Combining MIML and Distant Supervision for KBP Slot Filling

Jinjian Zhang, Huachun Ma, Siliang Tang*, and Fei Wu
College of Computer Science and Technology,
Zhejiang University
Hangzhou, Zhejiang, China
{jinjianzhang, mahuachun, siliang, wufei}@zju.edu.cn

Abstract

This paper describes the DCD slot filling system for the TAC Cold Start evaluations 2015 Task. The ZJU_DCD_SF 2015 slot filling system mainly uses several classical methods to obtain the slot filler. For Named Entity Recognize and coreference, we apply Stanford Core NLP system. For generating training data, we use the distant supervision, which is a very popular way among the slot-filling task. For classification in sentence level, our system uses the Multi Instances Multi Label method, which is pretty suitable for the nature of the training data generated by distant supervision.

1 Introduction

In this paper, we describe the ZJU_DCD_SF system for TAC KBP 2015 Cold Start Slot Filling (SF) task, which is organized by NIST. This year is our first time to participate this competition.

We used a combination of distant supervision (Mintz et al., 2009) and Multi Instances Multi Labels (Surdeanu M et al., 2012) structured prediction.

This paper is organized as follows: First, an overview of our team's slot filling system (Section 2). Second, the technical details of our distant supervision method. Finally, the performance of the system in the shared task is presented.

2 System Overview

Our slot filling system is a combination of distant supervision and Multi Instance Multi Labels. Slot filling task aims to obtain the information about entities like person, organization or geometry political entities from unstructured text data like news or web forums. There are many challenges in the task like alias of entity, information retrieval, coreference resolution, query expansion, training data generation, relation classification and slot filler inference.

Our slot filling system tries to alleviate and address these problems. In order to get the slot filler about a person, organization or geo-political entity, the following steps need to be performed:

1. Preprocessing the documents
2. Expansion of query
3. Retrieval of documents
4. Retrieval of sentences
5. Mapping relations
6. Relation classification
7. Searching provenances

3 Extraction of candidates

In our slot filling system, we need to extract the candidates, which contain the slot filler information first.

3.1 Preprocessing Documents

All the source documents used by our system have been tokenized. We use Stanford Core NLP (Manning C D et al., 2014) to do this job.

3.2 Expanding Queries

The relation classification model needs many candidates. We need expand queries to address this problem. We add alias for every entity. The aliases were extracted by Freebase dataset (www.freebase.com).

3.3 Searching Candidates

Our system searching the candidates by whoosh, an open source, fast and pure python search engine

* corresponding author

library. Whoosh has following advantages (Matt Chaput, et al.):

- Whoosh is fast, but uses only pure Python, so it will run anywhere Python runs, without requiring a compiler.
- Whoosh's ranking function can be easily customized.
- Whoosh creates very small indexes compared to many other search libraries.
- All indexed text in Whoosh must be Unicode.
- Whoosh lets you store arbitrary Python objects with indexed documents.

4 Features

The representation of candidates mainly based on lexical features, syntactic features and position features. This representation mainly comes from Mintz's way (Mintz et al., 2009).

4.1 Lexical Features

Lexical Features describe specific words between and surrounding the two entities in the candidates extracted by whoosh. Mintz's lexical features include followings:

- The sequence of word between the two entities
- The part-of-speech tags of these words
- A flag indicating which entity came first in the sentence
- A window of *k* words to the left of Entity *1* and their
- part-of-speech tags
- A window of *k* words to the right of Entity 2 and their

Our lexical features consist of the conjunction of all above components.

4.2 Syntactic Features

A dependency parse consists of a set of words and chunks (e.g. ‘Edwin Hubble’, ‘Missouri’, ‘born’), linked by directional dependencies. For each sentence, we extract a dependency path between each pair of entities. A dependency path consists of a series of dependencies, directions and words/ chunks representing a traversal of the parse. Part-of-speech tags are not included in the dependency path. They consist of the conjunction of:

- A dependency path between the two entities
- For each entity, one ‘window’ node that is not part of the dependency path

As for the implementation, we use the Stanford Core NLP system (Manning C D et al., 2014).

5 Relation Classification

Given the candidates of slot filler and sentences contain the entity and slot filler information, our systems applied our Multi Instances Multi Labels model to label the candidates and get the relation. This work is similar with Mihai's way (Surdeanu M et al., 2012).

Our model assumes that each relation mention involving an entity pair has exactly one label, but allows the pair to exhibit multiple labels across different mentions. Since we do not know the actual relation label of a mention in the distantly supervised setting, we model it using a latent variable *z* that can take one of the *k* pre-specified relation labels as well as an additional NIL label, if no relation is expressed by the corresponding mention.

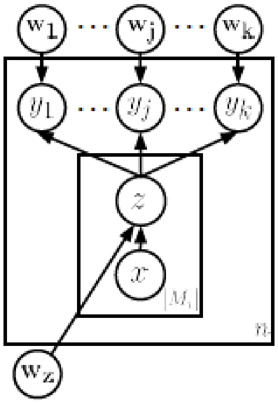

Figure 1. MIML model

We model the multiple relation labels an entity pair and we use a multi-label classifier that takes as input the latent relation types of the all the mentions involving that pair. The two-layer hierarchical model is shown graphically in Figure 1, and is described more formally below. The model includes one multi-class classifier (for *z*) and a set of binary classifiers (for each y_j). The *z* classifier assigns latent labels from *L* to individual relation mentions or NIL if no relation is expressed by the mention. Each y_j classifier decides if relation *j* holds for the given entity, using the mention-level classifications as input.

6 Slot Filling results

6.1 Additional Data

As training data, we use the data from LDC. Additionally, we use the Freebase dataset (www.freebase.com) to get the aliases of entity and slot filler.

6.2 Submissions

We have submitted two submissions for the TAC KBP Cold Start slot-filling track.

- ZJU_DCD_SF1 Features of this submit are lexical and syntactic features.
- ZJU_DCD_SF2 Features of this submit are word2vec and position features.

Experimental results of these systems are shown in Table 1.

	Precision	Recall	F1
ZJU_DCD_SF1	0.0773	0.0277	0.0408
ZJU_DCD_SF2	0.0781	0.0204	0.0323

Table 1. Results

7 Conclusion

In this paper, we presented an overview of the ZJU_DCD_SF system for the KBP 2015 English Cold Start Slot Filling (SF) task. The system uses a combination of distant supervision and multi instances multi labels. In the future work, we would like to use some neural networks ways like CNN and RNN.

Acknowledgments

This work was supported in part by the China Knowledge Centre for Engineering Sciences and Technology (CKCEST), the NSFC (No. 61402401), and the Zhejiang Provincial NSFC (No. LQ14F010004).
We would like to acknowledge Hongliang Dai for his help in developing this system and writing.

References

Surdeanu M, Tibshirani J, Nallapati R, et al. Multi-instance multi-label learning for relation extraction[C]//Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Association for Computational Linguistics, 2012: 455-465.

Riedel S, Yao L, McCallum A. Modeling relations and their mentions without labeled text[C]//Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer Berlin Heidelberg, 2010: 148-163.

www.freebase.com

Mintz M, Bills S, Snow R, et al. Distant supervision for relation extraction without labeled data[C]//Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2. Association for Computational Linguistics, 2009: 1003-1011.

Manning C D, Surdeanu M, Bauer J, et al. The Stanford CoreNLP Natural Language Processing Toolkit[C]//ACL (System Demonstrations). 2014: 55-60.

Angeli G, Tibshirani J, Wu J, et al. Combining Distant and Partial Supervision for Relation Extraction[C]//EMNLP. 2014: 1556-1567.

Matt Chaput, et al. http://bitbucket.org/mchaput/whoosh

Min B, Grishman R. Challenges in the Knowledge Base Population Slot Filling Task[C]//LREC. 2012: 1137-1142.

Angeli G, Gupta S, Jose M, et al. Stanford's 2014 slot filling systems[J]. TAC KBP, 2014, 695.

Hoffmann R, Zhang C, Ling X, et al. Knowledge-based weak supervision for information extraction of overlapping relations[C]//Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. Association for Computational Linguistics, 2011: 541-550.