BUPTTeam Participation at TAC 2012 Knowledge Base Population

Yongmei Tan, Zhichao Wang, Xue Yang, Zhihao Wang and Lei Liu
Center for Intelligence Science and Technology and Technology
Beijing University of Posts and Telecommunications
Beijing, China
{ymtan,XueYang_bupt,wzh_bypt}@bupt.edu.cn

Abstract

This paper overviews BUPTTeam's participation in the Entity Linking task at TAC 2012. In this paper we propose a method to link the queries and KB entries based on four steps: 1) query expansion, 2) candidates generation, 3) candidates ranking, 4) clustering. In the initial stage, we expand the queries from the background documents using the defined rules. We generate in the knowledge base according to the expansions. In entity linking, the most crucial step is ranking the KB candidates and selection the best node. In ranking stage, we use some features to find the best node. For a top-ranked node, we need to detect it whether is the corresponding KB entry. We target the node which is not in the knowledge base as "NILxxxx" and cluster them. The evaluation results show that our method is effective for RTE task.

1 Introduction

The goal of Knowledge Base Population (KBP) track at Text Analysis Conference (TAC) 2012 is to automatically discover information about named entities and to expand this information in a reference knowledge base. The main task is divided into two subtasks: Entity Linking (EL) and Slot Filling (SL).

In the Entity Linking task, given a query that contains a name string, a source document ID, and a pair of UTF-8 character offsets indicating the start and end location of the name string in the document, the main demand of the system is to provide the ID of the KB entry to which the name refers, or a "NILxxxx" ID if there is no such KB entry. In addition, the system needs to cluster the entities which are targeted as "NILxxxx". For example, when seeing a sentence "I was born in Washington, DC.", the query "Washington, DC" should be linked to the Wikipedia page "http://en.wikipedia.org/wiki/Washington,_D.C.". An Entity Linking system should link the query to the corresponding KB entry, or determine that no corresponding entry exists.

The method is divided into four parts. In the initial stage, the method expands the query to a richer set of forms using Wikipedia structure mining or coreference resolution in the source documents. We find all possible KB entries that a query might link to as candidates in the following stage. In the third stage, the system ranks all candidates and finds the best node. In the final stage, we detect the NILs which are not in the reference knowledge base and cluster them.

2 Related work

Since the first KBP track held in 2009, the research in the area of entity linking has greatly developed. State-of-the-art method has been proposed by Monahan et al. (2011). In this method, the problem of entity linking is recast as one of cross-document entity coreference. The team compare an approach where deductive entity linking informs cross-document coreference to an inductive approach where coreference and linking judgments are mutually beneficial. LCC team takes an inductive approach which treats the problem as cross-document coreference which entity linking. Rather than only clustering the detected NIL mentions, they cluster all entities while using output from entity linker as suggestions but not fact. This is not counter to the deductive approach which first links all of the entities and then cluster the remaining NIL mentions.

Silviu Cucerzan (2011) suggests a way to extracts and disambiguates globally all entities from each target document and then maps the target string to one of the entities extracted from the document instead of focusing only on the provided target strings. The main features employed by the system are topics associated with the entities in the knowledge base, which are derived from Wikipedia categories, list pages, and lexico-syntactic link pattern.

Wei Zhang et al.(2011) propose an approach for system is done through three steps: 1) Expanding query to reduce the ambiguities of the mention 2) linking the entities to KB entries or NIL and 3) clustering NIL queries. The team expands the query from its context can effectively reduce the ambiguities of the mention, under the assumption that two name variants in the same document refer to the same entity. Candidate generation finds all the possible KB candidates for the given query using Wikipedia source and string match. In their ranking stage, using a learning to rank method to rank all candidates and find the best node to detect it weather is the "NIL" node. They use three methods to cluster: Spectral Graph Partitioning, Hierarchical Agglomerative Clustering and Latent Dirichlet Allocation.

3 System architecture

The system of mono-lingual entity linking system architecture is described in figure 1. It includes 5 steps:

- Extracting queries and processing
- Candidates generation
- Candidates ranking
- Clustering

The architecture of the system is described in figure 1. We will describe each part in detail in the next sections.

4 Processing

4.1 Extracting queries and processing

The system extracts queries' information consists of a query ID (query_id), a name string (query_name), background document ID (query_docid), the background document's content and the location of the query in the corresponding document (query_loc). The method classifies queries into three types using Stanford Named Entity Recognizer and trains models specifically for each entity type.

In the initial stage expanding queries from documents can effectively reduce the ambiguities of the mention and narrow the range of candidates. For example, "Washington" in Wikipedia refers to three entities, but its expansion "Washington, DC" only refers to the capital of the USA. For a given query, the system expands it using the following approach:

1) For the capitalized query, we use the acronym expansion method to expand it from its context. The method finds all strings which begins with the corresponding letters. The correct strings must be conformed the following rules:

- The number of capitalized letters is len_q.
- All capitalized letters contain in the expansion word.
- The lowercase word can only be one of "and", "of", "for", "in" and "at".

2) The method defines different rules for each entity type:

- For a GPE query, if the query followed by a comma, we extract two words towards the back. If the two words have first capitalized letter and no lemma, the method expand the two words to the given query or expand the first word to the given query.
- For a PER query, we extract two words forward. If the first word begins with capitalized letter and ends with a dot, the method expands this word to the given word. If the second word begins with capitalized letter and ends with a dot, the method expands this word to the given word. If the second word begins with capitalized letter and ends with a lowercase letter, the method expands this word to the given word.
- For an ORG query, we extract two words forward and two words toward the back.

The system normalized the expansion words after expansion.

4.2 Candidates Generation

In this stage, the system attempt to identify every potentially correct KB entries for the query

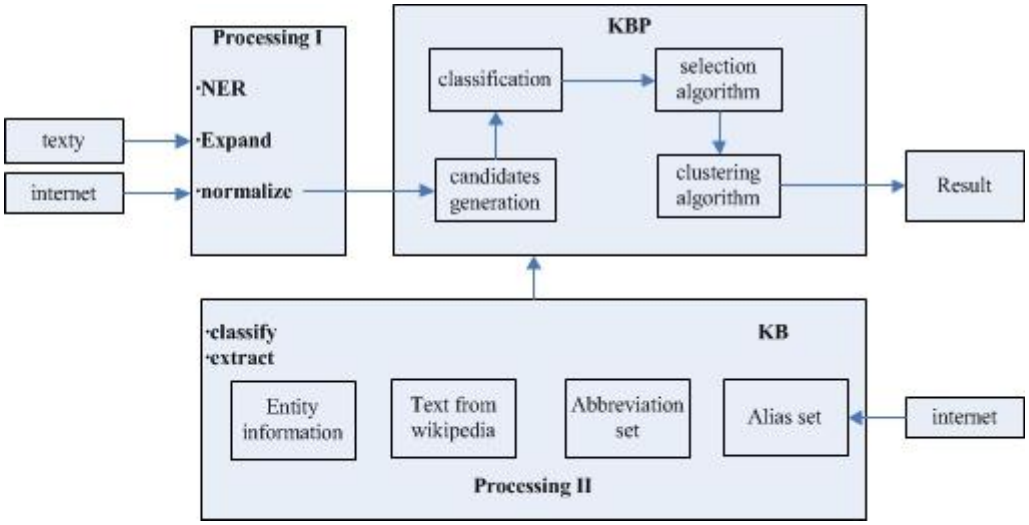

Figure 1 The system architecture

mention string. Following are the candidate generators used to map entity strings to potential referents.

Name variants: maps queries to the variants in the corresponding reference knowledge base entry.

Anchor text: maps queries to the anchor text in the corresponding reference knowledge base entry. Anchor text usually gives the user relevant descriptive or contextual information about the content of the link's destination.

Edit Distance: maps queries to the Edit Distance algorithm results. Edit distance between two strings of characters generally refers to the distance in which insertions and deletions have equal cost and replacements have twice the cost of an insertion. It may also refer to the whole class of string metrics that measure distance as the number of the operations required to transform a string into another.

Google API: maps queries to URLs from the Google API, which are constrained to the native Wikipedia website.

4.3 Candidate Ranking

In entity linking, the most crucial step is ranking the KB candidates and selection the best node. In the candidate ranking stage, the system extracts the best node in the candidates. Used features are as follows:

Surface Features: focus on the queries independent of context. Such as exact string match, acronym match, alias match, string match based on edit distance and name component match. These features test the similarity between the mention string and the candidate sense's name string.

Generation Features: indicate the origin of the candidate sense since some candidate sources are noisier than other. Another feature indicates the number of sources which generated the candidate.

Semantic Features: combine surface and contextual evidence to provide the entity types for the mention and candidate, along with their compatibility. In our system, named entity recognition is applied to document context for each mention.

Context Similarity Features: focus on the similarity between the query's background document and the corresponding candidates' documents.

After ranking, the method select the best node and detect the top-ranked candidate whether is the correct KB entry to give the entry ID or target as "NILxxxx". The selection algorithm is described as follows:

```
the candidate set of the query is cand ={c1,c2,…cn}
if count(cand)= 0:
output is NIL
else:
        if count(cand) = 1:
        output is c1
        else
        the threadshod is x:
        if all candidate's score is under x:
        output is NIL
        else
        output is the Optimal solution
```

4.4 Clustering

Several words in different sentences are not exactly matched but they are referred to the same identity, such as 'Bush' and 'George. W. Bush'. Within a single sentence, pronoun can be referred to a noun phrase. This problem can be resolved by coreference identification in order to ignore the differences in form and indicate the same identity.

In final stage, the method cluster the NIL queries based on the identity of the entity, such that queries referring the same entity are converged into the same cluster and give each cluster a signal ID as “NILxxxx”.

We use the same features as ranking stage to cluster NILs. Lots of NIL queries can be separated from each other by named entity type comparison. For example, the PER “Washington” and the GPE “Washington” can be separated by NE types. We first normalized the queries. Then cluster them according to the NE types and string match features.

5 Results and Discussion

We submitted runs for three system variants which are seen in Table 1.

Table 1 Submissions scores

Submission	F
BUPTTeam1	0.445
BUPTTeam2	0.578
BUPTTeam3	0.561

The method’s highest score is 0.578, each part’s score is shown in Table 2.

Table 2 BUPTTeam2 each score

Part	B^3+ F1
All	0.578
In KB	0.453
Not in KB	0.719
NW docs	0.631
WB docs	0.475
PER	0.703
ORG	0.523
GPE	0.451

As shown in Table 2, GPE has lowest score. For our system, GPE is still the most difficulty entity type. For some small location names, a system will need to acquire world knowledge in order to disambiguate them. The result of the queries on the web is unsatisfactory. According to our observation, the WB queries are not standardized compared to the NW queries.

6 Conclusions

In Entity Linking can be viewed as more traditional Data Mining where information mined from text can be connected to and stored with preexisting knowledge of entities in the world.

In our system, many useful features have not used in the candidates ranking and clustering. In the future research, we plan to use more features to improve system’s performance. The features are as follows:

Surface Features: In our system, we used some surface features. In the future, we plan to use more features, such as word match (number of the same words between the title of the candidate and the query) and word miss (number of the different words between the title of the candidate and the query).

Contextual Features: used the source document outside of the queries. For example, calculate the cosine similarity between given query’s document and the corresponding KB entry’s document.

Topic Features: similarity between the document and the text of candidate in a topical space. Topic feature provides a natural and effective way to model the context profile of each query.

Acknowledgments

We gratefully acknowledge the support of the Fundamental Research Funds for the Central Universities and the National Natural Science Foundation of China (NSFC) (90920006). Any opinions, findings, and conclusion or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of the Ministry of Education of the People’s Republic of China. Part of this work was performed while the author was visiting USC/ISI, where the project was supported by the Defense Advanced Research Projects Agency (DARPA) Machine Reading Program under Air Force Research Laboratory (AFRL) prime contract no. FA8750-09-C-0172. Any opinions, findings, and conclusion or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of the DARPA, AFRL, or the US government.

References

Sean Monahan, John Lehmann, Timothy Nyberg, Jesse Plymale, Arnold Jung. Cross-Lingual Cross-Document Coreference with Entity Linking.Proc. TAC2011. 2011. nist.gov

Silviu Cucerzan. TAC Entity Linking by Performing Full-document Entity Extraction and Disambiguation .Proc. TAC2011. 2011. nist.gov

Wei Zhang, Jian Su. Bin Chen. I2R-NUS-MSRA at TAC 2011: Entity Linking. Proc. TAC2011. 2011. nist.gov