Collective Entity Linking and a Simple Slot Filling Method for TAC-KBP 2011

Zhengyan He, Houfeng Wang
Key Laboratory of Computational Linguistics (Peking University)
Ministry of Education,China
hezhengyan.hit@gmail.com
wanghf@pku.edu.cn

Abstract

This paper summarize our work in TAC2010 knowledge base population track. We submit result for english entity linking and regular slot filling task. For entity linking we use a frequency based method as baseline and implement a collective method following (Han et al., 2011) for entity linking. For slot filling, we use wikipedia infobox as a source of supervision by mapping back to sentences to generate more training sentences. Evaluation result shows the strength and weakness of both our approaches.

1 Introduction

We participate in two main tasks of TAC 2011 knowledge base population track: english entity linking and regular slot filling. Entity linking is the task of link an entity occurence in text to an entry in knowledge base. Each query is a mention with its context document representing a potential entity in the knowledge base. The output is an entry in knowledge base that this mention describes, or nil if the refered entity is not in the knowledge base. Slot filling is the task of extracting from large collection of text the attributes of a given entity. The query is an entity (may or may not be in knowledge base) with an context document, the output is a list of attribute values of this entity. The attribute type is predefined for person and organization.

(Ji and Grishman, 2011) has a summary of last year's TAC KBP conference on successful approaches and methods used . They describe the main parts of both of the two task. However, the task remains very complex and needs very careful tuning in order to achieve satisfactory result.

2 Entity Linking Approach

2.1 Build entity lookup dictionary

Real world entity often has synonym(that is multiple names point to the same entity) and polysemy(that is one surface form has many possible entities). In entity linking,this ambiguity in entity linking is often solved by a lookup dictionary from surface form to entity list.

Following (Cucerzan, 2007), we first build a lookup dictionary from surface form to entity list. Specifically, we utilize the redirect page, disambiguation page and wikipedia hyperlink to build this dictionary. If a redirect page ponit to an entry in wikipedia, the redirect title is added as a surface form to the entry. If the entry appear in one disambiguation page, then the disambiguation title after removing content in parenthesis is added as a surface form to this entry. If any wikipedia inner hyperlink point to an entry, the text in the hyperlink is added as a surface form to this entry.

Furthermore, we count the percentage of each surface form a link and the frequency of each surface form point to different entry. This can be used to compute the commonness (Milne and Witten, 2008) of an surface form explained by an entry. we also store the inner and outer link of each entry page for fast computation of the relatedness (Milne and Witten, 2008) between two entry pages.

2.2 Baseline method

2.2.1 candidate generation

Given a query entity name with its surrounding context, we first generate candidates by dictionary lookup if the query string matches a redirect page title, a disambiguation page title, or hyperlink surface form. In this step we also resolve redirect to their actual target, this way we can merge the surface-entity count of duplicate entities.

2.2.2 candidate selection

In the baseline method, we rank candidate by how often the surface form link to a particular entry. As there is a large proportion of mentions point to the most frequent entity in TAC dataset, this forms a natural baseline.

2.3 Collective method

Following (Han et al., 2011), we utilize the context of the query to incorporate inter-entity relation. Entity disambiguation based on context bag of word similarity often suffer from sparsity problem. Han et al. (2011) use the context entities to disambiguate the current entity, and collectively disambiguate all the entities in the document.

In their approach, context information is represented by local mention to entity similarity and entity to entity relation. Mention to entity similarity is computed using a consine similarity over bag of words of the wikipedia article and the query document. Entity to entity relation is modeled using the relatedness measure of (Milne and Witten, 2008). A referent graph is constructed using this two information. Information can transmit from mention node to entity node and between entity node using a pagerank-like method , where the initial evidence is the importance of mention node. The iterative result is the rank of lists of entities with respect to a list of surface forms.

We implement this method but with a few difference. First we compute the local context similarity by compute the wordnet Jiang-Conrath Similarity between any pair of context words and entry page words. We sum the score as the final local context similarity between the mention and the entity.

Then we compute the inter-entity between any pair of entities generated by different mentions. We use inner link plus outer link as the total links of this page and compute entity relatedness proposed in (Milne and Witten, 2008).

2.4 Evaluation result

Our evaluation result shows that the collective method does no better than the baseline. We believe that this is largely due to the local context similarity introduces too much noise based on our observation. Maybe in this case simple use of bag of words achieves better result. The link measure is also not very effective .

Run	F-1
Highest F1 (no Web)	0.846
Median F1 (no Web)	0.716
ICL_KBP1(baseline)	0.683
ICL_KBP2(collective)	0.668

3 Slot Filling Implementation

In the second task of slot filling, we use the wikipedia infobox from the knowledge base as a source of supervision.

3.1 Extract entity attribute pairs

First, we obtain lists of entity-attribute pairs by mapping the facts in KB to the attributes specified in the TAC-KBP specification. The mapping from a infobox field to KBP field often have a one-to-many mapping. This step needs some consideration to some of the fields:

- the people's title field often contains the form like "general of U.S. Army", to retrieve more sentence instances for the training step, we remove the modifier part of the field after "of";
- the mapping to people's age field often contains a birth date of that person, so we use regular expression to separate the age and date part of this field and map them to per:age and per:date_of_birth field respectively;
- the mapping to per:country,stateorprovince,city_of_birth,death, org:country,stateorprovince,city_of_headquarters, per:countries,stateorprovinces,cities_of_residence often is a one-to-many mapping and need to separate the country,stateorprovince,city part of the field. We first collect lists of

city,state,province,country,nation,village,town from wikipedia entries whose categories contains a category with head noun matching the above keyword. The head noun is derived from the Stanford parser.

3.2 Mapping back entity attribute pairs to sentences

In this step, we collect sentences that contains entity attribute pair from previous step. We group the (relation, entity, attribute) tuples by entity, and query the knowledge base and the source collection with entity name. Then we filter out sentences without any attribute occurence.

This step may affect the recall of extracted sentences without coreference resolution, because the sentences with both a full person/organization name and its attribute is much fewer. Coreference resolution will further improve recall, but due to time limit we do not use more sophisticated method. In this step we obtain totally over 1.2 million sentences for person attributes and over 1.7 million sentences for organiztion attributes.

3.3 Extract patterns using training set

We follow the pattern matching approach of (Chen et al., 2010) by first extract pattern from half of the training sentences, then use the other half to evaluate the effectiveness of the patterns relative to each slot.

We use dependency parsing (Klein and Manning, 2003) for each sentence obtained from step 2, and store them for further processing. This is the major difference from (Chen et al., 2010) approach. In their approach patterns are mainly regular expressions of entity types and words. We believe this can achieve high precision but not very high recall. So we use dependency path patterns.

3.4 Evaluate patterns using development set

We evaluate the pattern from step 3 using patterns from development set. The pattern is a shortest dependency path between the query(entity) and the answer(attribute). Given a pattern relative to a slot, the pattern is ranked by precision $conf(p) = \frac{p.positive}{p.positive+p.negative}$, where correct extraction is estimated using pattern of development set sentences.

3.5 Slot extraction using dependency patterns

Due to long time of dependency parsing of millions of sentences, we could not finish the previous part before submission deadline. so we will return to this in the future.

4 Conclusion

In this paper we describe the methods we use in TAC 2011 knowledge base population track. In entity linking our method does not achieve satisfactory result. We will tuing the parameter and representation of the model in future to see if it can achieve better result.

5 Acknowledgement

This research is supported by National Natural Science Foundation of Chinese (No.60973053,91024009) and Research Fund for the Doctoral Program of Higher Education of China (No.20090001110047).

References

Z. Chen, S. Tamang, A. Lee, X. Li, W.P. Lin, M. Snover, J. Artiles, M. Passantino, and H. Ji. 2010. Cuny-blender tac-kbp2010.

Silviu Cucerzan. 2007. Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 708–716, Prague, Czech Republic, June. Association for Computational Linguistics.

X. Han, L. Sun, and J. Zhao. 2011. Collective entity linking in web text: a graph-based method. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information*, pages 765–774. ACM.

H. Ji and R. Grishman. 2011. Knowledge base population: successful approaches and challenges. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1148–1158. Association for Computational Linguistics.

D. Klein and C.D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 423–430. Association for Computational Linguistics.

David Milne and Ian H. Witten. 2008. Learning to link with wikipedia. In *CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management*, pages 509–518, New York, NY, USA. ACM.