BUPTTeam Participation at TAC 2013 Entity Linking

Xue Yang, Rui Wang, Maolin Li and Yongmei Tan
Center for Intelligence Science and Technology and Technology
Beijing University of Posts and Telecommunications
Beijing, China
{xueyang_bupt,08211444,mlli,ymtan}@bupt.edu.cn

Abstract

This paper overviews BUPTTeam's participation in the Entity Linking task. We tackle the problem of entity disambiguation for large collections of online pages. Specifically, Named Entity Disambiguation is the task to polish the ambiguities of named entities. Several methods have been proposed to solve this problem, but they are largely only focused on one disambiguation mean, Entity Linking or Entity Clustering. In this paper, we propose an original framework to disambiguate the ambiguities of named entities combining Entity Linking and Entity Clustering. The evaluation results show that our method is effective for Entity Linking task.

1 Introduction

Named Entity Disambiguation (NED) has attracted a lot of attention in recent years. The task of Named Entity Disambiguation is to identify entities by eliminating ambiguities.

Currently, Named Entity Disambiguation is divided into two means: Entity Linking (EL) and Entity Clustering. Entity Clustering realizes the purpose of disambiguation by using clustering method. The task of Entity Linking is that mapping the given mention to the entry in KB. Large scale of knowledge base spurred great interests in the Entity Linking task. Wikipedia[1] is the most popular knowledge base in current research. This method also has insufficient, such as the coverage of KB. Although the scale of KB is growing, the KB cannot include all mentions. The difficulty of Entity Linking task derived from three characteristics of named entity. 1) Name Ambiguity: names are often polysemy in that they are shared by different entities; 2) Name Variations: entities are often characterized by synonymy, being referred to by different name variants or aliases; 3) Absence: this coursed by the constraints of knowledge base, many entities will not appear in KB.

Each of the above two methods has its metrics, but they cannot tackle disambiguation problem in the round. Entity Clustering can cluster the mentions, but cannot give the exact meaning of the mentions. Entity Linking can cover the shortage, but it is subject to constraints of the scale of KB. For solving these problems, we propose a framework combining Entity Linking and Entity Clustering to tackle Named Entity Disambiguation problem. We first use Entity Linking method to map the mention to the entity entries in KB and give the corresponding entry id. If the framework cannot find the entry, this mention will label as a NIL mention. Then we cluster the NIL mentions using Entity Clustering method.

The main contributions of this paper are summarized as follows.

- We propose a framework which combines Entity Linking and Entity Clustering to solve Entity Disambiguation.
- We use an overall features extracting method to calculate the similarity among entities.

We extensive evaluate the performance of our framework over two public data sets and empirical

[1] http://www.wikipedia.org/

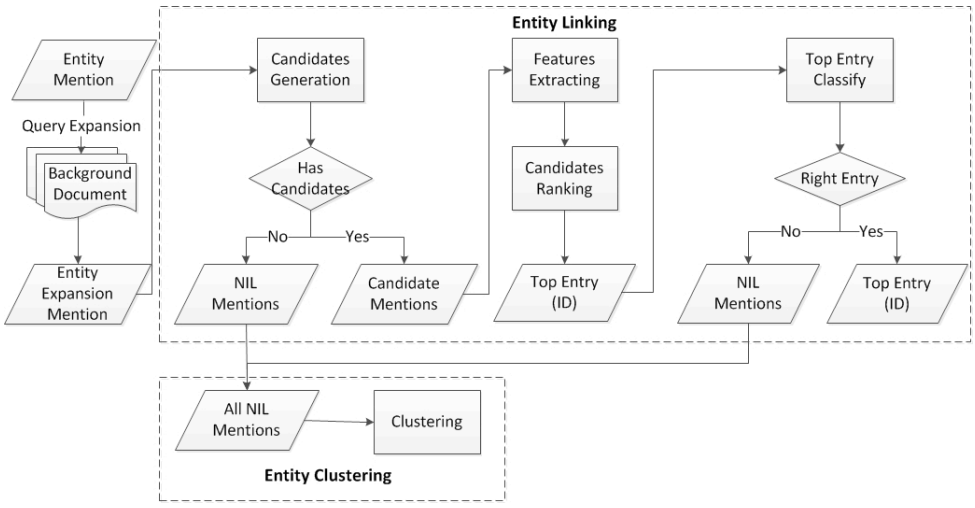

Figure 1. Framework

results show that our framework can achieve a high F-measure.

2 Related Work

Before emergence of the large scale knowledge base, disambiguation problem is regarded as a clustering task. This research started from (Bagga and Baldwin, 1998). Mann and Yarowsky added multiple features based on the work in (Mann and Yarosky, 2003). Bekkerman and McCallum used the MDC to solve the Named Entity Disambiguation problem in (Bekkerman and McCallum, 2005). Han et al. used SC to tackle the same task in (Han et al., 2005). On et al. propose two scalable graph partitioning algorithms known as multi-level graph partitioning and merging to tackle the large-scale Named Entity Disambiguation problem in (Byung et al., 2012).

As several knowledge bases like DBpedia [2] (Auer et al., 2007) and YAGO[3] (Suchanek et al., 2007; Suchanek et al., 2008) are available publicly, researchers have shown a great interest in Entity Linking which maps the textual entity mention to its corresponding entity entry in knowledge base. Bunescu and Pasca first tackled this problem by extracting multiple features from Wikipedia for disambiguation in (Bunescu and Pasca, 2006). Wei et al. propose a novel framework in (Shen et al., 2012). They linked named entities in text with a knowledge base unifying Wikipedia and WordNet[4](Miller,1995; Fellbaum, 1998). Zhang et al. also proposed three advancements for Entity Linking in (Zhang et al., 2011).

However, previous methods largely only focused one aspect of disambiguation means. The clustering methods can cluster the same entities together, but they cannot give the exact meaning of each group of entities. Entity Linking methods can give the exact meaning, but they are subject to the constraints of knowledge base scale. Hence, we propose an original framework to overcome the deficiency of previous methods combining Entity Linking and Entity Clustering.

3 Framework and Notations

In this paper, we propose an original framework combining Entity Linking and Entity Clustering. Entity Linking is defined as the task to map the mention *m* to the entity entry *e* in the Wikipedia knowledge base. If the mention *m* matches no entry, we label that mention as a NIL mention *nil*. For the NIL mention *nil*, our framework clusters the *nil*s which are referred to the same entity together in the Entity Clustering step. The framework is described in Figure.1.

Our framework with two modules as follows:

- **Entity Linking**
  For each mention $m \epsilon M$, we generate the set of candidates of entity entries E_m from the set of entities *E*. The entities set is extracted using multiple sources in Wikipedia. Our framework exploit a measure to rank E_m and find the top node e_{top} for each *m*. After ranking, we detect the mention-entity pair $<m,e_{top}>$ whether is the right pair, if the e_{top}

[2] http://dbpedia.org/About
[3] http://www.mpi-inf.mpg.de/yago-naga/yago/
[4] http://wordnet.princeton.edu/

in the pair is the right entry for the *m*, our framework will give the id of e_{top}, or our framework label *m* as a NIL mention *nil*.

- **Entity Clustering**
  We exploit a measure a clustering method to cluster *nil* mentions which are labeled as NIL mentions in Entity Linking step.

Some notations in this paper are summed up in Table 1.

Notation	Explanation
M	Named mention set
$m \epsilon M$	A named mention required to be linked
E	Entity set of knowledge base
$e \epsilon E$	An entity entry in the set of E
E_m	The set of candidates for mention *m*
$<m, e_{top}>$	Top mention-entity pair after ranking
nil	The un-linkable mention
string	The surface format of *m*

Table 1. Notations

4 Entity Linking

In this step, our framework achieves four goals: Mention Expansion, Candidates Generation, Candidates Ranking and NIL Mention Detection.

4.1 Mention Expansion

Our framework expands mentions using the following methods:

- For acronyms, we assume the length of the given mention *m* is *n*. In the method, first find the text label like string (*m*) or *m* (string). Then we find all words which is start with the first letter of m, and expanding forward or backward around this word.
- For part words, if mention is wholly contained in a string of named entity in the associated document, this named entity is selected as the expansion.

4.2 Candidates Generation

After expanding the queries, our framework generates the possible candidates set E_m for each mention *m*. Our framework uses following resources to generate candidates, Wikipedia titles, redirect titles, anchor texts and disambiguation pages.

In Candidates Generation step, our framework attempts that the entity *e* in the knowledge base is a candidate if mention *m* matches one of the above-mentioned surface.

4.3 Candidates Ranking

In this part, we rank all the retrieved candidates to filter independent candidates using ranking-SVM. The features are described as Table 2.

Feature Category	Feature Names
Surface Features	String Exact Match, String Expansion Match, String Part Match, Acronym Match, String Match based on Edit Distance, String Match based on LCS (Lonest Common Subsequence)
Source Features	Generation Source, String Match
Semantic Features	NER (Named Entity Recognition) Match
Contextual Features	Contextual Similarity
Position Features	Mention in Entity Text, Mention Expansion in Entity Text, Entity in Mention Document
Popularity Features	Popularity

Table 2. Features Overview

After extracting features, our framework uses Ranking-SVM method to rank the candidates, then our framework selects the top node.

4.4 NIL Mention Detection

The top node e_{top} which is selected in section 4.3 is tested by a binary classifier to determine if it is believed as the target entry for a name mention. If not, the mention is labeled as a NIL mention *nil*.

5 Entity Clustering

In this step, our framework clusters the NIL mentions *nil*s. The algorithm is introduced below.

1) The NIL mentions *nil*s are divided to subsets according to their NE types;

2) For each subset, the NIL mentions *nil*s are further clustered based on their names. Our framework clusters the mention pair which conform the following conditions.

- Mentions with the same name;
- The name of a mention wholly contained or contains the name of the other mention;

- The mention pair has a strong string similarity which is calculated by edit distance algorithm;
- The first letter of each word in a mention matches another mention.

After the above strategies, the NIL mentions *nil*s are divided to some subsets. Then, we rely on HAC (Hierarchical Agglomerative Clustering) to further cluster the mentions.

6 Experimental Results and Discussion

6.1 Scoring Metric

We use the KBP tract evaluation method which is a modified B-Cubed metric (called B-Cubed+). The correctness of the relation between two entity mentions *e* and *e'* is described in formula (1).

$$G(e,e') = \begin{cases} 1 \ iff \ L(e') \wedge C(e) = C(e') \wedge GI(e) \\ \quad = SI(e) = GI(e') = SI(e') \\ 0 \ otherwise \end{cases} \quad (1)$$

In formula (1), *L(e)* and *C(e)*: the category and the cluster of an entity mention *e*, *SI(e)* and *GI(e)I*: the framework and the gold-standard KB identifier for an entity mention *e*.

The precision and recall formula is described in formula (2) and (3),

$$Precision = Avg_e[Avg_{e',C(e)=C(e')}[G(e,e')]] \quad (2)$$

$$Recall = Avg_e[Avg_{e',L(e)=L(e')}[G(e,e')]] \quad (3)$$

and the F-Measure id described in formula (4).

$$F - Measure = 2 \times precision \times \frac{Recall}{Precision + Recall} \quad (4)$$

6.2 Experimental Results

We submitted runs for three system variants which are seen in Table.3.The highest score is 0.4499.

Run	Precision	Recall	F-Measure
1	0.4502	0.4495	**0.4499**
2	0.5222	0.4037	0.4424
3	0.5199	0.3693	0.4318

Table 3. English Entity Linking Evaluation Results

7 Conclusion

Entity disambiguation is a very important task for many applications such as Information Retrieval and Question Answering. In this paper we propose an original framework to disambiguate entity with entity linking and entity clustering. A large number of experiments were conducted over two public data sets. Experimental results show that our framework is efficiently.

References

A. Bagga and B. Baldwin. Entity-based cross-document coreferencing using the vector space model. In Proceedings of COLING, pages 79–85, 1998.

Bekkerman R, McCallum A (2005) Disambiguating web appearances of people in a social network. Proceedings of international world wide web conference, 2005

Byung-Won On, Ingyu Lee and Dongwon Lee. Scalable Clustering Methods for the Name Disambiguation Problem. Knowledge and Information Systems 2012 (32), 129-151

Christiane Fellbaum (1998, ed.) WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press.

F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: A core of semantic knowledge unifying wordnet and wikipedia. In Proceedings of WWW, pages 697–706, 2007.

F. Suchanek, G. Kasneci, and G. Weikum. Yago: A Large Ontology from Wikipedia and WordNet. Journal of Web Semantics, 6(3):203–217, 2008.

George A. Miller (1995). WordNet: A Lexical Database for English. Communications of the ACM Vol. 38, No. 11: 39-41.

G. S. Mann and D. Yarowsky. Unsupervised personal name disambiguation. In Proceedings of CONLL, pages 33–40, 2003.

Han H, Giles C, Zha H (2005) Name disambiguation in author citations using a k-way spectral clustering method. Proceedings of ACM/IEEE joint conference on digital libraries, June 2005

R. Bunescu and M. Pasca. Using Encyclopedic Knowledge for Named Entity Disambiguation. In Proceedings of EACL, pages 9–16, 2006.

S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, and Z. Ives. Dbpedia: A nucleus for a web of open data. In Proceedings of ISWC, pages 11–15, 2007.

Wei Chan, Jianyong Wang, Ping Luo, Min Wang. LINDEN: Linking Named Entities with Knowledge Base via Semantic Knowledge. Proceeding of the 21st international conference on world wide web, 2012

Wei Zhang, Yan Chuan Sim, Jian Su, Chew Lim Tan. Entity Linking with Effective Acronym Expansion, Instance Selection and Topic Modeling. Proceddings of the Twenty-Second International Joint Conference on Aritificial Intelligence,2011