Multilingual Dependency Learning: Exploiting Rich Features for Tagging Syntactic and Semantic Dependencies

Hai Zhao(赵海)[†], Wenliang Chen(陈文亮)[‡],
Jun'ichi Kazama[‡], Kiyotaka Uchimoto[‡], and **Kentaro Torisawa[‡]**
[†]Department of Chinese, Translation and Linguistics
City University of Hong Kong
83 Tat Chee Avenue, Kowloon, Hong Kong, China
[‡]Language Infrastructure Group, MASTAR Project
National Institute of Information and Communications Technology
3-5 Hikari-dai, Seika-cho, Soraku-gun, Kyoto, Japan, 619-0289
haizhao@cityu.edu.hk, chenwl@nict.go.jp

Abstract

This paper describes our system about multilingual syntactic and semantic dependency parsing for our participation in the joint task of CoNLL-2009 shared tasks. Our system uses rich features and incorporates various integration technologies. The system is evaluated on in-domain and out-of-domain evaluation data of closed challenge of joint task. For in-domain evaluation, our system ranks the second for the average macro labeled F1 of all seven languages, 82.52% (only about 0.1% worse than the best system), and the first for English with macro labeled F1 87.69%. And for out-of-domain evaluation, our system also achieves the second for average score of all three languages.

1 Introduction

This paper describes the system of National Institute of Information and Communications Technology (NICT) and City University of Hong Kong (CityU) for the joint learning task of CoNLL-2009 shared task (Hajič et al., 2009)[1]. The system is basically a pipeline of syntactic parser and semantic parser. We use a syntactic parser that uses very rich features and integrates graph- and transition-based methods. As for the semantic parser, a group of well selected feature templates are used with n-best syntactic features.

The rest of the paper is organized as follows. The next section presents the technical details of our syntactic dependency parsing. Section 3 describes the details of the semantic dependency parsing. Section 4 shows the evaluation results. Section 5 looks into a few issues concerning our forthcoming work for this shared task, and Section 6 concludes the paper.

2 Syntactic Dependency Parsing

Basically, we build our syntactic dependency parsers based on the MSTParser, a freely available implementation[2], whose details are presented in the paper of McDonald and Pereira (2006). Moreover, we exploit rich features for the parsers. We represent features by following the work of Chen et al. (2008) and Koo et al. (2008) and use features based on dependency relations predicted by transition-based parsers (Nivre and McDonald, 2008). Chen et al. (2008) and Koo et al. (2008) proposed the methods to obtain new features from large-scale unlabeled data. In our system, we perform their methods on training data because the closed challenge does not allow to use unlabeled data. In this paper, we call these new additional features rich features.

2.1 Basic Features

Firstly, we use all the features presented by McDonald et al. (2006), if they are available in data. Then we add new features for the languages having FEAT information (Hajič et al., 2009). FEAT is a set of morphological-features, e.g. more detailed part of speech, number, gender, etc. We try to align different types of morphological-features. For example,

[1]Our thanks give to the following corpus providers, (Taulé et al., 2008; Palmer and Xue, 2009; Hajič et al., 2006; Surdeanu et al., 2008; Burchardt et al., 2006) and (Kawahara et al., 2002).

[2]http://mstparser.sourceforge.net

we can obtain a sequence of gender tags of all words from a head h to its dependent d. Then we represent the features based on the obtained sequences.

Based on the results of development data, we perform non-projective parsing for Czech and German and perform projective parsing for Catalan, Chinese, English, Japanese, and Spanish.

2.2 Features Based on Dependency Pairs

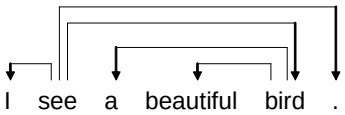

Figure 1: Example dependency graph.

Chen et al. (2008) presented a method of extracting short dependency pairs from large-scale auto-parsed data. Here, we extract all dependency pairs rather than short dependency pairs from training data because we believe that training data are reliable. In a parsed sentence, if two words have dependency relation, we add this word pair into a list named L and count its frequency. We consider the direction. For example, in figure 1, *a* and *bird* have dependency relation in the sentence "I see a beautiful bird.". Then we add word pair "a-bird:HEAD"[3] into list L and accumulate its frequency.

We remove the pairs which occur only once in training data. According to frequency, we then group word pairs into different buckets, with bucket LOW for frequencies 2-7, bucket MID for frequencies 8-14, and bucket HIGH for frequencies $15+$. We set these threshold values by following the setting of Chen et al. (2008). For example, the frequency of pair "a-bird:HEAD" is 5. Then it is grouped into bucket "LOW". We also add a virtual bucket "ZERO" to represent the pairs that are not included in the list. So we have four buckets. "ZERO", "LOW", "MID", and "HIGH" are used as bucket IDs.

Based on the buckets, we represent new features for a head h and its dependent d. We check word pairs surrounding h and d. Table 1 shows the word pairs, where h-word refers to the head word, d-word refers to the dependent word, h-word-1 refers to the word to the left of the head in the sentence, h-word+1 refers to the word to the right of the head, d-word-1 refers to the word to the left of the dependent, and d-word+1 refers the word to the right of the dependent. Then we obtain the bucket IDs of these word pairs from L.

We generate new features consisting of indicator functions for bucket IDs of word pairs. We call these features word-pair-based features. We also generate combined features involving bucket IDs and part-of-speech tags of heads.

h-word, d-word
h-word-1, d-word
h-word+1, d-word
h-word, d-word-1
h-word, d-word+1

Table 1: Word pairs for feature representation

2.3 Features Based on Word Clusters

Koo et al. (2008) presented new features based on word clusters obtained from large-scale unlabeled data and achieved large improvement for English and Czech. Here, word clusters are generated only from the training data for all the languages. We perform word clustering by using the clustering tool[4], which also was used by Koo et al. (2008). The cluster-based features are the same as the ones used by Koo et al. (2008).

2.4 Features Based on Predicted Relations

Nivre and McDonald (2008) presented an integrating method to provide additional information for graph-based and transition-based parsers. Here, we represent features based on dependency relations predicted by transition-based parsers for graph-based parser. Based on the results on development data, we choose the MaltParser for Catalan, Czech, German, and Spanish, and choose another MaxEnt-based parser for Chinese, English, and Japanese.

2.4.1 A Transition-based Parser: MaltParser

For Catalan, Czech, German, and Spanish, we use the MaltParser, a freely available implementa-

[3] HEAD means that *bird* is the head of the pair.

[4] http://www.cs.berkeley.edu/~pliang/software/brown-cluster-1.2.zip

tion[5], whose details are presented in the paper of Nivre (2003). More information about the parser can be available in the paper (Nivre, 2003).

Due to computational cost, we do not select new feature templates for the MaltParser. Following the features settings of Hall et al. (2007), we use their Czech feature file and Catalan feature file. To simply, we apply Czech feature file for German too, and apply Catalan feature file for Spanish.

2.4.2 Another Transition-based Parser: MaxEnt-based Parser

In three highly projective language, Chinese, English and Japanese, we use the maximum entropy syntactic dependency parser as in Zhao and Kit (2008). We still use the similar feature notations of that work. We use the same greedy feature selection of Zhao et al. (2009) to determine an optimal feature template set for each language. Full feature sets for the three languages can be found at website, http://bcmi.sjtu.edu.cn/~zhaohai.

2.4.3 Feature Representation

For training data, we use 2-way jackknifing to generate predicted dependency parsing trees by two transition-based parsers. Following the features of Nivre and McDonald (2008), we define features for a head h and its dependent d with label l as shown in table 2, where G_{Tran} refers to dependency parsing trees generated by the MaltParser or MaxEnt-base Parser and $*$ refers to any label. All features are conjoined with the part-of-speech tags of the words involved in the dependency.

Is $(h, d, *)$ in G_{Tran}?
Is (h, d, l) in G_{Tran}?
Is $(h, d, *)$ not in G_{Tran}?
Is (h, d, l) not in G_{Tran}?

Table 2: Features set based on predicted labels

3 n-best Syntactic Features for Semantic Dependency Parsing

Due to the limited computational resource that we have, we used the the similar learning framework as our participant in semantic-only task (Zhao et al., 2009). Namely, three languages, a single maximum entropy model is used for all identification and classification tasks of predicate senses or argument labels in four languages, Catalan, Czech, Japanese, or Spanish. For the rest three languages, an individual sense classifier still using maximum entropy is additionally used to output the predicate sense previously. More details about argument candidate pruning strategies and feature template set selection are described in Zhao et al. (2009).

	Normal	n-best	Matched
Ca	53	54	50
Ch	75	65	55
En	73	70	63

Table 3: Feature template sets:n-best vs. non-n-best

The same feature template sets as the semantic-only task are used for three languages, Czech, German and Japanese. For the rest four languages, we further use n-best syntactic features to strengthen semantic dependency parsing upon those automatically discovered feature template sets. However, we cannot obtain an obvious performance improvement in Spanish by using n-best syntactic features. Therefore, only Catalan, Chinese and English semantic parsing adopted these types of features at last.

Our work about n-best syntactic features still starts from the feature template set that is originally selected for the semantic-only task. The original feature template set is hereafter referred to 'the normal' or 'non-n-best'. In practice, only 2nd-best syntactic outputs are actually adopted by our system for the joint task.

To generate helpful feature templates from the 2nd-best syntactic tree, we simply let all feature templates in the normal feature set that are based on the 1st-best syntactic tree now turn to the 2nd-best one. Using the same notations for feature template representation as in Zhao et al. (2009), we take an example to show how the original n-best features are produced. Assuming $a.children.dprel.bag$ is one of syntactic feature templates in the normal set, this feature means that all syntactic children of the argument candidate (a) are chosen, and their dependant labels are collected, the duplicated labels are removed and then sorted, finally all these strings are concatenated as a feature. The cor-

[5]http://w3.msi.vxu.se/~nivre/research/MaltParser.html

Language	Features
Catalan	*p:2.lm.dprel*
_	*a.lemma + a:2.h.form*
_	*a.lemma + a:2.pphead.form*
_	*(a:2:p:2|dpPath.dprel.seq) + p.FEAT1*
Chinese	*a:2.h.pos*
_	*a:2.children.pos.seq + p:2.children.pos.seq*
_	*a:2:p:2|dpPath.dprel.bag*
_	*a:2:p:2|dpPathPred.form.seq*
_	*a:2:p:2|dpPath.pos.bag*
_	*(a:2:p:2|dpTreeRelation) + p.pos*
_	*(a:2:p:2|dpPath.dprel.seq) + a.pos*
English	*a:2:p:2|dpPathPred.lemma.bag*
_	*a:2:p:2|dpPathPred.pos.bag*
_	*a:2:p:2|dpTreeRelation*
_	*a:2:p:2|dpPath.dprel.seq*
_	*a:2:p:2|dpPathPred.dprel.seq*
_	*a.lemma + a:2.dprel + a:2.h.lemma*
_	*(a:2:p:2|dpTreeRelation) + p.pos*

Table 4: Features for n-best syntactic tree

responding 2nd-best syntactic feature will be a : $2.children.dprel.bag$. As all operations to generate the feature for $a.children.dprel.bag$ is within the 1st-best syntactic tree, while those for a : $2.children.dprel.bag$ is within the 2nd-best one. As all these 2nd-best syntactic features are generated, we use the same greedy feature selection procedure as in Zhao et al. (2009) to determine the best fit feature template set according to the evaluation results in the development set.

For Catalan, Chinese and English, three optimal n-best feature sets are obtained, respectively. Though dozens of n-best features are initially generated for selection, only few of them survive after the greedy selection. A feature number statistics is in Table 3, and those additionally selected n-best features for three languages are in Table 4. Full feature lists and their explanation for all languages will be available at the website, http://bcmi.sjtu.edu.cn/~zhaohai.

4 Evaluation Results

Two tracks (closed and open challenges) are provided for joint task of CoNLL2009 shared task. We participated in the closed challenge and evaluated our system on the in-domain and out-of-domain evaluation data.

	avg.	Cz	En	Gr
Syntactic (LAS)	77.96	75.58	82.38	75.93
Semantic (Labeled F1)	75.01	82.66	74.58	67.78
Joint (Macro F1)	76.51	79.12	**78.51**	71.89

Table 6: The official results of our submission for out-of-domain task(%)

	Test		Dev	
	Basic	ALL	Basic	ALL
Catalan	82.91	85.88	83.15	85.98
Chinese	74.28	75.67	73.36	75.64
Czech	77.21	79.70	77.91	80.22
English	88.63	89.19	86.35	87.40
German	84.61	86.24	83.99	85.44
Japanese	92.31	92.32	92.01	92.85
Spanish	83.59	86.29	83.73	86.22
Average	83.32	85.04 (+1.72)	82.92	84.82 (+1.90)

Table 7: The effect of rich features for syntactic dependency parsing

4.1 Official Results

The official results for the joint task are in Table 5, and the out-of-domain task in Table 6, where numbers in bold stand for the best performances for the specific language. For out-of-domain (OOD) evaluation, we did not perform any domain adaptation. For both in-domain and out-of-domain evaluation, our system achieved the second best performance for the average Macro F1 scores of all the languages. And our system provided the first best performance for the average Semantic Labeled F1 score and the forth for the average Labeled Syntactic Accuracy score for in-domain evaluation.

4.2 Further results

At first, we check the effect of rich features for syntactic dependency parsing. Table 7 shows the comparative results of basic features and all features on test and development data, where "Basic" refers to the system with basic features and "ALL" refers to the system with basic features plus rich features. We found that the additional features provided improvement of 1.72% for test data and 1.90% for development data.

Then we investigate the effect of different training data size for semantic parsing. The learning

	average	Catalan	Chinese	Czech	English	German	Japanese	Spanish
Syntactic (LAS)	85.04	85.88	75.67	79.70	89.19	86.24	92.32	86.29
Semantic (Labeled F1)	79.96	80.10	76.77	82.04	86.15	76.19	78.17	80.29
Joint (Macro F1)	82.52	**83.01**	76.23	80.87	**87.69**	81.22	85.28	**83.31**

Table 5: The official results of our joint submission (%)

Data	Czech	Chinese		English	
		normal	n-best	normal	n-best
25%	80.71	75.12	75.24	82.02	82.06
50%	81.52	76.50	76.59	83.52	83.42
75%	81.90	76.92	77.01	84.21	84.30
100%	82.24	77.35	77.34	84.73	84.80

Table 8: The performance in development set (semantic labeled F1) vs. training corpus size

curves are drawn for Czech, Chinese and English. We use 25%, 50% and 75% training corpus, respectively. The results in development sets are given in Table 8. Note that in this table the differences between normal and n-best feature template sets are also given for Chinese and English. The results in the table show that n-best features help improve Chinese semantic parsing as the training corpus is smaller, while it works for English as the training corpus is larger.

5 Discussion

This work shows our further endeavor in syntactic and semantic dependency parsing, based on our previous work (Chen et al., 2008; Zhao and Kit, 2008).

Chen et al. (Chen et al., 2008) and Koo et al. (Koo et al., 2008) used large-scale unlabeled data to improve syntactic dependency parsing performance. Here, we just performed their method on training data. From the results, we found that the new features provided better performance. In future work, we can try these methods on large-scale unlabeled data for other languages besides Chinese and English.

In Zhao and Kit (2008), we addressed that semantic parsing should benefit from cross-validated training corpus and n-best syntactic output. These two issues have been implemented during this shared task. Though existing work show that re-ranking for semantic-only or syntactic-semantic joint tasks may bring higher performance, the limited computational resources does not permit us to do this for multiple languages.

To analyze the advantage and the weakness of our system, the ranks for every languages of our system's outputs are given in Table 9, and the performance differences between our system and the best one in Table 10[6]. The comparisons in these two tables indicate that our system is slightly weaker in the syntactic parsing part, this may be due to the reason that syntactic parsing in our system does not benefit from semantic parsing as the other joint learning systems. However, considering that the semantic parsing in our system simply follows the output of the syntactic parsing and the semantic part of our system still ranks the first for the average score, the semantic part of our system does output robust and stable results. It is worth noting that semantic labeled F1 in Czech given by our system is 4.47% worse than the best one. This forby gap in this language further indicates the advantage of our system in the other six languages and some latent bugs or learning framework misuse in Czech semantic parsing.

6 Conclusion

We describe the system that uses rich features and incorporates integrating technology for joint learning task of syntactic and semantic dependency parsing in multiple languages. The evaluation results show that our system is good at both syntactic and semantic parsing, which suggests that a feature-oriented method is effective in multiple language processing.

References

Aljoscha Burchardt, Katrin Erk, Anette Frank, Andrea Kowalski, Sebastian Padó, and Manfred Pinkal. 2006.

[6]The difference for Chinese in the latter table is actually computed between ours and the second best system.

	average	Catalan	Chinese	Czech	English	German	Japanese	Spanish
Syntactic (LAS)	4	4	4	4	2	3	3	4
Semantic (Labeled F1)	1	1	3	4	1	2	2	1
Joint (Macro F1)	2	1	3	4	1	3	2	1

Table 9: Our system's rank within the joint task according to three main measures

	average	Catalan	Chinese	Czech	English	German	Japanese	Spanish
Syntactic (LAS)	0.73	1.98	0.84	0.68	0.69	1.24	0.25	1.35
Semantic (Labeled F1)	-	-	0.38	4.47	-	2.42	0.09	-
Joint (Macro F1)	0.12	-	0.15	2.40	-	1.22	0.37	-

Table 10: The performance differences between our system and the best one within the joint task according to three main measures

The SALSA corpus: a German corpus resource for lexical semantics. In *Proceedings of LREC-2006*, Genoa, Italy.

Wenliang Chen, Daisuke Kawahara, Kiyotaka Uchimoto, Yujie Zhang, and Hitoshi Isahara. 2008. Dependency parsing with short dependency relations in unlabeled data. In *Proceedings of IJCNLP-2008*, Hyderabad, India, January 8-10.

Jan Hajič, Jarmila Panevová, Eva Hajičová, Petr Sgall, Petr Pajas, Jan Štěpánek, Jiří Havelka, Marie Mikulová, and Zdeněk Žabokrtský. 2006. Prague Dependency Treebank 2.0.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, Pavel Straňák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. 2009. The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of CoNLL-2009*, Boulder, Colorado, USA.

Johan Hall, Jens Nilsson, Joakim Nivre, Gülsen Eryiğit, Beáta Megyesi, Mattias Nilsson, and Markus Saers. 2007. Single malt or blended? a study in multilingual parser optimization. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 933–939, Prague, Czech, June.

Daisuke Kawahara, Sadao Kurohashi, and Kôiti Hasida. 2002. Construction of a Japanese relevance-tagged corpus. In *Proceedings of LREC-2002*, pages 2008–2013, Las Palmas, Canary Islands.

Terry Koo, Xavier Carreras, and Michael Collins. 2008. Simple semi-supervised dependency parsing. In *Proceedings of ACL-08: HLT*, pages 595–603, Columbus, Ohio, USA, June.

Ryan McDonald and Fernando Pereira. 2006. Online learning of approximate dependency parsing algorithms. In *Proceedings of EACL-2006*, pages 81–88, Trento, Italy, April.

Ryan McDonald, Kevin Lerman, and Fernando Pereira. 2006. Multilingual dependency analysis with a two-stage discriminative parser. In *Proceedings of CoNLL-X*, New York City, June.

Joakim Nivre and Ryan McDonald. 2008. Integrating graph-based and transition-based dependency parsers. In *Proceedings of ACL-08: HLT*, pages 950–958, Columbus, Ohio, June.

Joakim Nivre. 2003. An efficient algorithm for projective dependency parsing. In *Proceedings of the 8th International Workshop on Parsing Technologies (IWPT 03)*, pages 149–160, Nancy, France, April 23-25.

Martha Palmer and Nianwen Xue. 2009. Adding semantic roles to the Chinese Treebank. *Natural Language Engineering*, 15(1):143–172.

Mihai Surdeanu, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. 2008. The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of the CoNLL-2008*.

Mariona Taulé, Maria Antònia Martí, and Marta Recasens. 2008. AnCora: Multilevel Annotated Corpora for Catalan and Spanish. In *Proceedings of the LREC-2008*, Marrakesh, Morroco.

Hai Zhao and Chunyu Kit. 2008. Parsing syntactic and semantic dependencies with two single-stage maximum entropy models. In *Proceedings of CoNLL-2008*, pages 203–207, Manchester, UK, August 16-17.

Hai Zhao, Wenliang Chen, Chunyu Kit, and Guodong Zhou. 2009. Multilingual dependency learning: A huge feature engineering method to semantic dependency parsing. In *Proceedings of CoNLL-2009*, Boulder, Colorado, USA.