64,263	472,790	1:89	Title, authors, year,venue	6
Music	19,375	251,715	1:16	Title, artist, album, year, language, number	7

between DBLP and GS. The Music data set contains records from the MusicBrainz database[3]. This data set is corrupted [10] and consists of five sources with duplicates for 50% of the original records. To avoid the comparison of the full Cartesian product of vectors, we applied blocking [4] and filtering [14].

The ratios between matches and non-matches (with blocking and filtering applied) shown in Table 1 highlight the imbalance of these data sets and emphasise the challenges of selecting a representative training data set. The similarity vectors (of dimension n) were calculated using string comparison functions on the different attributes shown in Table 1, such as q-gram based Jaccard and Soft-TF/IDF [4]. To classify the similarity vectors as matches and non-matches, we used the decision tree classifier implemented in the Weka toolkit [7].

Our proposed active learning approach is implemented in Java 1.8 and we ran all experiments on a desktop machine equipped with an Intel Core i7-4470 CPU with 8x3.40 GHz CPUs, and 32 GBytes of main memory. To facilitate repeatability, both code and data sets are available from the authors.

We evaluated different parameter settings for our approach. As initialisation method we used *farthest first*, *stratified sampling* and *random selection*, set $\alpha = [0.3, 0.4, 0.5, 0.6, 0.7]$ to weight the *entropy* and *uncertainty* in Eqn. (1) when determining informative similarity vectors, set the number of selected vectors in each iteration as $k = [30, 35, 40, 45, 50]$, and the total budget $b = [200, 500, 1000, 2000, 5000]$. We set default values as $\alpha = 0.5$, $k = 30$, $b = 1000$ and *farthest first* as the initialisation method, because we obtained good results with these settings for all three data sets based on preliminary experiments.

We compared our approach with the two basic active learning approaches *Smallest Margin* [25] and *Entropy* [21], the *Uncertainty* selection approach [16], as well as the only budget limited active learning approach for ER we are aware of (named *Clu-AL*) [26]. We do not compare our approach with *MinExpError* [16] because this approach does not scale well for large budgets. Furthermore, we compared our approach with both fully supervised decision tree and support vector machine (using RBF and linear kernels) classifiers, as also used for comparison in previous work on active learning for ER [26].

To allow a comparative evaluation of our proposed approach with these earlier approaches we use the F-measure [9]. We acknowledge that there are issues when this measure is used to comparatively evaluate different ER classifiers, however there is currently no accepted alternative to the F-measure we are aware of.

[3] Available at: `https://musicbrainz.org`

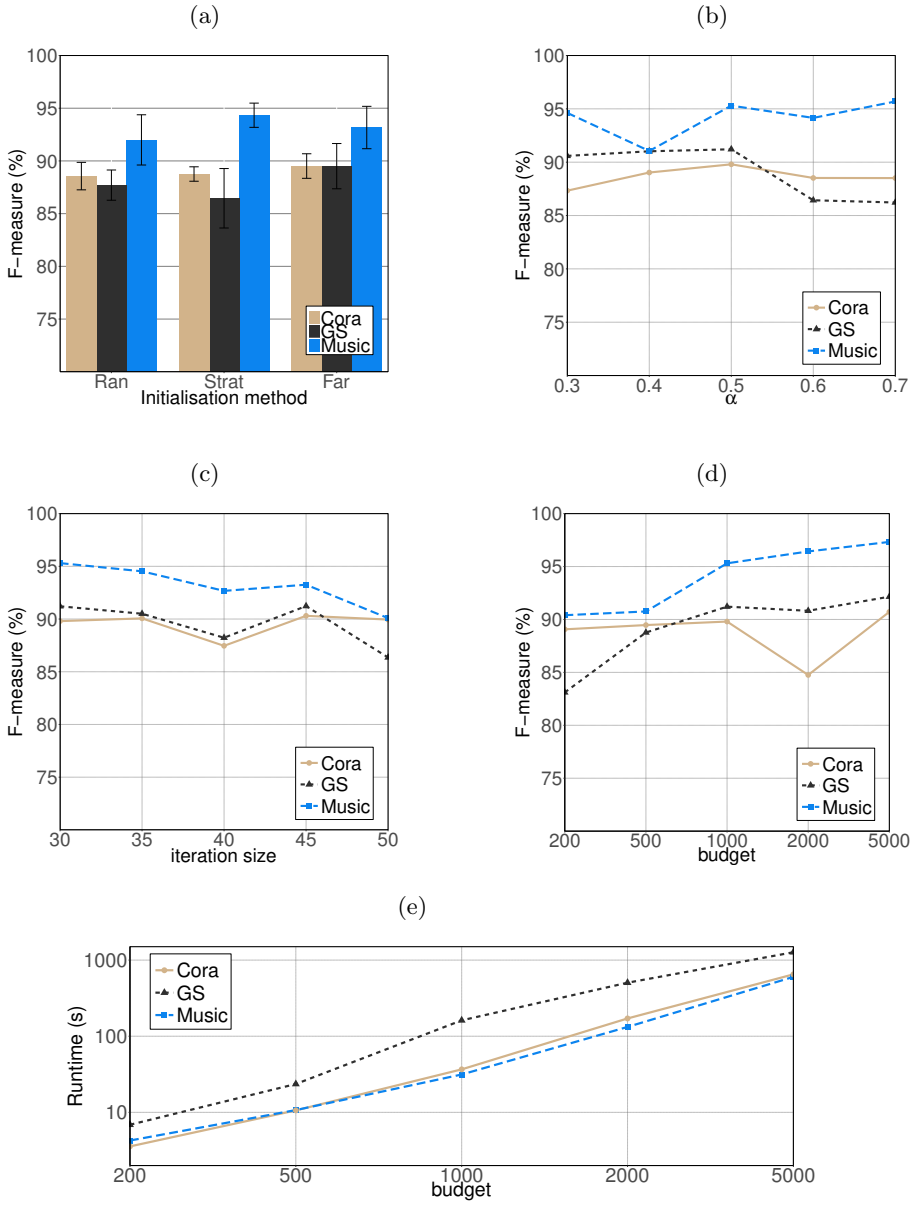


Fig. 5: Classification F-measure results for (a) different initialisation methods, (b) different values for weight parameter α of $info$, (c) different numbers of similarity vectors per iteration k, (d) different total budgets b, and (e) runtime for different total budgets b.

5.1 Parameter Evaluation

Figure 5a shows the obtained ER classification quality for different initialisation methods averaged over different iteration sizes k. Farthest first slightly outperforms stratified sampling and random selection by 0.75% and 0.95%, respectively, for the Cora data set, and by 3.1% and 1.8% for Google Scholar. On the other hand, Farthest first achieves a lower F-Measure by 1.17% compared to stratified sampling for the Music data set. The small differences in F-measure results for the different initial selection strategies show that our main selection strategy based on the search space of informative vectors performs effectively independent of the initial set of similarity vectors.

As can be seen in Fig. 5b, changes for the weight parameter α only slightly influence the ER classification quality, between 2% to 4%, for the three data sets. For the Cora data set we observe a decreasing quality for $\alpha > 0.5$. With an α weight over 0.5 our approach prioritises the *entropy* of a vector more than the *uncertainty*, and therefore the approach mainly selects vectors as informative that are located in-between true matches and non-matches.

For all three data sets, the F-measure slightly decreases with a higher number of selected similarity vectors, k, per iteration as shown in Fig. 5c. This indicates that a higher number of selected similarity vectors increases the probability for selecting non-informative vectors. An increasing budget generally leads to an improvement of F-measure results as shown in Fig. 5d. Even for a small budget of $b = 200$, for all three data sets our approach achieves F-measure results of above 80%, with an increase up to 97% for the Music data set as more informative vectors are added to the training set. The runtime scales quadratically with respect to the total budget as shown in Fig. 5e, however, all runtimes are below 200 seconds for budgets up to $b = 1,000$.

5.2 Comparison with Existing Approaches

We compare our active learning approach, named *InfoSpace-AL*, with the active learning approaches *Smallest Margin*, *Entropy*, and *Uncertainty*, as well as the clustering based active learning approach *Clu-AL* [26]. We also compare our approach with supervised approaches using fully supervised SVM and decision tree classifiers. To compare the different active learning approaches, we experimentally determined a suitable number of similarity vectors to select in each iteration, k, for each approach separately over all data sets. We use the following values for k: *Smallest Margin*: 45, *Entropy*: 50, *Uncertainty*: 45, and *InfoSpace-AL*: 30. The *Clu-AL* approach follows an adaptive strategy for determining the number of similarity vectors it selects in each iteration.

Figure 6 shows the F-Measure of the considered approaches according to different budgets b. *InfoSpace-AL* is the only approach that, for a small budget, achieves an F-Measure above 80% for all three data sets. *Smallest Margin* and *Uncertainty* result in a high variance with an increasing budget, where the F-Measure achieved by *Uncertainty* is reduced by up to 8.7% from a budget of $b = 200$ to $b = 500$. In contrast, *InfoSpace-AL* achieves more stable F-Measure results

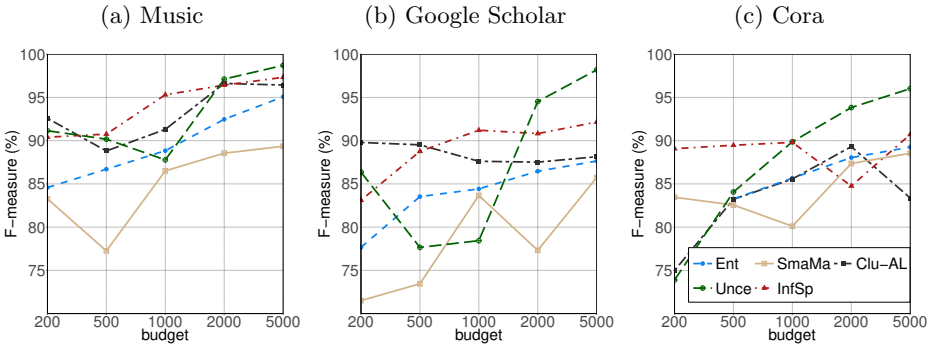


Fig. 6: F-measure results of our approach (named *InfoSpace-AL*, InfSp) as compared with the other active learning approaches *Entropy* (Entr) [21], *Smallest Margin* (SmaMa) [25], *Clu-AL* [26] and *Uncertainty* (Unce) [16].

Table 2: F-measure results of our approach (InfoSpace-AL) as compared with fully supervised classifiers (SVM and DTree) for a budget of $b = 1,000$.

Data set	Dtree	SVM	**InfoSpace-AL**
Google Scholar	88.63%	91.44%	91.21%
Cora	84.09%	82.22%	89.80%
Music	96.80%	96.90%	95.30%

compared to *Uncertainty* even for small budgets of $200 \leq b \leq 1,000$. *InfoSpace-AL* and *Clu-AL* both achieve high F-Measure results for each data set for small budgets of $b = 500$ and $b = 1,000$. However, we observe that *Uncertainty* achieves high F-Measure values above 90% for each data set if the budget is above $b = 2,000$. To summarise, our approach achieves results comparable to *Clu-AL* and *Uncertainty*, and it is one of the best performing approaches for small budgets of up-to $b = 1,000$.

To evaluate the two supervised approaches, we applied 10-fold cross validation. Our approach achieves comparable results compared to the fully supervised approaches as shown in Table 2. Our informativeness-based active learning approach outperforms the supervised approaches by around 5.7% in F-Measure for the Cora data set. On the other hand, the supervised approaches achieve higher F-Measure results for the Google Scholar and Music data sets compared to our active learning approach. However, we emphasise that our approach achieves these comparable results with a moderate manual classification effort, so that the labelling effort is reduced by around 99% compared to a fully supervised classifier that requires much larger training data sets which are commonly not available in real-world ER applications.

6 Conclusions and Future Work

We have proposed an active learning approach for entity resolution (ER) that iteratively selects similarity vectors into a training data set based on the informativeness of vectors for a current training data set. Unlike with existing active learning approaches for ER, the main advantage of our approach is that it is independent of any intermediate classification results since it determines the search space for new vectors based on a defined informativeness measure considering the location of vectors in the vector space, as well as the uncertainty of the search space. In each iteration, our approach selects new vectors according to the most informative vectors. The evaluation showed that our approach can achieve results comparable to fully supervised approaches where much larger training data sets are required to achieve a high ER quality compared to our budget limited approach. Moreover, our approach outperforms a previous state-of-art active learning method for ER that is also based on a limited budget for the number of manual classifications possible. Furthermore, our approach does also not rely on the assumption of monotonicity of precision [26].

For future work we aim to investigate adaptive methods for determining an optimal number k of selected similarity vectors in each iteration such that the probability for selecting non-informative similarity vectors is minimised. We also plan to investigate filtering methods that initially reduce the set of vectors $\mathbf{W}$ to avoid the selection of non-informative vectors. Moreover, we like to integrate metric space approaches to improve the efficiency of the approach for determining new unlabelled similarity vectors.

Acknowledgements

This work was partially funded by the Australian Research Council (ARC) under Discovery Project DP160101934, and Universities Australia and the German Academic Exchange Service (DAAD).

References

1. Arasu, A., Götz, M., Kaushik, R.: On active learning of record matching packages. In: ACM SIGMOD. pp. 783–794. Indianapolis (2010)
2. Bellare, K., Iyengar, S., Parameswaran, A.G., Rastogi, V.: Active sampling for entity matching. In: ACM SIGKDD. pp. 1131–1139. Beijing (2012)
3. Christen, P.: Automatic record linkage using seeded nearest neighbour and support vector machine classification. In: ACM SIGKDD. pp. 151–159. Las Vegas (2008)
4. Christen, P.: Data Matching – Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection. Springer (2012)
5. Dasgupta, S.: Two faces of active learning. Theoretical Computer Science 412(19), 1767 – 1781 (2011)
6. Ertekin, S., Huang, J., Bottou, L., Giles, L.: Learning on the border: Active learning in imbalanced data classification. In: ACM CIKM. pp. 127–136. Lisbon (2007)

7. Frank, E., Hall, M., Holmes, G., Kirkby, R., Pfahringer, B., Witten, I.H., Trigg, L.: Weka-a machine learning workbench for data mining. In: Data mining and knowledge discovery handbook, pp. 1269–1277. Springer (2009)
8. Gokhale, C., Das, S., Doan, A., Naughton, J.F., Rampalli, N., Shavlik, J., Zhu, X.: Corleone: Hands-off crowdsourcing for entity matching. In: ACM SIGMOD. pp. 601–612. Snowbird, Utah (2014)
9. Hand, D.J., Christen, P.: A note on using the F-measure for evaluating record linkage algorithms. Statistics and Computing 28(3), 539–547 (2017)
10. Hildebrandt, K., Panse, F., Wilcke, N., Ritter, N.: Large-scale data pollution with apache spark. IEEE Transactions on Big Data (2017)
11. Köpcke, H., Rahm, E.: Frameworks for entity matching: A comparison. Data and Knowledge Engineering 69(2), 197–210 (2010)
12. Köpcke, H., Thor, A., Rahm, E.: Evaluation of entity resolution approaches on real-world match problems. PVLDB Endowment 3(1-2), 484–493 (2010)
13. Köpcke, H., Rahm, E.: Training selection for tuning entity matching. In: QDB/MUD. pp. 3–12. Auckland (2008)
14. Köpcke, H., Thor, A., Rahm, E.: Learning-based approaches for matching web data entities. IEEE Internet Computing 14(4), 23–31 (2010)
15. McCallum, A., Nigam, K., Ungar, L.H.: Efficient clustering of high-dimensional data sets with application to reference matching. In: ACM SIGKDD. pp. 169–178. Boston (2000)
16. Mozafari, B., Sarkar, P., Franklin, M., Jordan, M., Madden, S.: Scaling up crowd-sourcing to very large datasets: A case for active learning. PVLDB Endowment 8(2), 125–136 (Oct 2014)
17. Naumann, F., Herschel, M.: An introduction to duplicate detection. Synthesis Lectures on Data Management, Morgan and Claypool Publishers (2010)
18. Nentwig, M., Hartung, M., Ngonga Ngomo, A.C., Rahm, E.: A survey of current link discovery frameworks. Semantic Web 8, 419–436 (2017)
19. Ngonga Ngomo, A.C., Lyko, K.: Eagle: Efficient active learning of link specifications using genetic programming. In: The Semantic Web: Research and Applications. pp. 149–163. Berlin, Heidelberg (2012)
20. Rahm, E., Do, H.H.: Data cleaning: Problems and current approaches. IEEE Data Engineering Bulletin 23(4), 3–13 (2000)
21. Settles, B.: Active learning literature survey. Tech. rep., University of Wisconsin-Madison Department of Computer Sciences (2009)
22. Shannon, C.E.: A mathematical theory of communication. Bell system technical journal 27 (1948)
23. Sherif, M.A., Ngonga Ngomo, A.C., Lehmann, J.: Wombat – a generalization approach for automatic link discovery. In: The Semantic Web. Cham (2017)
24. Singh, R., Meduri, V.V., Elmagarmid, A.K., Madden, S., Papotti, P., Quiané-Ruiz, J., Solar-Lezama, A., Tang, N.: Synthesizing entity matching rules by examples. PVLDB 11(2), 189–202 (2017)
25. Tsai, M.H., Ho, C.H., Lin, C.J.: Active learning strategies using SVMs. In: The 2010 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE, Barcelona (2010)
26. Wang, Q., Vatsalan, D., Christen, P.: Efficient interactive training selection for large-scale entity resolution. In: PAKDD. Ho Chi Minh City, Vietnam (2015)
27. Wang, S., Xiao, X., Lee, C.H.: Crowd-based deduplication: An adaptive approach. In: ACM SIGMOD. pp. 1263–1277. Melbourne (2015)