A Cascade Method for Detecting Hedges and their Scope in Natural Language Text

Buzhou Tang, Xiaolong Wang, Xuan Wang, Bo Yuan, Shixi Fan
Key Laboratory of Network Oriented Intelligent Computation
Harbin Institute of Technology Shenzhen Graduate School
Shenzhen, Guangdong, China
{tangbuzhou,yuanbo.hitsz}@gmail.com
{wangxl,wangxuan,fanshixi}@insun.hit.edu.cn

Abstract

Detecting hedges and their scope in natural language text is very important for information inference. In this paper, we present a system based on a cascade method for the CoNLL-2010 shared task. The system composes of two components: one for detecting hedges and another one for detecting their scope. For detecting hedges, we build a cascade subsystem. Firstly, a conditional random field (CRF) model and a large margin-based model are trained respectively. Then, we train another CRF model using the result of the first phase. For detecting the scope of hedges, a CRF model is trained according to the result of the first subtask. The experiments show that our system achieves 86.36% F-measure on biological corpus and 55.05% F-measure on Wikipedia corpus for hedge detection, and 49.95% F-measure on biological corpus for hedge scope detection. Among them, 86.36% is the best result on biological corpus for hedge detection.

1 Introduction

Hedge cues are very common in natural language text. Vincze et al. (2008) report that 17.70% of the sentences in the abstract section and 19.94% of sentences in the full paper section contain hedges on BioScope corpus. As Vincze et al. (2008) suggest that information that falls in the scope of hedges can not be presented as factual information. Detecting hedges and their scope in natural language text is very important for information inference. Recently, relative research has received considerable interest in the biomedical NLP community, including detecting hedges and their in-sentence scope in biomedical texts (Morante and Daelemans, 2009). The CoNLL-2010 has launched a shared task for exploiting the hedge scope annotated in the BioScope (Vincze et al., 2008) and publicly available Wikipedia (Ganter and Strube, 2009) weasel annotations. The shared task contains two subtasks (Farkas et al., 2010): 1. learning to detect hedges in sentences on BioScope and Wikipedia; 2. learning to detect the in-sentence scope of these hedges on BioScope.

In this paper, we present a system based on a cascade method for the CoNLL-2010 shared task. The system composes of two components: one for detecting hedges and another one for detecting their scope. For detecting hedges, we build a cascade subsystem. Firstly, conditional random field (CRF) model and a large margin-based model are trained respectively. Then, we train another CRF model using the result of the first phase. For detecting the scope of hedges, a CRF model is trained according to the result of the first subtask. The experiments show that our system achieves 86.36% F-measure on biological corpus and 55.05% F-measure on Wikipedia corpus for hedge detection, and 49.95% F-measure on biological corpus for hedge scope detection. Among them, 86.36% is the best result on biological corpus for hedge detection.

2 System Description

As there are two subtasks, we present a system based on a cascade supervised machine learning methods for the CoNLL-2010 shared task. The architecture of our system is shown in Figure 1.

The system composes of two subsystems for two subtasks respectively, and the first subsystem is a two-layer cascaded classifier.

2.1 Hedge Detection

The hedges are represented by indicating whether a token is in a hedge and its position in the CoNLL-2010 shared task. Three tags are used for

Proceedings of the Fourteenth Conference on Computational Natural Language Learning: Shared Task, pages 13–17,
Uppsala, Sweden, 15-16 July 2010. ©2010 Association for Computational Linguistics

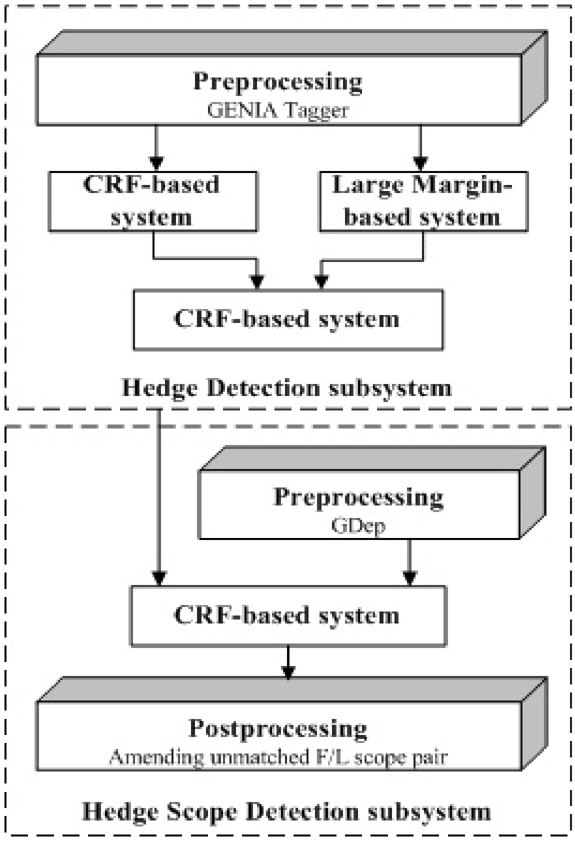

Figure 1: System architecture

this scheme, where O_cue indicates a token outside of a hedge, B_cue indicates a token at the beginning of a hedge and I_cue indicates a token inside of a hedge. In this subsystem, we do preprocessing by GENIA Tagger (version 3.0.1)[1] at first, which does lemma extraction, part-of-speech (POS), chunking and named entity recognition (NER) for feature extraction. For the output of GENIA Tagger, we convert the first char of a lemma into lower case and BIO chunk tag into BIOS chunk tag, where S indicates a token is a chunk, B indicates a token at the beginning of a chunk, I indicates a token inside of a chunk, and O indicates a token outside of a chunk. Then a two-layer cascaded classifier is built for prediction. There are a CRF classifier and a large margin-based classifier in the first layer and a CRF classifier in the second layer.

In the first layer, the following features are used in our system:

- Word and Word Shape of the lemma: we used the similar scheme as shown in (Tsai et al., 2005).
- Prefix and Suffix with length 3-5.
- Context of the lemma, POS and the chunk in the window [-2,2].
- Combined features including L_0C_0, L_iP_0 and L_iC_0, where $-1 \leq i \leq 1$ L denotes the lemma of a word, P denotes a POS and C denotes a chunk tag.
- The type of a chunk; the lemma and POS sequences of it.
- Whether a token is a part of the pairs ”neither ... nor” and ”either ... or” as both tokens of a pair are always labeled with the same tag.
- Whether a token can possibly be classified into B_cue, I_cue or O_cue; its lemma, POS and chunk tag for each possible case: these features are extracted according to a dictionary extracted from training corpus, which lists all possible hedge tag for each word in the training corpus.

In the second layer, we used some features about the result of the last layer besides those mentioned above. They are listed as follow:

- The lemma and POS sequences of the hedge predicted by each classifier.
- The times of a token classified into B_cue, I_cue and O_cue by the first two classifiers.
- Whether a token is the last token of the hedge predicted by each classifier.

2.2 Hedge Scope Detection

We follow the way of Morante and Daelemans (2009) to represent the scope of a hedge, where F_scope indicates a token at the beginning of a scope sequence, L_scope indicates a token at the last of a scope sequence, and NONE indicates others. In this phase, we do preprocessing by GDep Tagger (version beta1)[2] at first, which does lemma extraction, part-of-speech (POS), chunking, named entity recognition (NER) and dependency parse for feature extraction. For the output of GDep Tagger, we deal with the lemma and chunk tag using the same way mentioned in the last section. Then, a CRF classifier is built for prediction, which uses the following features:

[1]http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/tagger/

[2]http://www.cs.cmu.edu/ sagae/parser/gdep

- Word.
- Context of the lemma, POS, the chunk, the hedge and the dependency relation in the window [-2,2].
- Combined features including L_0C_0, L_0H_0, L_0D_0, L_iP_0, P_iC_0,P_iH_0, C_iH_0, P_iD_0,C_iD_0, where $-1 \leq i \leq 1$ L denotes the lemma of a word, P denotes a POS, C denotes a chunk tag, H denotes a hedge tag and D denotes a dependency relation tag.
- The type of a chunk; the lemma and POS sequences of it.
- The type of a hedge; the lemma, POS and chunk sequences of it.
- The lemma, POS, chunk, hedge and dependency relation sequences of $1st$ and $2nd$ dependency relation edges; the lemma, POS, chunk, hedge and dependency relation sequences of the path from a token to the root.
- Whether there are hedges in the $1st$, $2nd$ dependency relation edges or path from a token to the root.
- The location of a token relative to the negation signal: previous the first hedge, in the first hedge, between two hedge cues, in the last hedge, post the last hedge.

At last, we provided a postprocessing system for the output of the classifier to build the complete sequence of tokens that constitute the scope. We applied the following postprocessing:

- If a hedge is bracketed by a F_scope and a L_scope, its scope is formed by the tokens between them.
- If a hedge is only bracketed by a F_scope, and there is no L_scope in the sentence, we search the first possible word from the end of the sentence according to a dictionary, which extracted from the training corpus, and assign it as L_scope. The scope of the hedge is formed by the tokens between them.
- If a hedge is only bracketed by a F_scope, and there are at least one L_scope in the sentence, we think the last L_scope is the L_scope of the hedge, and its scope is formed by the tokens between them.
- If a hedge is only bracketed by a L_scope, and there is no F_scope in the sentence, we search the first possible word from the beginning of the sentence to the hedge according to the dictionary, and assign it as F_scope. The scope of the hedge is formed by the tokens between them.
- If a hedge is only bracketed by a L_scope, and there are at least one F_scope in the sentence, we search the first possible word from the hedge to the beginning of the sentence according to the dictionary, and think it as the F_scope of the hedge. The scope of the hedge is formed by the tokens between them.
- If a hedge is bracketed by neither of them, we remove it.

3 Experiments and Results

Two annotated corpus: BioScope and Wikipedia are supplied for the CoNLL-2010 shared task. The BioScope corpus consists of two parts: biological paper abstracts and biological full papers, and it is used for two subtasks. The Wikipedia corpus is only used for hedge detection. The detailed information of these two corpora is shown in Table 1 and Table 2, respectively.

	Abstracts	**Papers**	**Test**
#Documents	1273	9	15
#Sentences	11871	2670	5003
%Hedge sent.	17.70	19.44	15.75
#Hedges	2694	682	1043
#AvL. of sent.	30.43	27.95	31.30
#AvL. of scopes	17.27	14.17	17.51

Table 1: The detailed information of BioScope corpus. "AvL." stands for average length.

	Train	**Test**
#Documents	2186	2737
#Sentences	11111	9634
%Hedge sentences	22.36	23.19
#Hedges	3133	3143
#AvL. of sentences	23.07	20.82

Table 2: The detail information of Wikipedia corpus. "AvL." stands for average length.

In our experiments, CRF++-0.53[3] implemen-

[3]http://crfpp.sourceforge.net/

tation is employed to CRF, and svm_hmm_3.10[4] implementation is employed to the large margin method. All parameters are default except C (the trade-off between training error and margin, C=8000, for selecting C, the training corpus is partitioned into three parts, two of them are used for training and the left one is used as a development dataset) in svm_hmm. Both of them are state-of-the-art toolkits for the sequence labeling problem.

3.1 Hedge Detection

We first compare the performance of each single classifier with the cascaded system on two corpora in domain, respectively. Each model is trained by whole corpus, and the performance of them was evaluated by the official tool of the CoNLL-2010 shared task. There were two kinds of measure: one for sentence-level performance and another one for cue-match performance. Here, we only focused on the first one, and the results shown in Table 3.

Corpus	System	Prec.	Recall	F1
BioScope	CRF	87.12	86.46	86.79
	LM	85.24	87.72	86.46
	CAS	85.03	87.72	86.36
Wikipedia	CRF	86.10	35.77	50.54
	LM	82.28	41.36	55.05
	CAS	82.28	41.36	55.05

Table 3: In-sentence performance of the hedge detection subsystem for in-domain test. ”Prec.” stands for precision, ”LM” stands for large margin, and ”CAS” stands for cascaded system.

From Table 3, we can see that the cascaded system is not better than other two single classifiers and the single CRF classifier achieves the best performance with F-measure 86.79%. The reason for selecting this cascaded system for our final submission is that the cascaded system achieved the best performance on the two training corpus when we partition each one into three parts: two of them are used for training and the left one is used for testing.

For cross-domain test, we train a cascaded classifier using BioScope+Wikipedia cropus. Table 4 shows the results.

As shown in Table 5, the performance of cross-domain test is worse than that of in-domain test.

[4]http://www.cs.cornell.edu/People/tj/svm_light/svm-hmm.html

Corpus	Precision	Recall	F1
BioScope	89.91	73.29	80.75
Wikipedia	81.56	40.20	53.85

Table 4: Results of the hedge detection for cross-domain test. ”LM” stands for large margin, and ”CAS” stands for cascaded system.

3.2 Hedge Scope Detection

For test the affect of postprocessing for hedge scope detection, we test our system using two evaluation tools: one for scope tag and the other one for sentence-level scope (the official tool). In order to evaluate our system comprehensively, four results are used for comparison. The ”gold” is the performance using golden hedge tags for test, the ”CRF” is the performance using the hedge tags prediction of single CRF for test, the ”LM” is the performance using the hedge tag prediction of single large margin for test, and ”CAS” is the performance of using the hedge tag prediction of cascaded subsystem for test. The results of scope tag and scope sentence-level are listed in Table 5 and Table 6, respectively. Here, we should notice that the result listed here is different with that submitted to the CoNLL-2010 shared task because some errors for feature extraction in the previous system are revised here.

HD	tag	Precision	Recall	F1
gold	F_scope	92.06	78.83	84.94
	L_scope	80.56	68.67	74.14
	NONE	99.68	99.86	99.77
CRF	F_scope	78.83	66.89	72.37
	L_scope	72.52	60.50	65.97
	NONE	99.56	99.75	99.65
LM	F_scope	77.25	67.57	72.09
	L_scope	72.33	61.41	66.42
	NONE	99.56	99.73	99.31
CAS	F_scope	77.32	67.86	72.29
	L_scope	72.00	61.29	66.22
	NONE	99.57	99.73	99.65

Table 5: Results of the hedge scope tag. ”HD” stands for hedge detection subsystem we used, ”LM” stands for large margin, and ”CAS” stands for cascaded system.

As shown in Table 5, the performance of L_scope is much lower than that of F_scope. Therefore, the first problem we should solve is

HD subsystem	Precision	Recall	F1
gold	57.92	55.95	56.92
CRF	52.36	48.40	50.30
LM	51.06	48.89	49.95
CAS	50.96	48.98	49.95

Table 6: Results of the hedge scope in-sentence. "HD" stands for hedge detection subsystem we used, "LM" stands for large margin, and "CAS" stands for cascaded system.

how to improve the prediction performance of L_scope. Moreover, compared the performance shown in Table 5 and 6, about 15% (F1 of L_scope in Table 5 - F1 in Table 6) scope labels are mismatched. An efficient postprocessing is needed to do F-L scope pair match.

As "CRF" hedge detection subsystem is better than the other two subsystems, our system achieves the best performance with F-measure 50.30% when using the "CRF" subsystem.

4 Conclusions

This paper presents a cascaded system for the CoNLL-2010 shared task, which contains two subsystems: one for detecting hedges and another one for detecting their scope. Although the best performance of hedge detection subsystem achieves F-measure 86.79%, the best performance of the whole system only achieves F-measure 50.30%. How to improve it, we think some complex features such as context free grammar may be effective for detecting hedge scope. In addition, the postprocessing can be further improved.

Acknowledgments

We wish to thank the organizers of the CoNLL-2010 shared task for preparing the datasets and organizing the challenge shared tasks. We also wish to thank all authors supplying the toolkits used in this paper. This research has been partially supported by the National Natural Science Foundation of China (No.60435020 and No.90612005), National 863 Program of China (No.2007AA01Z194) and the Goal-oriented Lessons from the National 863 Program of China (No.2006AA01Z197).

References

Richárd Farkas, Veronika Vincze, György Móra, János Csirik, György Szarvas. 2010. The CoNLL-2010 Shared Task: Learning to Detect Hedges and their Scope in Natural Language Text. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning (CoNLL-2010): Shared Task*, pages 1–12, Uppsala, Sweden, July. Association for Computational Linguistics.

Viola Ganter and Michael Strube. 2009. Finding hedges by chasing weasels: Hedge detection using wikipedia tags and shallow linguistic features. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 173–176, Suntec, Singapore, August. Association for Computational Linguistics.

Roser Morante and Walter Daelemans. 2009. Learning the scope of hedge cues in biomedical texts. In *Proceedings of the BioNLP 2009 Workshop*, pages 28–36, Boulder, Colorado, June. Association for Computational Linguistics.

Tzong-Han Tsai, Chia-Wei Wu, and Wen-Lian Hsu. 2005. Using Maximum Entropy to Extract Biomedical Named Entities without Dictionaries. In *Second International Joint Conference on Natural Language Processing*, pages 268–273.

Veronika Vincze, György Szarvas, Richárd Farkas, György Móra, and János Csirik. 2008. The BioScope corpus: biomedical texts annotated for uncertainty, negation and their scopes. *BMC Bioinformatics*, 9(Suppl 11):S9.