Financial Entity Record Linkage with Random Forests

Kunho Kim
EE & CS
Pennsylvania State University
University Park, PA 16802, USA
kunho@cse.psu.edu

C. Lee Giles
Information Sciences and Technology
EE & CS
Pennsylvania State University
University Park, PA 16802, USA
giles@ist.psu.edu

ABSTRACT

Record linkage refers to the task of finding same entity across different databases. We propose a machine learning based record linkage algorithm for financial entity databases. Record linkage on financial databases are essential for information integration on certain financial entity, since those databases do not have common unified identifier. Our algorithm works in two steps to determine if a pair of record is same entity or not. First we check with proposed rules if the record pair can be exactly matched after cleaning the entity name and address. Second, inspired by earlier work on author name disambiguation, we train a binary Random Forest classifier to decide the linkage. To reduce and scale the computation, this process is done only for candidate pairs within a proposed heuristic. Initial evaluation for precision, recall and F1 measures on two different linking tasks in the Financial Entity Identification and Information Integration (FEIII) Challenge show promising results.

CCS Concepts

•**Information systems** → **Information retrieval;**

Keywords

Record Linkage; Random Forest

1. INTRODUCTION

Data integration is a frequent problem with multiple databases. Record linkage is the task of integrating (linking) the information of a certain entity between different data sources, assuming there is no unique common identifier. Although there are common attributes among data sources, they often have different formats. For example, one data source may use an abbreviation to represent the street address while another uses the full name. Such data make the record linkage problem hard to solve with only simple heuristics.

Here, we present a machine learning based record linkage algorithm to solve Financial Entity Identification and Information Integration (FEIII) Challenge[1]. The goal of this challenge is to automatically link financial entities among different financial data sources. We follow the two-step algorithm proposed in the baseline system, Duke deduplication engine[2]. The first step is to find matching pairs by exact matching. To improve the precision, we create rules to clean the entity name and address. Then we analyze matches among ambiguous pairs. Instead of using a simple heuristic, we use a binary Random Forest (RF) classifier, which has been used for evaluating matches in the author and inventor name disambiguation [3],[4],[5]. Features are extracted from common attributes from the data sources. We use the first two tasks of FEIII challenges for algorithm evaluation.

DSMM'16, June 26-July 01 2016, San Francisco, CA, USA

ISBN 978-1-4503-4407-4/16/06…$15.00

DOI: http://dx.doi.org/10.1145/2951894.2951908

2. RECORD LINKAGE ALGORITHM

Our algorithm starts with selection of matching candidates from whole database records. For each candidate, we perform exact matching with proposed rules to clean the data. If the record is not matched, we make a final decision with a binary RF classifier. Details for each step are below.

2.1 Selecting Candidates

Instead of comparing all possible pairs between data sources, we select candidate record pairs that can potentially match. This step is essential to making the algorithm scale, since target data sources are often consist of millions of records. From our experiments, we use *first word of entity name+state* as a heuristic to select the candidates. Prefix articles, capitalized or not (e.g. The, A, An, a) in the entity name are ignored.

2.2 Exact Matching with Rules

Financial entity records have various name and address representations among the data sources. To perform an accurate match, we first clean the entity name and address based on the rules shown in table 1 and 2 respectively. All the rules are implemented using regular expressions. As such they can be rapidly checked and updated. Then for each candidate pairs, we check if the entity name and the zip code are exactly matched. If so, the pair is labeled as the same entity.

2.3 Random Forest Classifier

We use a binary RF classifier for the final classification of candidate pairs that are not exactly matched. The RF

[1]https://ir.nist.gov/dsfin/
[2]https://github.com/larsga/Duke

Table 1: Rules to clean the entity name

Rule	Example
Remove dots	U.S. Bank → US Bank
Remove article	The First → First
Abbreviations to full form	Corp. → Corporation
& → and	B&W → B and W
Remove postfix "company"	Trust Company → Trust
Remove postfix "/..."	Bank /TA → Bank

Table 2: Rules to clean the entity address

Rule	Example
Remove dots	P.O. Box → PO Box
Unify direction representation	N → North
& → and	M&T → M and T
Abbreviations to full form	Rd → Road

classifier is an ensemble learning classifier that learns a set of decision trees [1]. The RF classifier has been used for matching record pairs for several record linkage and disambiguation problems [3],[4],[5]. We use common attributes among financial entity databases to generate features - entity name, street address, city, state, and zip code. Features are generated from string distances, including Jaro-Winkler [6], Jaccard [2] and exact string matching. Exact matching is defined with 3 different values: 2 if both records are not empty and match, 1 if any of them is empty, and 0 if records do not match. Table 3 shows a list of all features used. The RF classifier is trained with 100 trees and for each split 2 different features are considered.

The first task was to create record linkages from entities in the Federal Financial Institution Examination Council (FFIEC) to those in the Legal Entity Identifiers (LEI) database. The second was linkage from the FFIEC to the Securities and Exchange Commission (SEC) database. To train the RF, we manually labeled 1,000 records between FFIEC and LEI, FFIEC and SEC. For each entity record, we manually selected a keyword from the record name and consider for labeling only those records that have the keyword. To avoid overfitting we remove unnecessary negative pairs while training the RF. We also train an additional RF which combines all of the two training sets. Measured out-of-bag(OOB) errors of all trained RF had a 0.5% minimum.

3. RESULTS

For evaluation we measured pairwise precision, recall, and F1 scores. The results shown in Table 4 generally have better results for linking FFIEC→LEI than FFIEC→SEC. The LEI database has its own version of cleaned attributes and has less ambiguity among records, which can then be used to train a better RF classifier. Also, while labeling, we found that the SEC database has some ambiguous true matches, e.g. a matched record only has an entity name, while other fields are empty. Those cases are manually difficult to do, making the linking task even harder for FFIEC→SEC. When training with all datasets, using samples from only the target databases gives better results for the first task. However, for the second task there was some improvement on recall, which we believe was because positive LEI samples removed ambiguity in the SEC samples.

Table 3: Features used in the random forest

Category	Features
Name	Jaro-Winkler, Jaccard
Address	Jaro-Winkler, Jaccard
City	Jaro-Winkler, Exact
State	Exact
Zip	Exact

Table 4: Record linkage evaluation

Task	Training Set	Precision	Recall	F1
FFIEC→LEI	LEI	99.16%	95.77%	97.44%
	LEI+SEC	97.71%	94.56%	96.11%
FFIEC→SEC	SEC	87.84%	84.78%	86.28%
	LEI+SEC	86.78%	85.65%	86.21%

4. CONCLUSIONS

We present a record linkage algorithm using Random Forests for financial entity database linkage. After selecting candidate matching pairs to link, we use a two step method. First, we propose rules to clean the data and check if a candidate pair is exactly matched. Second, we use a binary Random Forest classifier to make the final decision. Results on two different record linkage tasks in FEIII challenge showed promising results. For the future work, one could apply the algorithm to bigger databases and improve scalability with better blocking functions and parallelization.

5. ACKNOWLEDGMENTS

We gratefully acknowledge partial support from the National Science Foundation.

6. REFERENCES

[1] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[2] P. Jaccard. Étude comparative de la distribution florale dans une portion des alpes et du jura. *Bulletin de la Société Vaudoise des Sciences Naturelles*, 37:547–579, 1901.

[3] M. Khabsa, P. Treeratpituk, and C. L. Giles. Online person name disambiguation with constraints. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries(JCDL'15)*, pages 37–46, 2015.

[4] K. Kim, M. Khabsa, and C. L. Giles. Inventor name disambiguation for a patent database using a random forest and dbscan. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries(JCDL'16)*, 2016.

[5] P. Treeratpituk and C. L. Giles. Disambiguating authors in academic publications using random forests. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries(JCDL'09)*, pages 39–48, 2009.

[6] W. E. Winkler. String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage. In *Proceedings of the Section on Survey Research Methods*, pages 354–359. American Statistical Association, 1990.