Automated Public Data Refining

Martin LIPTÁK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova 3, 842 16 Bratislava, Slovakia
mliptak@gmail.com

Public institutions have legal obligations to share certain data on the Web. While public registers (e.g. businesses, organizations) and bulletins (public procurements) are essential for business communication, other data increase transparency of public institutions and enable public investigation (public contracts). Despite the fact that these data are becoming publicly available on the Web, there are two problems.

The first problem is format and structure that might not be suitable for machine processing. For example some documents are published as scanned images with censored names and prices. This makes such documents difficult to investigate by a human expert and almost impossible to process with computer. For example company liquidations are published in periodic PDF bulletins as unstructured text content and it is difficult to reliably find out if a company is being liquidated or the liquidation is being cancelled. Fortunately the most common format is HTML, which is easy to parse and in most cases provides structure. However, even in correctly parsed and structured data, there are various mistypings, disambiguities and duplicates. These inconsistencies are the second problem and we address them in this paper.

Mistypings, duplicates and disambiguities are a major problem not only in the public data domain. In fact every database possibly merged of multiple sources needs to be cleaned so that queries provide reliable results. This process is widely known as data integration, duplicate detection or record linkage [1]. M. Bilenko and R. Mooney in [2] propose to employ learnable string distance functions for duplicate detection task. M. Hernández and S. Stolfo in [3] have developed a method for removing duplicates from databases of 100 milion to 1 bilion records in a matter of days.

We propose a duplicate detection method based on machine learning algorithms. We use a logistic regression classifier to predict whether samples are duplicates or not. The classifier trains weights of features, provided by user for particular database (like Levenshtein distance of compared fields or presence of particular combination of substrings in compared fields). The user also provides a labelled set of samples that is used to train the classifier. Trained classifier can detect duplicates by predicting using learned feature weights.

* Supervisor: Ing. Ján Suchal, Institute of Informatics and Software Engineering

Spring 2012 PeWe Workshop, April 1, 2012, pp. 69–70.

We have evaluated our method on a real-world database of people occurring in Business Register of the Slovak Republic provided by foaf.sk. There are many duplicates and it is difficult to determine, who exactly occurs in which company. There is a set of heuristics already detecting duplicates on foaf.sk. We have used their results for training and as a baseline for measuring precision, recall and F_1 score. We are comparing names and addresses of people. Our features are string equality (=), Levenshtein distance (L), N-Grams (NG), degree combinations and degree disjunctions.

Table 1: Results

Feature set	Precision	Recall	F_1
=(names), =(addresses)	0.8777	0.9874	0.9293
L(names), L(addresses)	0.8782	0.9923	0.9318
2G(names), 2G(addresses)	0.8777	0.9874	0.9293
3G(names), 3G(addresses)	0.8777	0.9874	0.9293
4G(names), 4G(addresses)	0.8777	0.9874	0.9293
5G(names), 5G(addresses)	0.8777	0.9874	0.9293
6G(names), 6G(addresses)	0.8777	0.9874	0.9293
L(names), L(addresses), Degree combinations	0.8803	0.9622	0.9194
L(names), L(addresses), Degree disjunctions	0.882	0.9777	0.9274

Table 1 shows that using machine learning approach for duplicate detection yields reasonably high precision-recall metrics. However, data samples are simple and further research with more complicated samples needs to be done. Our results have clearly shown, that we need a new data set with our own labels (created manually) instead of baseline foaf.sk labels. Besides name and address attributes, it would be reasonable to include relations of people to companies for better results.

Acknowledgement. This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0675/11.

References

[1] Fellegi, I.P., Sunter, A.B.: Record Linkage, 1969.
[2] Bilenko, M., Mooney, R.: Employing trainable string similarity metrics for information integration. In: Proceedings of the IJCAI-2003 Workshop on Information Integration on the Web, 2003, pp. 67–72.
[3] Hernández, M.A., Stolfo, S.J.: Real-world data is dirty: Data cleansing and the merge/purge problem. Data mining and knowledge discovery, 1998.