PKU_HIT: An Event Detection System Based on Instances Expansion and Rich Syntactic Features

Shiqi Li[1], Pengyuan Liu[2], Tiejun Zhao[1], Qin Lu[3] and Hanjing Li[1]
[1]School of Computer Science and Technology,
Harbin Institute of Technology, Harbin 150001, China
{sqli,tjzhao,hjlee}@mtlab.hit.edu.cn
[2]Institute of Computational Linguistics,
Peking University, Beijing 100871, China
liupengyuan@pku.edu.cn
[3]Department of Computing,
The Hong Kong Polytechnic University, Hong Kong, China
csluqin@comp.polyu.edu.hk

Abstract

This paper describes the PKU_HIT system on event detection in the SemEval-2010 Task. We construct three modules for the three sub-tasks of this evaluation. For target verb WSD, we build a Naïve Bayesian classifier which uses additional training instances expanded from an untagged Chinese corpus automatically. For sentence SRL and event detection, we use a feature-based machine learning method which makes combined use of both constituent-based and dependency-based features. Experimental results show that the Macro Accuracy of the WSD module reaches 83.81% and F-Score of the SRL module is 55.71%.

1 Introduction

In this paper, we describe the system submitted to the SemEval-2010 Task 11 on event detection in Chinese news sentences (Zhou, 2010). The objective of the task is to detect and analyze basic event contents in Chinese news sentences, similar to the frame semantic structure extraction task in SemEval-2007. However, this task is a more complex as it involves three interrelated subtasks: (1) target verb word sense disambiguation (WSD), (2) sentence semantic role labeling (SRL) and (3) event detection (ED).

Therefore, the architecture of the system that we develop for the task consists of three modules: WSD, SRL and ED. First, the WSD module is to recognize key verbs or verb phrases which describe the basic event in a sentence, and then select an appropriate situation description formula for the recognized key verbs (or verb phrases); Then, the SRL module anchors the arguments to suitable constituents in the sentence, and then label each argument with three functional tags, namely constituent type tag, semantic role tags and event role tag. Finally, in the ED module, complete situation description of the sentence can be achieved by combining the results of the WSD module and the SRL module.

For the WSD module, we consider the subtask as a general WSD problem. First of all, we automatically extract many instances from an untagged Chinese corpus using a heuristic rule inspired by Yarowsky (1993). Then we train a Naïve Bayesian (NB) classifier based on both the extracted instances and the official training data. We then use the NB classifier to predict situation the description formula and natural explanation of each target verb in testing data.

For the SRL module, we use a rich syntactic feature-based learning method. As the state-of-the-art method in the field of SRL, feature-based method represents a predicate-argument structure (PAS) by a flat vector using a set of linguistic features. Then PAS can be directly classified by machine learning algorithms based on the corresponding vectors. In feature-based SRL, the

Proceedings of the 5th International Workshop on Semantic Evaluation, ACL 2010, pages 304–307,
Uppsala, Sweden, 15-16 July 2010. ©2010 Association for Computational Linguistics

significance of syntactic information in SRL was proven by (Punyakanok et al., 2005). In our method, we exploit a rich set of syntactic features from two syntactic views: constituent and dependency. As the two syntactic views focus on different syntactic elements, constituent-based features and dependency-based features can complement each other in SRL to some extent. Finally, the ED module can be readily implemented by combining the SRL and the WSD result using some simply rules.

2 System Description

2.1 Target Verb WSD

The WSD module is based on a simple heuristic rule by which we can extract sense-labeled instances automatically. The heuristic rule assumes that one sense per 3-gram which is proposed by us initially through investigating a Chinese sense-tagged corpus STC (Wu et al., 2006). The assumption is similar to the celebrated one sense per collocation supposition (Yarowsky, 1993), whereas ours has more expansibility. STC is an ongoing project which is to build a sense-tagged corpus containing sense-tagged 1, 2 and 3 months of People's Daily 2000 now. According to our investigation, given a specific 3-gram ($w_{-1}w_{\text{verb}}w_1$) to any target verb, on average, we expect to see the same label 95.4% of the time. Based on this observation, we consider one sense per 3-gram ($w_{-1}w_{\text{verb}}w_1$) or at least we can extract instances with this pattern.

For all the 27 multiple-sense target verbs in the official training data, we found their 3-gram ($w_{-1}w_{\text{verb}}w_1$) and extracted the instances with the same 3-gram from a Chinese monolingual corpus – the 2001 People's Daily (about 116M bytes). We consider the same 3-gram instances should have the same label. Then an additional sense-labeled training corpus is built automatically in expectation of having 95.4% precision at most. And this corpus has 2145 instances in total (official training data have 4608 instances).

We build four systems to investigate the effect of our instances expansion using the Naïve Bayesian classifier. System configuration is shown in Table 1. In column 1, BL means baseline, X means instance expansion, 3 and 15 means the window size. In column 2, w_i is the *i-th* word relative to the target word, $w_{i-1}w_i$ is the 2-gram of words, w_j/j is the word with position information ($j\in$[-3,+3]). In the last column, 'O' means using only the original training data and 'O+A' means using both the original and additional training data. Syntactic feature and parameter optimizing are not used in this module.

System	Features	Window Size	Training Data
BL_3	$w_{\text{i}}, w_{\text{i-1}}w_{\text{i}}, w_{\text{j}}/j$	±3	O
X_3		±3	O+A
BL_15		±15	O
X_15		±15	O+A

Table 1: The system configuration

2.2 Sentence SRL and Event Detection

We use a feature-based machine learning method to implement the SRL module in which three tags are labeled, namely the semantic role tag, the event role tag and the phrase type tag. We consider the SRL task as a four-step pipeline: (1) **parsing** which generates a constituent parse tree for the input sentence; (2) **pruning** which filters out many apparently impossible constituents (Xue and Palmer, 2004); (3) **semantic role identification (SRI)** which identifies the constituent that will be the semantic role of a predicate in a sentence, and (4) **semantic role classification (SRC)** which determines the type of identified semantic role. The machine learning method takes PAS as the classification unit which consists of a target predicate and an argument candidate. The SRI step utilizes a binary classifier to determine whether the argument candidate in the PAS is a real argument. Finally, in the SRC step, the semantic role tag and the event role tag of each identified argument can be obtained by two multi-value classifications on the SRI results. The remaining phrase type tag can be directly extracted from the constituent parsing tree.

The selection of the feature set is the most important factor for the feature-based SRL method. In addition to constituent-based features and dependency-based features, we also consider WSD-based features. To our knowledge, the combined use of constituents-based syntactic features and dependency-based syntactic features is the first attempts to use them both on the feature level of SRL. As a prevalent kind of syntactic features for SRL, constituent-based features have been extensively studied by many researchers. In this module, we use 34 constituent-based features, 35 dependency-based features, and 2 WSD-based features. Among the constituent-based features, 26 features are manually selected from effective features proven by existing SRL studies and 8 new features are defined by us. Firstly, the 26 constituent-based features used by others are:

- *predicate* (c1), *path* (c2), *phrase type* (c3), *position* (c4), *voice* (c5), *head word* (c6), *predicate subcategorization* (c7), *syntactic frame* (c8), *head word POS* (c9), *partial path* (c10), *first/last word* (c11/c12), *first/last POS* (c13/c14), *left/right sibling type* (c15/c16), *left/right sibling head* (c17/c18), *left/right sibling POS* (c19/c20), *constituent tree distance* (c21), *temporal cue words* (c22), *Predicate POS* (c23), *argument's parent type*(c24), *argument's parent head* (c25) and *argument's parent POS* (c26).

And the 8 new features we define are:

- *Locational cue words* (c27): a binary feature indicating whether the constituent contains location cue word.
- *POS pattern of argument* (c28): the left-to-right chain of POS tags of argument's children.
- *Phrase type pattern of argument* (c29): the left-to-right chain of phrase type labels of argument's children.
- *Type of LCA and left child* (c30): The phrase type of the Lowest Common Ancestor (LCA) combined with its left child.
- *Type of LCA and right child* (c31): The phrase type of the LCA combined with its right child.
- Three features: *word bag of path* (c32), *word bag of POS pattern* (c33) and *word bag of type pattern* (c34), for generalizing three sparse features: *path* (c7), *POS pattern argument* (c28) and *phrase type pattern of argument* (c29) by the bag-of-words representation.

Secondly, the selection of dependency-based features is similar to that of constituent-based features. But dependency parsing lacks constituent information. If we want to use dependency-based features to label constituents, we should map a constituent to one or more appropriate words in dependency trees. Here we use head word of a constituent to represent it in dependency parses. The 35 dependency-based features we adopt are:

- *Predicate/Argument relation* (d1/d2), *relation path* (d3), *POS pattern of predicate's children* (d4), *relation pattern of predicate's children* (d5) , *child relation set* (d6), *child POS set* (d7), *predicate/argument parent word* (d8/d9), *predicate/argument parent POS* (d10/d11), *left/right word* (d12/d13), *left/right POS* (d14/d15), *left/right relation* (d16/d17), *left/right sibling word* (d18/d19), *left/right sibling POS* (d20/d21), *left/right sibling relation* (d22/d23), *dep-exists* (d24) and *dep-type* (d25), *POS path* (d26), *POS path length* (d27), *relation path length* (d28), *high/low support verb* (d29/d30), *high/low support noun* (d31/d32) and *LCA's word/POS/relation* (d33/d34/d35).

In this work, the dependency parse trees are generated from the constituent parse trees using a constituent-to-dependency converter (Marneffe et al., 2006). The converter is suitable for semantic analysis as it can retrieve the semantic head rather than the general syntactic head.

Lastly, the 2 WSD-based features are:

- *Situation description formula* (s1): predicate's situation description formula generated by the WSD module.
- *Natural explanation* (s2): predicate's natural explanation generated by the WSD module.

3 Experimental Results and Discussion

3.1 Target Verb WSD

System	Micro-A (%)	Macro-A (%)	Rank
BL_3	81.30	83.81	3/7
X_3	79.82	82.58	4/7
BL_15	79.23	82.18	5/7
X_15	77.74	81.42	6/7

Table 2: Official results of the WSD systems

Table 2 shows the official result of the WSD system. BL_3 with window size three using the original training corpus achieves the best result in our submission. It indicates the local features are more effective in our systems. There are two possible reasons why the performances of the X system with instance expansion are lower than the BL system. First, the additional instances extracted based on 3-gram provide a few local features but many topical features. But, local features are more effective for our systems as mentioned above. The local feature related information that the classifier gets from the additional instances is not sufficient. Second, the granularity of the WSD module is too small to be distinguished by 3-grams. As a result, the additional corpus built upon 3-gram has more exceptional instances (noises), and therefore it impairs the performance of X_3 and X_15. Taking the verb '属于' (belong to) as an example, it has two senses in the task, but both senses have the same natural explanation: '归一某方面或为某方所有' (part of or belong to), which is always considered as the sense in general SRL. The difference between the two senses is in their situation description formulas: 'partof (x,y)+NULL' vs. 'belongto (x,y)+NULL'.

3.2 Sentence SRL and Event Detection

In the SRL module, we use the training data provided by SemEval-2010 to train the SVM classifiers without any external resources. The training data contain 4,608 sentences, 100 target predicates and 13,926 arguments. We use the SVM-Light Toolkit (Joachims, 1999) for the implementation of SVM, and use the Stanford Parser (Levy and Manning, 2003) as the parser and the constituent-to-dependency converter. We employ the linear kernel for SVM and set the regularization parameter to the default value which is the reciprocal of the average Euclidean norm of the training data. The evaluation results of our SRL module on the official test data are shown in Table 3, where 'AB', 'SR', 'PT' and 'ER' represent argument boundary, semantic role tag, phrase type tag, and event role tag.

Tag	Precision(%)	Recall(%)	F-Score(%)
AB	73.10	66.83	69.82
AB+SR	67.44	61.65	64.42
AB+PT	61.78	56.48	59.01
AB+ER	69.05	63.12	65.95
Overall	58.33	53.32	55.71

Table 3: Official results of the SRL system

It is clear that 'AB' plays an important role as the labeling of the other three tags is directly based on it. Through analyzing the results, we find that errors in the recognition of 'AB' are mainly caused by two factors: the automatic constituent parsing and the pruning algorithm. It is inevitable that some constituents and hierarchical relations are misidentified in automatic parsing of Chinese. These errors are further enlarged by the heuristic-based pruning algorithm because the algorithm is built upon the gold-standard paring trees, and therefore a lot of real arguments are pruned out when using the noisy automatic parses. So the pruning algorithm is the current bottleneck of SRL in the evaluation.

System	Micro-A (%)	Macro-A (%)	Rank
BL_3	20.33	20.19	4/7
X_3	20.05	20.23	5/7
BL_15	20.05	20.22	6/7
X_15	20.05	20.14	7/7

Table 4: Official results of the ED systems

From the fact that the results of 'AB+SR' and 'AB+ER' are close to that of 'AB', it can be inferred that the SR and ER results should be satisfactory if the errors in 'AB' are not propagated. Furthermore, the result of 'AB+PT' is low as the phrase types here is inconsistent with those in Stanford Parser. The problem should be improved by a set of mapping rules.

Finally, in the ED module, we combine the results of WSD and SRL by filling variables of the situation description formula obtained by the WSD module with the arguments obtained by the SRL module according to their event role tags. Table 4 shows the final results which are generated by combining the results of WSD and SRL. Obviously the reduced overall ranking comparing to WSD is due to the SRL module.

4 Conclusions

In this paper, we propose a modular approach for the SemEval-2010 Task on Chinese event detection. Our system consists of three modules: WSD, SRL and ED. The WSD module is based on instances expansion, and the SRL module is based on rich syntactic features. Evaluation results show that our system is good at WSD, semantic role tagging and event role tagging, but poor at pruning and boundary detection. In future studies, we will modify the pruning algorithm to reduce the bottleneck of the current system.

Acknowledgments

This work is partially supported by the Hong Kong Polytechnic University under Grant No. G-U297 and G-U596, and by the National Natural Science Foundation of China under Grant No. 60736014 and 60803094.

References

Thorsten Joachims. 1999. Making large-Scale SVM Learning Practical. Advances in Kernel Methods. *Support Vector Learning*, B. Schölkopf and C. Burges and A. Smola (ed), MIT Press.

Roger Levy and Christopher D. Manning. 2003. Is it harder to parse Chinese, or the Chinese Treebank. *Proceedings of ACL-2003.*

Vasin Punyakanok, Dan Roth, and Wentau Yih. 2005. The necessity of syntactic parsing for semantic role labeling. *Proceedings of IJCAI-2005*.

Yunfang Wu, Peng Jin, Yangsen Zhang, and Shiwen Yu. 2006. A Chinese corpus with word sense annotation. *Proceedings of ICCPOL-2006*.

David Yarowsky. 1993. One sense per collocation. *Proceedings of the ARPA Workshop on Human Language Technology*.

Qiang Zhou. 2010. SemEval-2010 task 11: Event detection in Chinese News Sentences. *Proceedings of SemEval-2010*.