An Effective Configuration Learning Algorithm for Entity Resolution

Khai Nguyen and Ryutaro Ichise

The Graduate University for Advanced Studies, Japan
National Institute of Informatics, Japan
{nhkhai,ichise}@nii.ac.jp

1 Introduction

Entity resolution is the problem of finding co-referent instances, which at the same time describe the same topic. It is an important component of data integration systems and is indispensable in linked data publication process. Entity resolution has been a subject of extensive research; however, seeking for a perfect resolution algorithm remains a work in progress.

Many approaches have been proposed for entity resolution. Among them, supervised entity resolution has been revealed as the most accurate approach [6, 2]. Meanwhile, configuration-based matching [2, 3, 5, 4] attracts most studies because of its advantages in scalability and interpretation.

In order to match two instances of different repositories, configuration-based matching algorithms estimate the similarities between the values of the same attributes. After that, these similarities are aggregated into one matching score. This score is used to determine whether two instances are co-referent or not. The declarations of equivalent attributes, similarity measures, similarity aggregation, and acceptance threshold are specified by a matching configuration, which can be automatically optimized by a learning algorithm. Configuration learning using genetic algorithm has been a research topic of some studies [2, 5, 3]. The limitation of genetic algorithm is that it costs numerous iterations for reaching the convergence. We propose *cLearn* as a heuristic algorithm that is effective and more efficient. *cLearn* can be used to enhance the performance of any configuration-based entity resolution system.

2 Approach

A configuration specifies the property mappings, similarity measures, similarity aggregation strategy, and matching acceptance threshold. Property mappings and similarity measures are combined together into similarity functions. Given series of initial similarity functions, similarity aggregation options, and the labeled instances pairs, the mission of *cLearn* is to select the optimal configuration.

cLearn begins with the consideration of each single similarity function and then checks their combinations. When checking the new combination this algorithm applies a heuristic for selecting most potentially optimal configuration. Concretely, the heuristic accepts the new combination if only its performance

Table 1. F1 score of the compared systems on OAEI 2010.

Training size	System	Sider-Drugbank	Sider-Diseasome	Sider-DailyMed	Sider-DBpedia	DailyMed-DBpedia
5%	ScSLINT+*cLearn*	**0.911**	**0.824**	**0.777**	0.6414	**0.424**
	Adaboost	0.903	0.794	0.733	0.641	0.375
Varied by subset	ScSLINT+*cLearn*	**0.894**	**0.829**	**0.722**		
	ObjectCoref	0.464	0.743	0.708		

is better than that of the combined elements. This heuristic is reasonable as a series of similarity functions that reduces the performance has little possibility of generating a further combination with improvement. In addition to finding for similarity functions, the algorithm also optimizes the similarity aggregator and matching acceptance threshold.

cLearn is implemented as part of ScSLINT framework, and its source code is available at http://ri-www.nii.ac.jp/ScSLINT.

3 Evaluation

Table 1 reports the comparison between *cLearn* and other supervised systems, including ObjectCoref [1] and Adaboost-based instance matching system [6]. OAEI 2010 dataset is used and the same amount of training data is given to each pair of compared systems. According to this table, *cLearn* consistently outperforms other algorithms.

cLearn is efficient as the average numbers of configurations that *cLearn* has to check before stopping is only 246. This number is promising because it is much smaller than that of using genetic algorithm, which is reported in [2] with a recommendation of 500 configurations for each iteration.

With the effectiveness, potential efficiency, and small training data requirement of *cLearn* on a real dataset like OAEI 2010, we believe that *cLearn* has promising application in supervised entity resolution, including using active learning strategy to even reduce the annotation effort.

References

[1] Hu, W., Chen, J., Cheng, G., Qu, Y.: Objectcoref & falcon-ao: results for oaei 2010. In: 5th Ontology Matching. pp. 158–165 (2010)
[2] Isele, R., Bizer, C.: Active learning of expressive linkage rules using genetic programming. Web Semantics: Science, Services and Agents on the World Wide Web 23, 2–15 (2013)
[3] Ngomo, A.C.N., Lyko, K.: Unsupervised learning of link specifications: Deterministic vs. non-deterministic. In: 8th Ontology Matching. pp. 25–36 (2013)
[4] Nguyen, K., Ichise, R., Le, B.: Interlinking linked data sources using a domain-independent system. In: 2nd JIST. LNCS, vol. 7774, pp. 113–128. Springer (2013)
[5] Nikolov, A., d'Aquin, M., Motta, E.: Unsupervised learning of link discovery configuration. In: 9th ESWC. LNCS, vol. 7295, pp. 119–133. Springer (2012)
[6] Rong, S., Niu, X., Xiang, W.E., Wang, H., Yang, Q., Yu, Y.: A machine learning approach for entity resolution based on similarity metrics. In: 11th ISWC. LNCS, vol. 7649, pp. 460–475. Springer (2012)