Combining POS Tagging, Lucene Search and Similarity Metrics for Entity Linking

Shujuan Zhao[1], Chune Li[1], Shuai Ma[⋆,1], Tiejun Ma[2], and Dianfu Ma[1]

[1] SKLSDE Lab, Beihang University, China
[2] University of Southampton, UK
{zhaosj@act.,lichune@act.,mashuai@,madf@}buaa.edu.cn
tiejun.ma@soton.ac.uk

Abstract. Entity linking is to detect proper nouns or concrete concepts (a.k.a mentions) from documents, and to map them to the corresponding entries in a given knowledge base. In this paper, we propose an entity linking framework POSLS consisting of three components: mention detection, candidate selection and entity disambiguation. First, we use part of speech tagging and English syntactic rules to detect mentions. We then choose candidates with Lucene search. Finally, we identify the best matchings with a similarity based disambiguation method. Experimental results show that our approach has an acceptable accuracy.

Keywords: Entity Linking, POS Tagging, Lucene Search, Similarity Metrics, Mention Detection

1 Introduction

Entity linking is to detect proper nouns or concrete concepts (a.k.a mentions) from documents, and to map them to the corresponding entries in a given knowledge base (KB) [1]. The research has attracted a lot of interests since its invention, due to the rapid expansion of Web information that leads to a great need of extracting useful knowledge from the Web. Moreover, the structured Web knowledge, i.e. Wikipedia[1], is increasingly becoming mature, which makes it possible to dig out more detailed information. For example, question answering tasks first find the expansion terms or synonyms of questions by linking user queries to Wikipedia, and then search answers with these synonyms [2]. However, linking entities manually is tedious and requires a lot of efforts. Our goal is to link mentions automatically in documents to Wikipedia URLs to significantly reduce manual efforts. Thus we propose an entity linking framework POSLS, which combines Part-Of-Speech tagging [3], Lucene search [4] and similarity metrics [5] for entity linking, and we correlate a target mention with its corresponding unique

⋆ Contact author, and Shuai is supported in part by NGFR 973 grant 2014CB340300 and 863 grant 2011AA01A202, SKLSDE Lab grant SKLSDE-2012ZX-08, and the Fundamental Research Funds for the Universities.

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

KB entry. Specifically, the KB here contains 2,860,422 entries, and each entry is associated with two columns: entity names (in forms of Wikipedia URLs) and the collected mentions associated with the entities.

Given a corpus of plain texts and a KB as inputs, our approach first utilizes the Stanford's POS tagging technique [3] to analyze categories (verb, noun or adjective etc.) of every word, after which proper nouns are detected. Moreover, most of the concrete concepts could be identified based on the relationships of adjacent words. Next, the candidate URLs (entities) corresponding to a mention are to be detected. In order to speed up the search process, we treat each entry of the KB as a document and index them in Lucene search engine [4]. After querying the mention using Lucene, we further decide the best matching URLs, by comparing the similarity between the detected mentions and the given entity mentions in the KB for candidate URLs with similarity metrics, e.g., q-grams and edit distance (see [5] for a survey). Meanwhile, we also utilize history information for the disambiguation of some proper nouns in the process.

In conclusion, our main contributions are (1) an entity linking framework combining existing techniques, such as POS Tagging, Lucene search and similarity metrics, (2) a set of English syntactic rules for entity detection, and (3) an entity disambiguating method based on similarity metrics.

Organization. The paper is organized as follows. Section 2 depicts the related work. Section 3 describes how our framework POSLS works for entity linking. Section 4 contains an experimental analysis, followed by the conclusion section.

2 Related Work

The challenges of entity linking lie in mention detection and entity disambiguation. As to detecting mentions, Medelyan et al. [6] proposed an n-gram method, using a sliding window on the input article and comparing every n-gram with stop words omitted (stop words appear frequently and have no special meanings, such as "a" and "the"). Mihalcea and Csomai [7] constructed a controlled vocabulary composed of Wikipedia article titles and surface forms (anchor texts that refer to other Wikipedia links), to which is referred in mention detection. However, it is costly and time-consuming to prepare a vocabulary bank and use n-gram searching when the input document is large. Mendes et al. [8] firstly used LingPipe Extract Dictionary-Based Chunker [9] and then exploited POS tagger of LingPipe to get rid of mentions that were made of verbs, adjectives, adverbs and prepositions. Our approach is similar, but we first make the POS analysis, and then take advantage of English syntactic rules to find mentions that are primarily nominal phrases.

For entity disambiguation, the key is to compute the relevance with certain similarity metrics to get the top match results. [10] built a vector space based on a bag-of-word model and made use of the cosine similarity, and [11] further used the category feature of Wikipedia attributes. Other better methods may consider overlap (Jaccard Cofficient) between the first paragraph of the Wikipedia

article and input document [7] and the edit distance between titles and mentions [12], or apply intricate interlinks of articles to compute the relevance score of URLs [13]. Generally, pure similarity comparing methods get an ordinary performance in specific domains. Nowadays, similarity metrics combining machine learning [12, 14] or graph model [13] usually obtain a good performance. However, preparing a representative training set is costly and constructing graph is time-consuming. Here we utilize similarity metrics [5] and previous detected mentions in a document to find the best matchings.

3 POSLS: An Entity Linking Framework

Given a corpus of documents and a knowledge base with Wikipedia URLs and their mapping mentions, our entity linking framework POSLS automatically detects mentions (proper nouns and concrete concepts) in these documents, and links them to best matching KB entries. The final output contains documents IDs, offsets of the detected mentions in the documents, detected mentions and their matching URLs in KB.

Table 1. English Syntactic Rules for Concrete Mentions.

No.	**Rules**	**Meanings**	**Explanations**
R_1	(NN| NNP| NNS| NNPS)$^+$	mentions composed of nouns	noun phrases: one or more singular, plural or proper nouns
R_2	(JJ| JJS)$^+ \cdot R_1$	mentions starting with adjectives and ending with nouns	one or more adjectives followed by R_1
R_3	$R_1 \cdot$ 'of' $\cdot$ S$^* \cdot R_1$	mentions with "of", such as "History of China"	noun phrases followed by "of", an arbitrary string and R_1
R_4	$R_1 \cdot$ T $\cdot R_1$	mentions with the genitive marker, e.g., "Stanford's POS tagging technique"	noun phrases followed by T and R_1

Symbols *, $^+$, | and $\cdot$ denote any number of occurrences, one or more occurrences, alternation and concatenation, respectively. NN: singular noun or mass; NNP: proper singular noun; NNS: plural noun; NNPS: proper plural noun; JJ: adjective; JJS: superlative adjective; S: characters; T: the generative maker 's or '.

3.1 Detecting Mentions

The first step is to detect possible mentions appeared in the input documents. We utilize the Stanford's POS tagging technique [3], which further uses the Penn Treebank Tag set [15] to get proper nouns and acquire concrete concepts, in terms of a set of English syntactic rules expressed in regular expressions, shown in Table 1. The rules are reasonable for mentions that are nominal phrases.

The four rules could embody necessary possible concepts appeared in English articles. Experiments also show that the method could detect mentions with a

high recall. Besides, we ignore common single words, such as "trip" and "school", from a list of 1500 most frequently used nouns[2].

3.2 Searching Candidates

The second step is to find a list of URL candidates from the given KB for each mention. Ambiguity is common in English because of polysemy, difference of contexts and morphological diversity (acronym, abbreviation and alterable order) [16]. For instance, "tree" may refer to "plant tree" or "tree data structure"; although "China" could be a country name, it could also refer to "the history of China" or "the culture of China" in different contexts. In the meanwhile, the acronym "KB" might have various linkings, which could refer to "Knowledge Base", "Kilobyte", or even a bank of Iceland –"Kaupthing Bank".

Realizing that the KB entries are excessive, we make use of Apache Lucene, a fast retrieval software library, to select matching URL candidates. We first tokenize URLs and entities in the KB, and build an inverted index for each token by treating each line as a small document. Different weights are set to tokens appeared in left and right columns under the assumption that there are almost no errors in the URL fields. Detected mentions in first step are sent to Lucene as queries, and the software ranks the given results based on the TF/IDF relevance between the mention and each line. We choose the top 40 as candidates.

3.3 Entity Disambiguation

The last step is to determine the best matching URL from the candidate set for each detected mention. Considering that the KB has provided mapping pairs between the URLs and entity mentions, we compare the similarities between the detected mentions and the provided entity mentions in the KB. Due to the diversity of expressions, there may be more than one collected entities, separated by "##", that corresponds to the same URL. We utilize similarity metrics, e.g., q-grams, edit distance and Jaro-Winkle [5], for entity disambiguation.

The method works as follows. For each candidate URL, we first get its provided mapping mentions in the KB. We then compare them with the detected mention based on similarity metrics. Meanwhile, we record the current most similar URL and its similarity value, which are updated constantly. When a URL has a similarity between its some given mention and the detected mention that is larger than the threshold, it becomes a possible mapping. Then we compare the similarity value with the current stored value. If the new value is higher, we update the most similar result and maximum similarity accordingly. In addition, we notice that one entity mention may appear in several KB entries, caused by the morphology of English. Therefore, when two URLs have the same mapping entity mention, we further compute the similarity of URLs and the detected mention. The one with a higher similarity becomes the most similar URL. Finally, the most similar URL is treated as the best matching. Moreover, we make use of history information to match some proper nouns which are parts of the

[2] http://www.talkenglish.com/Vocabulary/Top-1500-Nouns.aspx

other nouns. For these proper nouns, we just copy their corresponding complete mapping results directly. Though we mentioned that similarity metrics generally have an ordinary performance in entity linking, it is not true for the situation that the KB already gives the matching entries of URLs and entity mentions.

4 Experimental Study

The testing dataset is a corpus of domain-independent articles for entity linking, provided by the WISE challenge 2013. With the given correct answers, we evaluate the performance of our English syntactic rules and POSLS framework.

4.1 English Syntactic Rules Evaluation

By comparing with the correct answers, we tested our mention detection method with four datasets, each consisting of 100 or 200 articles. The results have a high average recall of 0.749, as shown in Table 2.

Table 2. English Syntactic Rules Results Evaluation

Group	# of articles	Recall
1	100	0.75603
2	100	0.73036
3	200	0.75294
4	200	0.76595

4.2 The POSLS Framework Evaluation

The POSLS method is implemented in JAVA, and all experiments are conducted on a dual core computer with 4G memory. We conducted a series of experiments on several data sets with the edit distance, q-gram and jaro-winkler metrics. The results are shown in Tables 3 and 4 with 100 and 600 testing articles, respectively. The results are computed by using exact equality of the detected mentions and standard answers (redirecting URLs are treated as correct for they refer to the same Wikipedia articles). In addition, if we consider those correct linkings (not listed in given answers), the following results could be much better.

Table 3. POSLS Results Evaluation with Different Similarity Metrics (100)

Similarity Functions	**Precision**	**Recall**	**F-measure**	**Time**
Edit Distance	0.2972973	0.5306122	0.3810793	84959ms
Q-gram	0.2701950	0.4948080	0.3495496	86345ms
Jaro-Winkler	0.2116380	0.4740260	0.2926271	82861ms

Table 4. POSLS Results Evaluation with Different Similarity Metrics (600)

Similarity Functions	**Precision**	**Recall**	**F-measure**	**Time**
Edit Distance	0.2720268	0.4491751	0.3388446	454381ms
Q-gram	0.2418758	0.4110738	0.3045528	466615ms
Jaro-Winkler	0.1916144	0.4035235	0.2598420	421335ms

From the experiment results, we can see that the edit distance similarity has the best performance in precision and recall, while the q-gram metric has the the best performance in efficiency. In general, our POSLS framework has an acceptable precision and recall.

5 Conclusion

In this paper, we propose an entity linking framework POSLS which combines POS tagging, Lucene search and similarity metrics. First, we built a set of English syntactic rules according to POS tags to detect proper nouns and concrete concepts. Then a group of candidates were selected, using Lucene search. Finally, we introduced a similarity based entity disambiguation method. We also experimentally verified the effectiveness and accuracy of the POSLS framework.

There is much to be done in the future. More sophisticated English syntactic rules are to be developed to improve the mention detection quality. Further, we are exploring new techniques to improve the efficiency and accuracy.

References

1. Mark Dredze, Paul McNamee, Delip Rao, Adam Gerber, and Tim Finin. Entity disambiguation for knowledge base population. In *COLING*, 2010.
2. Ian MacKinnon and Olga Vechtomova. Improving complex interactive question answering with wikipedia anchor text. In *ECIR*. 2008.
3. Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *HLT-NAACL*, 2003.
4. Lucene Search. http://lucene.apache.org/.
5. Ahmed K. Elmagarmid, Panagiotis G. Ipeirotis, and Vassilios S. Verykios. Duplicate record detection: A survey. *IEEE TKDE*, 19(1):1–16, 2007.
6. Olena Medelyan, Ian H Witten, and David Milne. Topic indexing with wikipedia. In *the Wikipedia and AI workshop at AAAI-08*, 2008.
7. Rada Mihalcea and Andras Csomai. Wikify!: linking documents to encyclopedic knowledge. In *CIKM*, 2007.
8. Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. Dbpedia spotlight: shedding light on the web of documents. In *I-SEMANTICS*, 2011.
9. B Carpenter and B Baldwin. Lingpipe, 2008.
10. Wei Zhang, Jian Su, Chew Lim Tan, and WenTing Wang. Entity linking leveraging automatically generated annotation. In *COLING*, 2010.
11. Silviu Cucerzan. Large-scale named entity disambiguation based on wikipedia data. In *EMNLP-CoNLL*, 2007.
12. Wei Zhang, Jian Su, Chew Lim Tan, Yunbo Cao, and Chin-Yew Lin. A lazy learning model for entity linking using query-specific information. In *COLING*, 2012.
13. Xianpei Han, Le Sun, and Jun Zhao. Collective entity linking in web text: a graph-based method. In *SIGIR*, 2011.
14. Zhicheng Zheng, Fangtao Li, Minlie Huang, and Xiaoyan Zhu. Learning to link entities with knowledge base. In *HLT-NAACL*, 2010.
15. Mitchell P Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. Building a large annotated corpus of English: The Penn Treebank. *Computational linguistics*, 19(2):313–330, 1993.
16. Bernd Kortmann, Edgar W. Schneider, Kate Burridge, Rajend Mesthrie, and Clive Upton. *A handbook of varieties of English A Multimedia Reference Tool. Volume 1: Phonology. Volume 2: Morphology and Syntax.* Mouton de Gruyter, 2004.