ICT: System Description for CoNLL-2012

Hao Xiong and Qun Liu
Key Lab. of Intelligent Information Processing
Institute of Computing Technology
Chinese Academy of Sciences
P.O. Box 2704, Beijing 100190, China
{xionghao, liuqun}@ict.ac.cn

Abstract

In this paper, we present our system description for the CoNLL-2012 coreference resolution task on English, Chinese and Arabic. We investigate a projection-based model in which we first translate Chinese and Arabic into English, run a publicly available coreference system, and then use a new projection algorithm to map the coreferring entities back from English into mention candidates detected in the Chinese and Arabic source. We compare to a baseline that just runs the English coreference system on the supplied parses for Chinese and Arabic. Because our method does not beat the baseline system on the development set, we submit outputs generated by the baseline system as our final submission.

1 Introduction

Modeling multilingual unrestricted coreference in the OntoNotes data is the shared task for CoNLL-2012. This is an extension of the CoNLL-2011 shared task and would involve automatic anaphoric mention detection and coreference resolution across three languages – English, Chinese and Arabic – using OntoNotes v5.0 corpus, given predicted information on the syntax, proposition, word sense and named entity layers. Automatic identification of coreferring entities and events in text has been an uphill battle for several decades, partly because it can require world knowledge which is not well-defined and partly owing to the lack of substantial annotated data.

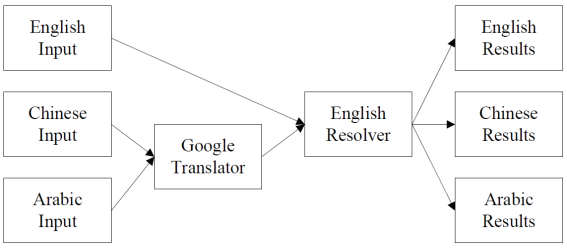

Figure 1: The overall process of our system, where we use Google Translator to translate Chinese and Arabic into English.

For more details, readers can refer to (Pradhan et al., 2012).

Before this year's task, researchers proposed two typical novel methods to address the problem of natural language processing across multiple languages: projection and joint learning (Rahman and Ng, 2012). Specific to this year's coreference resolution task, for projection based method, we could first develop a strong resolver or utilize a publicly available system on English, and translate other languages into English, eventually, we could project the coreferring entities resolved on English back into other language sides. Generally, a projection method is easier to develop since it doesn't need sentence alignment across multiple languages. Thus, in this year's task, we investigate a translation based model to resolve coreference on English, Chinese and Arabic. The whole process is illustrated in figure 1, in which we first use Google Translator to translate Chinese and Arabic into English, and we then employ a strong English coreference resolver to generate coreferring entities, after mapping entities from English into

Proceedings of the Joint Conference on EMNLP and CoNLL: Shared Task, pages 71–75,
Jeju Island, Korea, July 13, 2012.

Chinese and Arabic mention candidates, we could obtain coreferring entities for these languages.

Intuitively, the performance of coreference resolver on English should perform better than that on Chinese and Arabic since we have substantial corpus for English and coreference resolution on English is well studied compared to another two languages. Thus we could imagine that projecting the results from English into Chinese and Arabic should still beats the baseline system using monolingual resolution method. However, in our experiments, we obtain negative results on developing set that means our projection based model perform worse than the baseline system. According to our experimental results on developing set, finally, we submit results of baseline system in order to obtain better ranking.

The rest of this paper is organized as follows, in section 2, we will introduce our method in details, and section 3 is our experimental results, we draw conclusion in section 4.

2 Projection based Model

As the last section mentioned, we propose to use a projection based model to resolve coreference on multiple languages. The primary procedures of our method could be divided into three steps: first step is translation, where Google Translator is employed to translate Chinese and Arabic into English, second is coreference resolution for English, last is the projection of coreferring entities. Since the first step is clear that we extract sentences from Chinese and Arabic documents and translate them into English using Google Translator, hence in this section we will mainly describe the configuration of our English resolver and details of projection method.

2.1 English Resolver

In last year's evaluation task, the Standford Natural Language Processing Group ranked the first position and they also open their toolkit for research community, namely Standford CoreNLP (Lee et al., 2011) [1], better yet, their toolkit is optimized for CoNLL task. Thus we could use their toolkit as our English resolver and concentrate on bettering the projection of coreferring entities.

[1] http://nlp.stanford.edu/software/corenlp.shtml

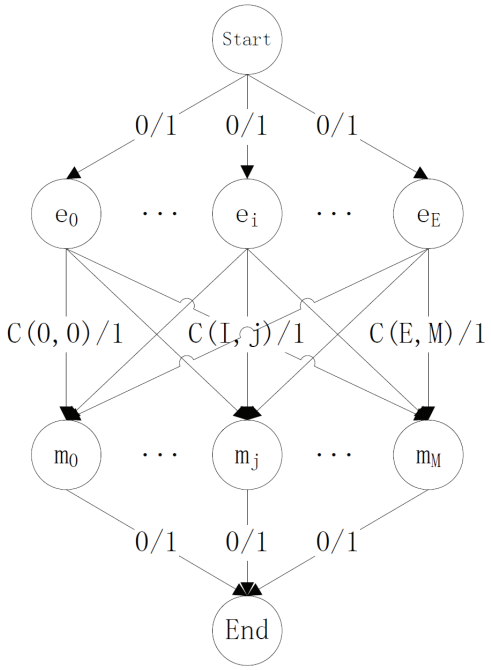

Figure 2: A minimum cost and maximum flow structure is used to solve the problem that mapping coreferring entities into each mention candidates with highest probability.

We use the basic running script that is "java -cp joda-time.jar:stanford-corenlp.jar:stanford-corenlp-models.jar:xom.jar -Xmx3g edu.stanford.nlp.pipeline.StanfordCoreNLP -filelist filelist.txt" to resolve the resolution, where "filelist" involves all documents need to be performed coreference resolution.

2.2 Projection of Coreferring Entities

After generating coreferring entities on English, the key step of our system is how to map them into mention candidates detected on Chinese and Arabic. For instance, assuming we translate Chinese documents into English and obtain coreferring entities e_1, e_2, e_i,.., e_E on translated English documents through aforementioned step, meanwhile, we consider all noun phrases(NP) in original Chinese documents and generate mention candidates m_1, m_2, m_j,.., m_M. Therefore, our task is to map each e_i into one mention candidate m_j with highest probability, and it can be obtained by the max-

Algorithm 1 Algorithm for computing similarity between two phrases in different languages.

1: **Input**: $w_{e_1}, .., w_{e_n}, w_{c_1}, .., w_{c_m}$, Phrase Table PT
2: $s[n] = [0, -\inf, .., -\inf]$
3: **for** $i \leftarrow 1..n$ **do**
4: **for** $j \leftarrow 0..10$ **do**
5: $s[i+j] = \max(s[i+j], s[i-1] + p(i, i+j))$
6: **Output:** $s[n]$ **V**

imization of the following formula,

$$\hat{P} = \sum_{e_i \in E, m_j \in M} \{a(i,j)b(j,i)p(i,j)\} \quad (1)$$

with constrains $\sum_{i,j}\{a(i,j)\} = 1$ and $\sum_{i,j}\{b(j,i)\} = 1$, where $p(i,j)$ is the probability of e_i mapping into m_j and $a(i,j)$ as well as $b(i,j)$ are integers guaranteeing each coreferring entity map into one mention and each mention has only one entity to be mapped into. To solve this problem, we reduce it as a Cost Flow problem since it is easier to understand and implement compared to other methods such as integer linear programming. Note that the number of mention candidates is theoretically larger than that of coreferring entities, thus this problem couldn't be reduced as the bipartite graph matching problem since it needs equal number of nodes in two parts.

Figure 2 shows the graph structure designed to solve this problem, where the symbols labeled on each edge is a two tuples(Cost,Flow), indicating the cost and flow for each edge. Since object of Cost Flow problem is to minimize the cost while maximizing the flows, thus we compute the $c(i,j)$ as $1 - p(i,j)$ in order to be consistent with the equation 1. To satisfy two constraints aforementioned, we set up two dummy nodes "Start" and "End", and connect "Start" to each entity e_i with cost 0 and flow 1 ensuring each entity is available to map one mention. We also link each mention candidate m_j to node "End" with the same value ensuring each mention could be mapped into by only one entity. Clearly, there is an edge with tuple $(1-p(i,j), 1)$ between each entity end mention indicating that each entity could map into any mention while with different probabilities. Thus, solving this Cost-Flow problem is equal to maximizing the equation 1 with two constraints. Since Cost-Flow problem is well studied, thus some algorithm can solve this problem in polynomial time (Ahuja et al., 1993). One may argue that we can modify translation decoder to output alignments between Chinese and translated English sentence, unfortunately, Google Translator API doesn't supply these information while its translation quality is obviously better than others for translating documents in OntoNotes, moreover, it is impossible to output alignment for each word since some translation rules used for directing translation include some unaligned words, thus an algorithm to map each entity into each mention is more applicable.

Clearly, another problem is how to compute $p(i,j)$ for each edge between entity and mention candidate. This problem could be casted as how to compute similarity of phrases across multiple languages. Formally, given an English phrases $w_{e_1}, .., w_{e_n}$ and a Chinese phrase $w_{c_1}, .., w_{c_m}$, the problem is how to compute the similar score S between them. Although we could compute lexical, syntactic or semantic similar score to obtain accurate similarity, here for simplicity, we just compute the lexical similarity using the phrase table extracted by a phrased-based machine translation decoder (Koehn et al., 2003). Phrase table is a rich resource that contains probability score for phrase in one language translated into another language, thus we could design a dynamic algorithm shown in Algorithm 1 to compute the similar score. Equation in line 5 is used to reserve highest similar score for its sub-phrases, and $p(i, i+j)$ is the similar score between sub-phrases $w_i, .., w_{i+j}$ and its translation. When we compute the score of the sub-phrases $w_i, .., w_{i+j}$, we literately pick one pt_i from PT and check whether $w_{c_1}, .., w_{c_m}$ involves pt_i's target side, if that we record its score until we obtain a higher score obtained by another pt_j and then update it. For instance, assuming the Chinese input sentence is "全球 第五 个 迪斯尼 乐园 即将 在 这里 向 公众 开放 。", and the Google translation of this sentence is "The world 's fifth Disneyland will soon open to the public . ". Following the aforementioned steps, we utilize English resolver to find a coreferring entity: "The world 's fifth Disneyland", and find two translation rules involving the former English phrase from the

bilingual phrase table: “The world ’s fifth Disneyland => 全球的第五个迪斯尼乐园 (probability=0.6) ” and “The world ’s fifth Disneyland => 全球第五个迪斯尼乐园 (probability=0.4)”. Since the Chinese translation of both rules all contain the noun phrase “全球 第五 个 迪斯尼 乐园” in the original Chinese input, we thus add this noun phrase into the coreferring entities as the English resolve finding with the probability 0.6.

3 Experiments

3.1 English Results

In this section, we will report our experimental results in details. We use Standford CoreNLP toolkit to generate results for English. Table 1 lists the F-score obtained on developing set.

3.2 Chinese and Arabic Results

As last section mentioned, we first translate Chinese and Arabic into English and then use CoreNLP to resolve coreference on English. To obtain high translation quality, we use Google Translator Toolkit [2]. And to compute similarity score, we run Giza++(Och and Ney, 2003) [3], an open source toolkit for word alignment, to perform word alignment. For Chinese, we use 1 million bilingual corpus provided by NIST MT evaluation task to extract phrase table, and for Arabic its size is 2 million. Note that, we extract phrase table from English to Chinese and Arabic with maximum phrase length 10. The reason is that our algorithm check English phrase whose length is less than 10 tokens. To compare our results, we also use CoreNLP to generate results for Chinese and Arabic. Since CoreNLP use some syntactic knowledge to resolving coreference, it can also output coreferring entities for other languages. From table 2 we find that although CoreNLP is not designed for other languages, it still obtain acceptable scores and beat our projection based model. The main reason is that our method is coarse and obtain lower precision for mention detection, while CoreNLP use some manually written rules to detect mention candidates. Another explanation is that projection based model is hard to map some phrases back into original languages, such as “that, it, this”. Moreover, translation quality for some corpus like web corpus is far from perfect, translation errors will surely affect the precision of coreference resolution. Thus, for the final testing set, we run the CoreNLP to generate the results.

3.3 Testing Results

Since CoreNLP beats our system in Chinese and Arabic, thus we run CoreNLP for all three languages. Table 3 lists the final results, and we also give results using golden parse tree for prediction in table 4. From these two tables, we find that for any language, the system using golden parse tree show better performance than the one using predicted system in term of each metric. The reason is that the CoreNLP resolve coreference on parse tree and employ some parse features to corefer. On the other hand, we could also see that the improvement is slight, because parsing errors affect little on finding mention candidates benefiting from high precision on noun phrase prediction. Finally, since we use an open source toolkit to generate results, unfortunately, we have no ranking in this task.

4 Conclusion

In this paper, we present a projection based model for coreference resolution. We first translate Chinese and Arabic into English, and then employ a strong English resolver to generate coreferring entities, after that a projection algorithm is designed to map coreferring entities into mention candidates detected in Chinese and Arabic. However, since our approach is coarse and due to limit time preparing for this task, the output generate by CoreNLP beats our results in three languages, thus we submit results generated by CoreNLP as our final submission.

Acknowledgments

The authors were supported by National Science Foundation of China, Contracts 90920004, and High-Technology R&D Program (863) Project No 2011AA01A207 and 2012BAH39B03. We thank organizers for their generous supplied resources and arduous preparation. We also thank anonymous reviewers for their thoughtful suggestions.

[2] http://www.google.cn/url?source=transpromo&rs=rsmf&q=http://translate.google.com/toolkit

[3] http://code.google.com/p/giza-pp/

	Mention	MUC	BCUB	CEAFE
CoreNLP	73.68%	64.58%	70.60%	46.64

Table 1: Experimental results on developing set(F-score) for English.

	Mention	MUC	BCUB	CEAFE
CoreNLP-Chinese	52.15%	38.16%	60.38%	34.58
Projection-Chinese	48.51%	32.31%	63.77%	24.72
CoreNLP-Arabic	52.97%	27.88%	60.75%	40.52
Projection-Arabic	42.68%	22.39%	62.18%	32.83

Table 2: Experimental results on developing set(F-score) for Chinese and Arabic using CoreNLP and our system.

	Mention	MUC	BCUB	CEAFE
CoreNLP-Chinese	49.82%	37.83%	60.30%	34.93
CoreNLP-Arabic	53.89%	28.31%	61.83%	42.97
CoreNLP-English	73.69%	63.82%	68.52%	45.36

Table 3: Experimental results on testing set(F-score) using predicted parse tree.

	Mention	MUC	BCUB	CEAFE
CoreNLP-Chinese	53.42%	40.60%	60.37%	35.75
CoreNLP-Arabic	55.17%	30.54%	62.36%	43.03
CoreNLP-English	75.58%	66.14%	69.55%	46.54

Table 4: Experimental results on testing set(F-score) using golden parse tree.

References

R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. 1993. Network flows: theory, algorithms, and applications. 1993.

P. Koehn, F.J. Och, and D. Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. Association for Computational Linguistics.

H. Lee, Y. Peirsman, A. Chang, N. Chambers, M. Surdeanu, and D. Jurafsky. 2011. Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 28–34. Association for Computational Linguistics.

F.J. Och and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1):19–51.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Proceedings of the Sixteenth Conference on Computational Natural Language Learning (CoNLL 2012)*, Jeju, Korea.

Altaf Rahman and Vincent Ng. 2012. Translation-based projection for multilingual coreference resolution. In *NAACL 2012*.