Word, Mention and Entity Joint Embedding for Entity Linking

Zhichun Wang, Danlu Wen, Yong Huang, Chu Li

Beijing Normal University, Beijing, China
zcwang@bnu.edu.cn

Abstract. Entity linking is a important for connecting text data and knowledge bases. This poster presents a word, mention and entity joint embedding method, which can be used in computing semantic relatedness in entity linking approaches.

1 Introduction

Recently, several large-scale Knowledge Bases have been created and successfully applied to many areas, such as DBpedia, YAGO, and Freebase. In many applications of knowledge bases, a basic task is to identify entities in text and linking them to a given knowledge base, which is usually called entity linking. The task of entity linking is challenging because of entity name variations and entity ambiguity. On one hand, one entity can be mentioned in text by different names; for example, both ”Beijing” and ”Peking” can refer to the same entity ”Beijing City”. One the other hand, the same mention can refer to multiple different entities; for example, ”Apple” may refer to ”Apple Inc” and the fruit ”Apple”, etc. Lots of work has been done on the problem of entity linking, [5] gives detailed review of all kinds of entity linking approaches.

In the entity linking approaches, computing semantic relatedness between entities and contextual context is very important for entity disambiguation. In this poster, we propose a new way to compute the relatedness. A word, mention and entity joint embedding learning methods is proposed. Based on the results of the joint embedding, different kinds of relatedness among words, mentions, and entities can be easily computed. The rest of this paper introduces the proposed embedding method in detail.

2 Word, Mention and Entity Joint Embedding

In this paper, we propose to use Skip-gram model [3] to jointly map entities, mentions and words to the same low-dimensional vector space. By using the jointly learned vectors, various relatednesses can be efficiently computed, such as *entity-word* relatedness, *mention-word* relatedness and *entity-entity* relatedness.

2.1 The Skip-gram model

The skip-gram model is a recently published learning framework to learn continuous word vectors from text corpora. Each word in the text corpora is mapped to a continuous embedding space. The model is trained to find word representations that are good at predicting the surrounding words in a sentence or a document. Given a sequence of training words $w_1, w_2, ..., w_T$, the objective of the model is to maximize the average log probability

$$O = \frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t) \tag{1}$$

where c is the size of training context, $p(w_{t+j}|w_t)$ is defined as

$$p(w_O|w_I) = \frac{\exp({v'_{w_O}}^{\mathrm{T}} v_{w_I})}{\sum_{w=1}^{W}\exp({v'_{w_O}}^{\mathrm{T}} v_{w_I})} \tag{2}$$

where v_w and v'_w are the input and output vector representations of w, and W is the number of words in the vocabulary. The learned vectors of words can capture the semantic similarity of words; similar words are mapped to nearby places in a low-dimensional vector space.

$$r(a,b) = \frac{\boldsymbol{a}\cdot\boldsymbol{b}}{\parallel \boldsymbol{a} \parallel\parallel \boldsymbol{b} \parallel}$$

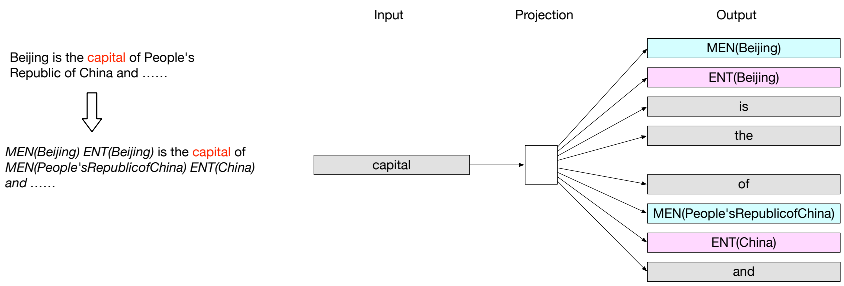

Fig. 1. An example of using the Skip-gram model to predict surrounding tokens of a specific token; here a token can be a word, a mention or an entity.

2.2 Joint Embedding by Skip-gram Model

Skip-gram model is initially designed to learn embeddings of words. In order to extend the word model to a joint model of word, entity and mention, we add mentions and entities in the training corpus which only contains words before. Let the original corpus be $\mathcal{C} = \{w_1, w_2, ..., w_N\}$, if a certain word sequence $s = \{w_i, ..., w_{i+k}\}$ in $\mathcal{C}$ is a mention to an entity e in the knowledge base, we replace s with two tokens $\{MEN(w_i w_{i+1} ... w_{i+k}), ENT(e)\}$; after that, the original word sequence containing

s in $\mathcal{C}$ becomes $\{w_{i-1}, MEN(w_i w_{i+1}...w_{i+k}), ENT(e), w_{i+k+1}\}$. After annotating all the mentions and their corresponding entities, $\mathcal{C}$ is converted to a hybrid corpus $\mathcal{C}'$ that containing words, mentions, and entities. $\mathcal{C}'$ is then used to train the Skip-gram model, which will generate representations in the same vector space for words, mentions, and entities. Figure 1 shows an example of using the Skip-gram model to predict the surrounding tokens of the word *capital* in the example sentence.

2.3 Using Wikipedia as Training Corpus

Annotating mentions and entities in a corpus is a time-consuming task. Fortunately, if we use Wikipedia as a knowledge base, it contains all the annotations we need. Figure 2 shows part of the page of Beijing in Wikipedia and its source text in editing model. In Wikipedia, a internal hyperlink is annotated by [[**entity** | **mention**]]; it also could be [[**entity**]] when the entity is mentioned by the exact name of it. Processing these inner links, we can generate the corpus containing words, mentions and entities together. So in this paper, Wikipedia is used as the target knowledge base that entities link to, and its articles are processed to train the skip-gram model.


Beijing

From Wikipedia, the free encyclopedia

"Peking" redirects here. For other uses, see Beijing (disambiguation) and Peking (disambiguation).

Beijing, formerly **Peking**, is the capital of the People's Republic of China and one of the most populous cities in the world. Its population in 2013 was 21,150,000.[8] The city proper is the 13th largest in the world. The metropolis, located in northern China, is governed as a direct-controlled municipality under the national government with 16 urban, suburban, and rural districts.[9] Beijing Municipality is surrounded by Hebei Province with the exception of neighboring Tianjin Municipality to the southeast; together the three divisions form the Jingjinji metropolitan region and the national capital region of China.[10]

'''Beijing''', formerly '''Peking''', is the capital of the [[China|People's Republic of China]] and one of the most [[List of metropolitan areas by population|populous cities in the world]] ...

Fig. 2. Part of the Wiki page of Beijing and its source text

3 Evaluation

To evaluate the effectiveness of the proposed embedding model, we use the embedding results in a entity linking approach. The entity linking approach was first introduced in [6], and we replace the relatedness measure in [6] with the cosine similarity between vectors of entities. And furthermore, we add relatedness between entities and their contextual words, which is computed as the cosine similarity between vectors of entities and words.

In the evaluation, English Wikipedia is used as the target Knowledge Base for entity linking. The dataset of *Yahoo Search Query Log To Entities* [1] is used for the evaluation.

[1] http://webscope.sandbox.yahoo.com/catalog.php ?datatype=l&did=66

This dataset contains manually identified links to entities in Wikipedia. In total, there are 2,635 queries in 980 search sessions, 4,691 mentions are annotated which link to 4,725 entities in Wikipedia.

We also compared our approach with two entity linking systems, Illinois Wikifier [4, 1] and DBpedia Spotlight [2]. Illinois Wikifier is a entity linking system that was developed by University of Illinois at Urbana-Champaign. DBpedia Spotlight is a system for automatically annotating text documents with DBpedia URIs. Because DBpedia is built from Wikipedia and each DBpedia URI corresponds to a Wikipedia entity, the results of DBpedia Spotlight can be easily converted to entity links of Wikipedia.

Table 1 shows the evaluation results of three different approaches. The precision and recall of each approach are evaluated. According to the results, our approach achieves the best precision and recall. It shows that the joint embedding model is effective in entity linking problem.

Table 1. Evaluation Results

Approach	Precision	Recall
DBpedia Spotlight	0.44	0.65
Wikifier	0.45	0.5
Ours	0.57	0.784

References

1. X. Cheng and D. Roth. Relational inference for wikification. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013.
2. J. Daiber, M. Jakob, C. Hokamp, and P. N. Mendes. Improving efficiency and accuracy in multilingual entity extraction. In *Proceedings of the 9th International Conference on Semantic Systems (I-Semantics)*, 2013.
3. T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc., 2013.
4. L. Ratinov, D. Roth, D. Downey, and M. Anderson. Local and global algorithms for disambiguation to wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1375–1384, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.
5. W. Shen, J. Wang, and J. Han. Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering*, 27(2):443–460, Feb 2015.
6. Z. Wang, J. Li, and J. Tang. Boosting cross-lingual knowledge linking via concept annotation. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI '13, pages 2733–2739. AAAI Press, 2013.