A CLASSIFICATION-BASED APPROACH FOR BIBLIOGRAPHIC METADATA DEDUPLICATION

Eduardo N. Borges[1], Karin Becker[2], Carlos A. Heuser[2] and Renata Galante[2]
[1]*Computational Science Center, Federal University of Rio Grande, Rio Grande, Brazil*
[2]*Institute of Informatics, Federal University of Rio Grande do Sul, Porto Alegre, Brazil*

ABSTRACT

Digital libraries of scientific articles describe them using a set of metadata, including bibliographic references. These references can be represented by several formats and styles. Considerable content variations can occur in some metadata fields such as title, author names and publication venue. Besides, it is quite common to find references that omit same metadata fields such as page numbers. Duplicate entries influence the quality of digital library services once they need to be appropriately identified and treated. This paper presents a comparative analysis among different data classification algorithms used to identify duplicated bibliographic metadata records. We have investigated the discovered patterns by comparing the rules and the decision tree with the heuristics adopted in a previous work. Our experiments show that the combination of specific-purpose similarity functions previously proposed and classification algorithms represent an improvement up to 12% when compared to the experiments using our original approach.

KEYWORDS

Deduplication, bibliographic metadata, classification, machine learning.

1. INTRODUCTION

Digital libraries are complex information systems composed of multiple collections of digital objects that provide several services to the community of users such as search, navigation, retrieval and recommendation. In digital libraries of scientific articles, such as ACM DL and IEEE Xplore, bibliographic references are the most important metadata. References are represented by several formats, including different styles. A style is a set of rules for formatting references and citations (e.g. IEEE, Harvard). In addition to difference in styles, metadata content representation (title, authors' names, publication venue, among others) can vary considerably. References can contain in addition variations in spelling, omission of words, and even spelling errors (Carvalho et al., 2008). Therefore, several heterogeneous references can represent the same article, as observed in the examples of Table 1.

Table 1. Heterogeneity of bibliographical references

Elmagarmid, A. et al., 2007. Duplicate record detection: a survey. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 19, No. 1, pp. 1-16.
Ahmed K. Elmagarmid, Vassilios S. Verykios. Duplicate Record Detection: A Survey, IEEE TKDE, 2007.
Elmagarmid, A.K.; Ipeirotis, P.G.; Verykios, V.S. Duplicate record detection. *Knowledge and Data Engineering*, IEEE Trans., 19(1), 2007.

The task of finding matching records in databases is called deduplication, record linkage or instance matching (Doan et al., 2004). Several approaches to record deduplication have been proposed in recent years. Most of these works focus on the deduplication task in the context of relational databases integration (Dorneles et al., 2009; Carvalho et al., 2008; Bilenko and Mooney, 2003; Cohen and Richman, 2002). Few automatic approaches have been specifically developed for digital libraries (Borges et al., 2001b; Lawrence et al., 1999). In the digital libraries domain, deduplication is generally based on the semantics of specific metadata fields. For example, fields that specify the authors or the title of a digital object are among the most discriminative fields of a record, and hence this information can be used as a strong evidence of similarity for the deduplication process. There may be several references with similar titles, but if the authors do not have

similar names, most probably they are different real-world objects (Borges et al., 2011). For instance, Baeza-Yates and Ribeiro-Neto (1999) as well as Manning et al. (2008) have published books with similar titles: *Modern Information Retrieval* and *Introduction to Information Retrieval*. Another specific problem of digital libraries is the variation in the representation of author names in bibliographic references and citations. Variations include abbreviations, inversions of names and omission of suffixes such as Jr (Ley, 2002).

This paper discusses an approach that combines similarity functions and classification algorithms for identifying duplicated bibliographic metadata, from which preliminary results were presented in (Borges et al., 2011a). The similarity functions specially designed for the metadata content were presented in (Borges et al., 2011b), together with a composition function to identify reference replicas. One of the drawbacks of this approach is that it requires the definition of similarity thresholds. To overcome this problem, we used the similarity scores to train classification algorithms to automatically identify duplicated references. When compared to the experiments using our original deduplication approach, the experiments developed in this paper revealed that the combination of specific-purpose similarity functions and classification algorithms represent an improvement in precision, recall and F-measures. The present paper builds up on previous work by refining the experiments, analyzing the characteristics of the classification models, and comparing them with the deduplication heuristics proposed in (Borges et al., 2011b).

The rest of this paper is organized as follows. In Section 2, we discuss related work. In Section 3, we present our classification-based approach to deduplicate bibliographic metadata. We give details on the performed experiments and discuss the obtained results in Section 4. We also present a comparative analysis of algorithms. Finally, in Section 5, we draw our conclusions and point out some future work directions.

2. RELATED WORK

The problem of bibliographic citation deduplication is explicitly discussed by Lawrence et al. (1999). The authors propose algorithms for matching references from different sources based on metrics like edit-distance, word matching, phrase matching and subfield extraction. Usually, deduplication algorithms combine the values of these metrics (or any other similarity functions) by generating a similarity score between the records. If this score exceeds a similarity threshold, these records are considered sufficiently similar to represent the same real-world object. Score values depend on the metadata content, the similarity functions and the matching algorithms. So the choice of effective similarity thresholds is not a trivial task. Dorneles et al. (2009) define a strategy to compare similarity scores. These scores are redefined according to the expected precision of record matching. This approach maps similarity scores into precision values using a training set. The choice of the expected precision is an easier task for the expert, but the identification of replicas still requires human intervention.

Other work have proposed strategies based on machine learning techniques, mostly supervised ones. These strategies estimate similarities and match duplicate records, without thresholds definition. Cohen and Richman (2002) propose a scalable and adaptive technique to group objects based on the string similarity of different records. The MARLIN system (Bilenko and Mooney, 2003) explores a framework for identifying replicas using adaptive string similarity metrics applied to each field, according to the domain of their values. The system defines two similarity metrics: one based on edit distance and another based on support vector machines (SVM). Carvalho et al. (2008) propose an approach based on genetic programming to find suitable similarity functions based on the combination of multiple pieces of evidence.

Most of the papers described in this section contribute with solutions for the general record deduplication problem, which do not take into account the specifics of the digital library domain. In our previous work (Borges et al., 2011b) we propose an approach for metadata record deduplication that is based on a set of similarity functions specially designed for the digital library domain. We have defined these functions based on the heuristics:

1. Duplicate references have very close publication years. Same year or one year of difference is a good indication since it covers, for example, the cases of events that sometimes have their formal proceedings published in the following year.

2. Duplicate references share most of the authors. Due to errors and problems in data acquisition, it is common to find references which authors' names list is not complete. Besides, we need to support variations of spelling, omission of middle names, abbreviations and inversions in names ordering.

3. Duplicate references have very similar titles, but if the authors do not have similar names, most probably they are different real-world objects.

Our unsupervised heuristic-based strategy greatly reduces the number of comparisons that use string matching algorithms using an efficient two-phase blocking method, but it is sensitive to the definition of similarity thresholds, such as minimum year difference, minimum percentage of authors' matches and minimum title distance. We starts by checking whether the publication years of a pair of digital objects are within the defined similarity range. Only objects whose absolute value of the difference between their publication years is less than or equal to a provided threshold will have their author names compared. This strategy is used to significantly reduce the number of further (more expensive) string comparisons. Then, the initials of the author names are extracted and compared. Only objects that reach the author name matching threshold will have their titles compared by the edit-distance function.

In Borges et al. (2011a), we propose to use classification algorithms to avoid the burden of similarity threshold definition. Now, in this paper, we have analyzed the discovered patterns by comparing the rules and the decision tree with the heuristics adopted in the original approach (Borges et al., 2011b).

3. CLASSIFICATION-BASED APPROACH FOR DEDUPLICATION

This section summarizes the classification-based approach for identification of deduplicate bibliographic metadata proposed in (Borges et al. 2011b). We define as replicas two or more references that are semantically equivalent, i.e. references that describe the same publication item indexed by a digital library. The metadata content is compared using similarity functions, which are chosen according to each metadata field. The score values returned by similarity functions are used to train a classification model which identifies duplicates automatically, without requiring human intervention to set up similarity thresholds. Our approach to deduplicate bibliographic metadata records is split into distinct phases according to a knowledge discovery process in databases (Fayyad, 1996). The following sections summarize the adopted process.

3.1 Data Selection and Preprocessing

Metadata standards like Dublin Core and MARC 21 are represented by a flat structure composed by several metadata fields. Our deduplication approach selects only metadata fields shared by different bibliographic references like books, articles, papers and Web pages. We propose to adopt only title, authors' names and publication year because these are the common attributes found in majority of references. In addition, they are less susceptible to noise when compared to metadata fields such as publication venue, page numbers, among others. For instance, "TKDE" and "IEEE Transactions on Knowledge and Data Engineering" describes the same Journal but do not share any substring and it is quite common to find references that omit page numbers. Besides title, authors' names and publication year, we assume there is a class field that means which real-word article the metadata record refers.

Next, we apply preprocessing operations in order to clean and normalize the selected metadata content. The first step is to clean all selected metadata fields by applying usual string transformations. Then the publication year is transformed into a valid integer in the domain. We use the four leftmost digits to extract the year from dates or timestamps in ISO standard as "2011-09-20 08:32:45". After that, besides the typical transformations in authors' names, we define the delimiter characters to be used by similarity functions that compare this metadata field. Finally, we remove noisy instances, i.e., the records that the number of authors is zero or do not have a valid publication year. Borges et al. (2011b) shows details about preprocessing.

3.2 Data Transformation

After preprocessing, the references are combined in pairs (ref_i, ref_j) generating new records combining the metadata fields of any two different references. The similarity functions proposed in Borges et al. (2011b) are applied on each new record, i.e. to compare pair of distinct references. The similarity scores are added as new fields of each record.

Table 2 shows the new fields and corresponding functions. These similarity functions return integers or real values varying in the range [0,1]. There are performed differences or similarities between the pairs of

publication years, authors' names and titles. If the pair of original classes are equal, a new binary class labeled "duplicated pair" is defined with value *yes*. Otherwise, the value *no* is assigned to duplicated pair. Then original string fields $authors_i$, $title_i$ and $class_i$, where i is the reference identifier, are removed. Only numerical fields with different distributions remain.

Table 2. Distance and similarity functions applied to each pair of references

	New metadata field	Distance or similarity function
1	authors number$_i$	numbers of authors from reference i
2	authors number$_j$	numbers of authors from reference j
3	authors number diff	absolute difference between the numbers of authors ($\|authors\ number_i - authors\ number_j\|$)
4	year diff	absolute difference between the publication years ($\|year_i - year_j\|$)
5	authors diff	difference between the authors according to the algorithm *NameMatch* (Borges et al., 2011b)
6	authors sim	similarity between the authors based on the normalized *authors diff*
7	title diff	edit-distance between $title_i$ and $title_j$
8	title sim	similarity between titles based on the normalized difference *title diff*
9	duplicated pair	binary class (*yes* if $class_i = class_j$ or *no* otherwise)

3.3 Data Mining and Interpretation of Results

The goal of the mining phase is to train a classification model for predicting whether two distinct bibliographic references refer to the same real-world article. Hence the data transformed as described in Section 3.2 are used as input of a classification algorithm. We have experimented with distinct types of classification techniques (Borges et al., 2011a) in order to understand the properties of the data. The best results so far have been yielded by the following classification algorithms (Tan et al., 2005):

- Naïve Bayes – based on the Bayes theorem;
- RIPPER – rule-based;
- C4.5 – based on decision trees.

Results are interpreted and evaluated with regard to the quality of the deduplication process. The classification results can be evaluated using the following metrics: precision, recall, f-measure (Manning et al., 2008) and Wilcoxon signed-rank test (Wilcoxon 1945).

4. EXPERIMENTAL EVALUATION

This section describes the experiments developed in order to test the use of classification algorithms for automatically deduplicate bibliographic metadata. We used a real database consisting of scientific articles references. The classification algorithms are evaluated by the quality of deduplication process. We have performed the experiments in a standard PC using Weka[1] data mining tool.

4.1 Dataset

The dataset used in our experiments was extracted from the Cora Collection. Cora is a collection of references extracted from a search engine for research papers in Computer Science (McCallum et al., 2000). References were segmented into multiple fields by an information extraction system, resulting in some crossover noise among the fields. For instance, some publication dates are captured in some fields other than year. There are 2191 records distributed in 305 distinct classes in the raw data. This collection has been used for experimental evaluation of related work (Borges et al., 2011a, 2011b; Carvalho et al., 2008; Bilenko and Mooney, 2003). Table 3 presents the structure of Cora metadata records.

The records were processed according to the procedure detailed in Section 3. We have also removed the metadata fields $authors\ number_i$, $authors\ number_j$ and records with evident outliers (e.g. references without publication year). The remaining instances were combined in pairs generating approximately 1.9 million of new records. The distance and similarity functions (Table 3) were applied on each new record. Finally, we have randomly selected 10% of instances 5 times in order to compose 5 distinct samples. We kept the same

[1] http://www.cs.waikato.ac.nz/ml/weka

proportion between classes. Each sample has 2,540 (1.3%) instances labeled as replicas (duplicated pair = *yes*) and 191,013 (98.7%) as distinct real-world objects (duplicated pair = *no*). The number of instances was reduced because the classification algorithms run entirely in memory and they have not linear complexity.

Table 3. The structure of Cora metadata records (raw data)

	Metadata field	Description
1	id	unique identification of a record
2	title	article title
3	authors	authors' names
4	year	publication year
5	venue	publication venue
6	other	other information contained in the reference such as page numbers, volume and issue
7	all	full reference (without field segmentation)
8	class	which real-word article this record refers

4.2 Deduplication Results

In this section we examine the experiments results. First, we performed a comparative analysis of classification algorithms applied to deduplication. Then, our classification-based approach is compared with our unsupervised heuristic-based approach (Borges et al., 2011b).

4.2.1 Evaluation of Classification Models

We have run the classification algorithms 5 times in 5 samples. Experiments results are summarized in Table 4. This table presents the specific parameters of each algorithm that achieved the best results and four quality measures: precision, recall, balanced f-measure (F1) and f-measure with double weight for recall (F2). The parameters were set in order to avoid overfitting to the data sets. Quality results are mean values with the standard deviation considering all runs and they refer only the class of interest (duplicated pair = yes).

Table 4. Classification results for replicas (10 fold cross-validation)

	Algorithm	Parameters	Precision (%)	Recall (%)	F1 (%)	F2 (%)
1	Naïve Bayes	Default	69.1 ± 0.7	**98.4 ± 0.2**	81.2 ± 0.5	90.7 ± 0.3
2	RIPPER	one optimization phase	**86.1 ± 1.2**	95.0 ± 1.7	**90.3 ± 0.5**	**93.0 ± 1.1**
3	C4.5	15% confidence for pruning at least 10 instances per l	**86.9 ± 1.5**	92.7 ± 2.9	**89.7 ± 0.6**	**91.4 ± 1,9**

By observing Table 4, we notice that all experiments achieved recall values higher than 90%, i.e. all tested algorithms were very effective for identifying replicas. When analyzing only the recall, the best result was yielded by the Naïve Bayes algorithm (line 1). However, this same classifier achieved precision values lower than 70%. Low precision denotes that many false positives were returned, which decreases the deduplication quality. The best values for precision were presented by the algorithms C4.5 and RIPPER.

The overall quality of deduplication can be assessed by the f-measure. RIPPER and C4.5 showed the best F1 results, yielding values around 90% (lines 2-3) whereas Naïve Bayes obtained f-measure and standard deviation values equal to 81.2 ± 0.5% (line 1). When F2 is used in the evaluation, Naïve Bayes got better results (90.7 ± 0.3%) because the high recall score, however RIPPER and C4.5 achieved the best F2 values.

Running more times, we have confirmed the behavior of the classification algorithms already experimented in Borges et al. (2011a). To better understand how classifiers use the distance and similarity values to label a pair of references as duplicated or not, we analyze the excerpts of models that are similar in nature that resulting from your various runs.

Table 5. Rule induction of RIPPER (random sample with 10% of instances)

	Predicate	Class	Labeled	Errors
1	title sim $\geq$ 0.62 $\wedge$ year diff $\leq$ 0	yes	2485	258
2	authors sim $\geq$ 0.67 $\wedge$ title sim $\geq$ 0.48 $\wedge$ year diff $\leq$ 1 $\wedge$ title diff $\geq$ 36	yes	13	3
3	authors sim $\geq$ 0.67 $\wedge$ title sim $\geq$ 0.72 $\wedge$ title diff = 4	yes	11	2
4	authors sim $\geq$ 0.67 $\wedge$ title sim $\geq$ 0.44 $\wedge$ year diff $\leq$ 1 $\wedge$ title diff $\leq$ 25	yes	9	1
5	authors sim $\geq$ 0.67 $\wedge$ title sim $\geq$ 0.74 $\wedge$ year diff $\leq$ 2 $\wedge$ title diff $\geq$ 16	yes	5	1
6	authors sim $\geq$ 0.67 $\wedge$ title sim $\geq$ 0.47 $\wedge$ year diff $\leq$ 1 $\wedge$ title diff $\geq$ 34	yes	9	2
7	$\forall$ instance	no	191021	275

Table 5 presents the set of rules inducted by RIPPER for the first sample. Each line shows a logical predicate including the metadata fields, the predicted class (duplicated pair = *yes* or *no*), the number of instances classified by the rule and the number of classification errors. The first rule classified most of the replicas using only the metadata fields *title sim* and *year diff*. This behavior shows that these attributes are the most discriminatory for the identification of duplicate references. The threshold 62% for the edit-distance between titles is enough to correctly classify 88% of the replicas. In addition to the attributes mentioned above, the following rules (2-6) use the similarity between authors. The threshold 67% means that 2/3 of the authors in a pair of references should match. These rules also deal with some specific issues on publication years and the titles length. The last rule classifies the vast majority of distinct pairs of references.

Figure 1 presents the decision tree generated by C4.5 for the same sample. Class values yes/no are presented as t/f respectively. The algorithm uses the gain ratio to adjust the information gain by partitioning entropy. title diff was the most discriminatory attribute (root) followed by year diff and title sim. C4.5 was able to correctly identify 85% of the replicas using only the edit-distance between titles and the difference between publication years. We notice that metadata fields corresponding to the results of difference functions (title diff) were considered more important than derived similarities fields (title sim). Besides, the only information about authors used was the difference between the numbers of authors, i.e., the classification model described by this decision tree does not consider who published the scientific articles, just how many authors they have. For some samples, the difference and/or similarity between authors was used, but only near the leaf nodes, classifying few instances.

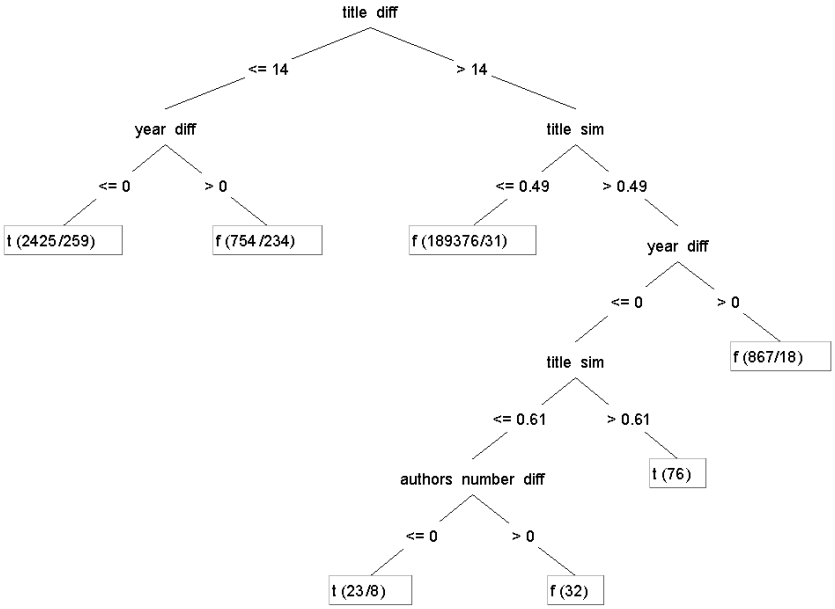

Figure 2. C4.5 decision tree (random sample with 10% of instances)

Table 6 summarizes the data features from the same sample as yielded by the Naïve Bayes algorithm. For each metadata field, it presents the mean values and the standard deviation. Naïve Bayes uses this information to estimate the probability a posteriori of two references to be considered replicas. The worst precision (69.1 ± 0.7%) achieved by Naïve Bayes is most probably due to the existence of discriminatory classification attributes such as title similarity and year difference fields. Naïve Bayes is a classification algorithm more suitable when such attributes are not present in the dataset.

To summarize, we can see that in general the metadata fields corresponding to the results of difference functions were considered more important than derived similarities fields based on these differences. *title diff* or *title sim* were the most discriminatory attributes, together with *year diff*, used as a complement to classify most replica instances. The distance and similarity between authors' names, when used, classified a very small fraction of the instances.

Table 6. Summary of data features used by Naïve Bayes

	Metadata filed	Class = no	Class = yes
1	authors number diff	1.88 ± 3.60	0.07 ± 0.39
2	year diff	4.02 ± 3.40	0.15 ± 0.70
3	authors diff	3.29 ± 3.63	0.15 ± 0.60
4	authors sim	0.09 ± 0.20	0.96 ± 0.13
5	title diff	49.7 ± 17.8	3.62 ± 9.23
6	title sim	0.24 ± 0.08	0.94 ± 0.12

4.2.2 Comparison with Baseline

Table 7 presents the best deduplication results considering our classification approach reported in this paper and our unsupervised heuristic-based approach previously proposed (Borges et al., 2011b). It shows the same metrics used so far considering the class of interest (duplicated pair = *yes*): precision, recall, F1 and F2. The results obtained by using classifiers outperformed the heuristic-based approach considering all four quality metrics. They represent a mean gain of 3.6% in precision, 14.2% in recall, 8.9% in F1 and 12% in F2. In summary, the use of classification algorithms have improved the quality of deduplication up to 12% yielding F2 and standard deviation values up to 91.4 ± 1,9%.

Table 7. Deduplication results for both approaches

Approach	Precision (%)	Recall (%)	F1 (%)	F2 (%)
Unsupervised heuristic-based (baseline)	83.9 ± 0.2	81.2 ± 0.3	82.4 ± 0.2	81.6 ± 0.2
Classification-based (C4.5)	**86.9 ± 1.5**	**92.7 ± 2.9**	**89.7 ± 0.6**	**91.4 ± 1,9**
Average improvement	3.6	14.2	8.9	12

To check whether our improvements are in fact statistically significant, we performed a paired Wilcoxon test (Wilcoxon, 1945) comparing the proposed approach using classification with the original unsupervised heuristic-based approach. We have used the Wilcoxon test because the samples are not normally distributed. The values of 1-tailed p considering 1000 observations were lower than 0.001 for recall, F1 and F2. The value of 1-tailed p for precision was 0.33, because the standard deviation of 1.5% is a high value when compared to the average improvement of 3.6%. Therefore, our approach has a performance that is statistically superior to the baseline for all metrics except precision since p values should be lower than the statistical significance threshold $\alpha = 0.01$.

When comparing the classification models with our heuristics, we can observe that *year diff* is indeed a very important attribute, but the most discriminatory one is indeed the title. In our approach, titles are only compared if similarity of years and authors indicate evidence that the digital objects may be replicas. This is justified by the fact that the detection approach is done by searching the whole database, instead of generating a classification model on training data that can be used to forecast. In that sense, our heuristics are similar to lazy classification algorithms that do not require training.

5. CONCLUSION

This paper presents an extension previous works to identify duplicated bibliographic metadata. Instead of setting similarity thresholds, we use the scores returned by the similarity functions specially designed for the metadata content to train classification algorithms that identify duplicated references. The main benefits of using classification algorithms are the increase in the quality of the deduplication process and the automatic identification of duplicates, without requiring human intervention for similarity threshold definition.

The results of our experiments show that the classification algorithms, combined with the similarity functions, identify up to 90% of duplicated citations with quality up to up to 91.4 ± 1,9% measured according to F2. Except for precision, all quality metrics presented improves that are statistically significant with regard to baseline.

The advantages of the classification approach are the better quality, considering the experiments. However, the cost is concentrated in the preprocessing of instances and training the data. The unsupervised approach proposed in (Borges et al., 2011b) can be considered a lazy classification, as it does not required training, which can be useful for digital libraries that suffer constant updates and the training model may become obsolete. In that case, the heuristics that are focused on reducing the cost of string matching can be

valuable. Indeed, for very large databases, the cost to calculate the title distance and similarity can be very high or even prohibitive. Recent solutions in the literature (Christen, 2011; Baxter et al., 2003), including multiple blocking methods, could be used to optimize the preprocessing phase by applying the similarity functions only over the best candidates for matching. However, some duplicates could not be detected because they could not have been marked as candidates. Besides an efficiently blocking strategy, the evaluation of similarities can be parallelized using a programming model as MapReduce (dal Bianco et al., 2011; Dean and Ghemawat, 2008).

Future work will include new experiments with multiple blocking strategies, parallelized evaluation of similarities and other datasets. The use of synthetic data allows varying parameters such as number of replicas and the distance between the original and the replicated values present in the repository of bibliographic references.

REFERENCES

Baeza-Yates, R. and Ribeiro-Neto, B., 1999. *Modern Information Retrieval*. ACM Press / Addison-Wesley.

Baxter, R. et al, 2003. A comparison of fast blocking methods for record linkage. *Proceedings of ACM SIGKDD Workshop on Data Cleaning, Record Linkage and Object Consolidation*. pp. 25-27.

Bilenko, M. and Mooney, R., 2003. Adaptive duplicate detection using learnable string similarity measures. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. pp. 39-48.

Borges, E. et al., 2011. An automatic approach for duplicate bibliographic metadata identification using classification. *International Conference of the Chilean Computer Science Society* (accepted for publication).

Borges, E. et al., 2011. An unsupervised heuristic-based approach for bibliographic metadata deduplication. *Information Processing and Management*, Vol. 47, No. 5, pp. 706-718.

Carvalho, M. et al., 2008. Replica identification using genetic programming. *Proceedings of the ACM Symposium on Applied Computing*. pp. 1801-1806.

Christen, P., 2011. A survey of indexing techniques for scalable record linkage and deduplication. *IEEE Transactions on Knowledge and Data Engineering*, PrePrints, doi: 10.1109/TKDE.2011.127.

Cohen, W. and Richman, J., 2002. Learning to match and cluster large high-dimensional data sets for data integration. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. pp. 475-480.

dal Bianco, G. et al., 2011. A fast approach for parallel deduplication on multicore processors. *Proceedings of the ACM Symposium on Applied Computing*. pp. 1027-1032.

Dean, J. and Ghemawat, S., 2008. Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, Vol. 51, No. 1, pp. 107-113.

Doan, A. et al., 2004. Introduction to the special issue on semantic integration. *SIGMOD Record*, Vol. 33, No. 4, pp. 11-13.

Dorneles, C. et al., 2009. A strategy for allowing meaningful and comparable scores in approximate matching. *Information Systems*, Vol. 34, No. 8, pp. 673-689.

Fayyad, U. et al., 1996. The KDD process for extracting useful knowledge from volumes of data. *Communications of the ACM*, Vol. 39, No. 11, pp. 27-34.

Lawrence, S. et al., 1999. Digital libraries and autonomous citation indexing. *IEEE Computer*, Vol. 32, No. 6, pp. 67-71.

Ley, M., 2002. The DBLP computer science bibliography: evolution, research issues, perspectives. *String Processing and Information Retrieval*. Lecture Notes in Computer Science, Vol. 2476, Springer, pp. 481-486.

Manning, C. et al., 2008. *Introduction to Information Retrieval*. Cambridge University Press.

McCallum, A. et al., 2000. Automating the construction of internet portals with machine learning. *Information Retrieval*, Vol. 3, No. 2, pp. 127-163.

Tan, P.-N. et al., 2005. *Introduction to Data Mining*. Addison-Wesley.

Wilcoxon, F., 1945. Individual comparisons by ranking methods. *Biometrics*, Vol 1, pp. 80–83.