

Exploiting Rich Features for Detecting Hedges and Their Scope

Xinxin Li, Jianping Shen, Xiang Gao, Xuan Wang

Harbin Institute of Technology Shenzhen Graduate School

Shenzhen, Guangdong, China

{lixxin2, jpshen2008}@gmail.com,

sky0306201@163.com, wangxuan@insun.hit.edu.cn

Abstract

This paper describes our system about detecting hedges and their scope in natural language texts for our participation in CoNLL-2010 shared tasks. We formalize these two tasks as sequence labeling problems, and implement them using conditional random fields (CRFs) model. In the first task, we use a greedy forward procedure to select features for the classifier. These features include part-of-speech tag, word form, lemma, chunk tag of tokens in the sentence. In the second task, our system exploits rich syntactic features about dependency structures and phrase structures, which achieves a better performance than only using the flat sequence features. Our system achieves the third score in biological data set for the first task, and achieves 0.5265 F1 score for the second task.

1 Introduction

In recent years, a fair amount of approaches have been developed on detecting speculative and negative information from biomedical and natural language texts, for its benefit to the applications like information extraction. These approaches evolve from hand-crafted rule-based approaches, which use regular expressions to match the sentences or its grammatical parsing, such as NegEx (Chapman et al., 2001), Negfinder (Mutalik et al., 2001), and NegExpander (Aronow et al., 1999), to machine learning approaches, including semi-supervised methods (Medlock and Briscoe, 2007; Szarvas, 2008), and supervised methods (Morante and Daelemans, 2009).

In this paper, we describe the machine learning system submitted to CoNLL-2010 Shared task (Farkas et al., 2010). Our system formalizes these two tasks as consecutive sequence labeling problems, and learns the classifiers using conditional random fields approach. In the first task, a model is trained to identify the hedge cues in sentences, and in the second task, another model is used to find the

corresponding scope for each hedge cue generated in the first task. Our system follows the study of Morante and Daelemans (2009), but applies more refined feature selection. In the first task, we use a greedy forward procedure to select features for the classifier. In the second task, we exploit rich syntactic information to improve the performance of the model, from dependency structures and phrase structures. A rule-based post processing procedure is used to eliminate the errors brought by the classifier for each task.

The remainder of the paper is organized as follows. In section 2, we briefly describe the task and the details of our system, including how to select features for the hedge cue detection system, and how to find the corresponding scope for each hedge cue. The experimental results are discussed in section 3. In section 4 we put forward some conclusion.

2 System Description

We model these two tasks for identifying the hedge cues and finding their scope as two consecutive sequence labeling problems, such as chunking, segmentation and named entity recognition, and train the classifiers using conditional random fields approach (Lafferty et al., 2001). For each task, a post-processing procedure is used to refine the results from the classifier.

In the first task, we detect the hedge cue by classifying the tokens of a sentence as being at the beginning of, inside or outside of the hedge signal. In the second task, we find the scope of a hedge cue by classifying the tokens of a sentence as being the first one of, the last one or neither of the scope.

A sentence from biological full articles data set omitting the id number is shown below in Figure 1. In this sentence, there is only one hedge cue, the phrase “raises an interesting question”, and its corresponding scope is the sequence from token “raises” to token “acid”.

<pre><sentence>This <xcope><cue>raises an interesting question</cue>: "Is there a 23rd amino acid</xcope>?".</sentence></pre>

Figure 1: A sentence with hedge cue and scope annotation in biological full articles data set

2.1 Hedge detection

Since hedge cues usually consist of one or more tokens, we predict the tokens in BIO representation, whether the token is the first token of a hedge cue (B-cue), inside a hedge cue (I-cue), or outside of the hedge cue (O-cue). For the sentence in Figure 1, token “raises” is denoted as B-cue, tokens “an interesting question” all as I-cue, and the other tokens in the sentence as O-cue.

The classifier is trained using conditional random fields (Lafferty et al., 2001), which combines the benefits of conditional models with the global normalization of random field models, and avoid the label bias problem that exists in maximum entropy Markov models (MEMMs). The CRF model we use is implemented as CRF++ 0.51¹. The parameters of the CRF classifier are set as defaults.

We use a greedy forward procedure to select a better feature sets for the classifier according to the evaluation results in the development set. We first start from a basic feature set, and then add each feature outside the basic set and remove each feature inside the basic set one by one to check the effectiveness of each feature by the performance change in the development set. This procedure is repeated until no feature is added or removed or the performance is not improved.

The selected features are listed below:

- C_n ($n=-2,-1, 0, 1, 2$)
- $C_n C_{n+1}$ ($n=-1,0$)
- $C_{n-1} C_n C_{n+1}$ ($n=-1,0,1$)
- $C_{n-2} C_{n-1} C_n C_{n+1}$ ($n=0,1$)

Where C denote features of each token, including FORM, LEMMA, and POS (in Table 1), C_0 represents the feature of current token and $C_n(C_{-n})$ represents the feature of the token n positions to the right (left) of current token. $C_n C_{n+1}$ denote the combination of C_n and C_{n+1} . So are $C_{n-1} C_n C_{n+1}$ and $C_{n-2} C_{n-1} C_n C_{n+1}$.

¹ <http://crfpp.sourceforge.net/>

Feature Name	Description
FORM	Word form or punctuation symbol.
LEMMA	Lemma or stem of word form.
POS	Part-of-speech tag of the token.
CHUNK	Chunk tag of the token, e.g. B_NP, B_SBAR, and I_NP.
TCHUNK	Chunk type of the token, e.g. NP.

Table 1: Description of features of each token

Although our system is based on token, chunk features are also important. Analyzing the training data set, it is shown that if one token in a chunk is in the hedge cue, the other tokens in the chunk are usually in the same hedge cue. The chunk feature can provide more information for the multiword hedge cues. The LEMMA, POS, and CHUNK of each token used in our system are determined using GENIA tagger (Tsuruoka et al., 2005).

The selected CHUNK features in our system are listed as follows:

- C_n ($n=-3, -2,-1, 0, 1, 2, 3$)
- $C_n C_{n+1}$ ($n=-3, -2,-1, 0, 1, 2, 3$)
- $C_{n-1} C_n C_{n+1}$ ($n=-2,-1,0,1,-2$)
- $C_{n-2} C_{n-1} C_n C_{n+1}$ ($n=-1,0,1,2$)

We can obtain the preliminary results using the CRF model-based classifier, but there are some missed or incorrectly classified hedge cues which can be recognized by rule-based patterns. Through statistical analysis on the training and development data sets, we obtain some effective rules for post processing, including:

- If the first token of a NP chunk tag is annotated as I-cue, the whole NP chunk is in the hedge cues.
- If the B-VP chunk tag of a token is followed by a B-SBAR chunk tag, the token is annotated as B-cue.
- If token “that” follows token “indicate” and the POS of token “that” is IN, the chunk tag of token “that” is B-SBAR, then the “indicate” will be annotated with B-cue and “that” will be annotated with I-cue.
- If token “indicate” is followed by token “an” or token “a”, then the token “indicate” is annotated as B-cue.

2.2 Scope finding

In this task, we train a classifier to predict whether each token in the sentence is in the scope by classifying them as the first one (F-scope), the last one (L-scope), or neither (NONE) of the scope, which is the same as Morante and Daelemans (2009). For the sentence in Figure 1, token “raises” is denoted as F-scope, token “acid” as L-scope, and the other tokens in the sentence as NONE.

After the classification, a post processing procedure is used to match the scope to each hedge, guaranteeing that each hedge has only one corresponding scope sequence, and must be inside its scope sequence. There is no cross between different scope sequences, but inclusion is allowed. The hedges are selected from the first task.

The classifier is also implemented using conditional random fields model, and the parameters of the CRF classifier are set as defaults. We first build a set of baseline sequence features for the classifier, some borrowed from Morante and Daelemans (2009). The selected baseline sequence features are:

- Of the token in focus: FORM, POS, LEMMA, CHUNK, TCHUNK, combination of FORM and POS; POS, LEMMA, CHUNK, TCHUNK of two tokens to the left and three tokens to the right; first word, last word, chain of FORM, POS of two chunks to the left and two chunks to the right; All combination of POS in the window of length less than 3; All combination of CHUNK in the window of length 2.
- Of the left closest hedge: chain of the FORM, POS, LEMMA, CHUNK, and TCHUNK; All combination of POS and FORM in the window of length 2.
- Of the right closest hedge: chain of the FORM, POS, LEMMA, CHUNK, and TCHUNK; All combination of POS and FORM in the window of length 2.
- Of the tokens between the left closest hedge and the token in focus: chain of FORM, POS, LEMMA, CHUNK and TCHUNK; the number.
- Of the tokens between the right closest hedge and the token in focus: chain of FORM, POS, LEMMA, CHUNK and TCHUNK; the number.

- Others: the number of hedge cues in the sentence; the sequence relation between the token in focus and hedge cues (LEFT, RIGHT, MIDDLE, IN, NULL)

Besides the sequence features listed above, syntactic features between the token in focus and hedge cues are explored in our classifier. Huang and Low (2007) notes that structure information stored in parse trees helps identifying the scope of negative hedge cues, and Szarvas (2008) points out that the scope of a keyword can be determined on the basic of syntax. Thus we believe that a highly accurate extraction of syntactic structure would be beneficial for this task.

For sentences in the dataset, their dependency structures are extracted using GENIA Dependency parser (Sagae and Tsujii, 2007), and phrase structure using Brown self-trained biomedical parser (McClosky, 2009). Figure 2 shows the corresponding dependency tree and Figure 3 shows the corresponding phrase structure tree for the sentence in Figure 1. In the following part in the section, we will illustrate these syntactic features and give examples for their value. We take the token “acid” as the token in focus, to determine whether it is classified as F-scope, L-scope or NONE.

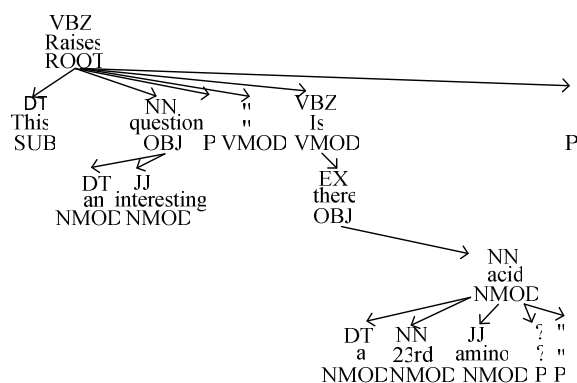


Figure 2: Dependency tree of the sentence in Figure 1

For the token “acid” in the dependency trees in Figure 2, its father node is the token “there”, and the dependency relation between these two token is “NMOD”.

Dependency features between the token in focus and the left closest hedge cue are:

- Dependency relation of the token in focus to its father, left closest hedge to its

father and the dependency relation pair: NOMD, ROOT, ROOT+NMOD.

- Chain of POS: ->VBZ<-VBZ<-EX<-NN
- Chain of POS without consecutive redundant POS: ->VBZ <-EX<-NN
- POS of their nearest co-father: VBZ
- Whether it is a linear relation (self, up, down, no): up
- Kinship (grandfather, grandson, father, son, brother, self, no): no.
- The number of tokens in the chain: 4

Similar features are extracted for dependency relation between the token in focus and its right closest hedge cue. There is no right hedge cue for token “acid”. Thus these features are set as “NULL”.

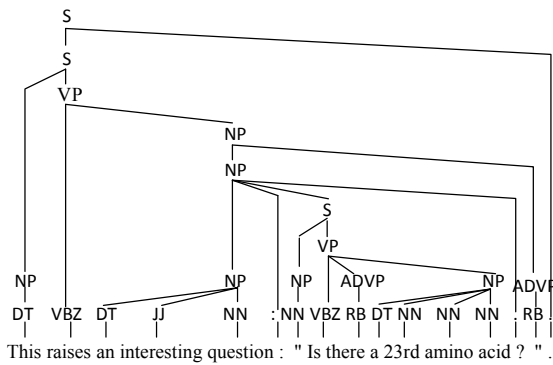


Figure 3: Phrase structure tree of the sentence in Figure 1

Phrase structure features between the token in focus and its left closest hedge cue are:

- Chain of syntactic categories: VBZ->VP<- NP <-NP <-S<-VP <-NP<-NN
- syntactic categories without consecutive redundant ones: VBZ->VP<-NP<-S<-VP<- NP<-NN
- Syntactic category of their nearest co-father: VP
- The number of syntactic categories in the chain: 8

The phrase structure features between the token in focus and the nearest right hedge cue are similar, setting as “NULL”.

Scope finding requires each hedge cue has only one corresponding scope. A hedge-scope

pair is true positive only if the hedge cue and its corresponding scope are correctly identified. We perform the post processing procedure in sequence:

- For each hedge cue from the beginning to the end of the sentence, find its left closest F-scope which has not been identified by other hedge cues, and identify it as its F-scope.
- For each hedge cue from the end to the beginning of the sentence, find its right closest L-scope which has not been identified by other hedge cues, and identify it as its L-scope.
- For each hedge:
 - ♦ If both its F-scope and L-scope is identified, then done;
 - ♦ If only its F-scope is identified, then its L-scope is set as L-scope of the last hedge cue in the sentence if it exists or according to the dictionary which we build with training data set;
 - ♦ If only its L-scope is identified, then its F-scope is set as its first token;
 - ♦ If none of its F-scope and L-scope is identified, then discard the hedge cue.

3 Overall Results

In this section we will present our experimental results for these two tasks. In the first task, the chief evaluation is carried on sentence level: whether a sentence contains hedge/weasel cue or not. Our system compares the performance of different machine learning algorithm, CRF and SVM-HMM on hedge cue detection. A post processing procedure is used to increase the recall measure for our system.

In the second task, three experiments are performed. The first experiment is used to validate the benefit of dependency features and phrase structure features for scope finding. The second experiment is designed to evaluate the effect of abstract dataset on full article dataset. These two experiments are all performed using gold hedge cues. The performance of our scope finding system with predicted hedge cues is presented in the third experiment.

3.1 Hedge detection

The first experiment is used to compare two machine learning algorithms, SVM-HMM and CRF. We train the classifiers on abstract and full articles data sets. The results of the classifier on evaluation data set are shown in Table 2.

Model	Precision	Recall	F1
SVM-HMM	88.71	81.52	84.96
CRF	90.4	81.01	85.45

Table 2: Results of hedge cues detection using CRF and SVM-HMM

From Table 1, it is shown that CRF model outperforms SVM-HMM model in both precision and recall measure. The results are obtained without post processing. The experimental result with post processing is shown in Table 3.

Feature	Precision	Recall	F1
Without Post processing	90.4	81.01	85.45
Post processing	90.1	82.05	85.89

Table 3: Result of biological evaluation data set without/with post processing

By post processing, some mislabeled or incorrectly classified hedge cues can be recognized, especially the recall of the I-cue improved largely, from 55.26% to 68.51%. Though the precision is a little lower, the F1 measure increases 0.44%.

3.2 Scope finding

To measure the benefit of syntactic features on scope finding task, we perform the experiment with different features on abstract data set, of which we split two-thirds as training data, and the other one third as testing data. The results are presented in Table 4.

We take the classifier with sequence features as baseline classifier. From Table 4, it is shown that adding dependency features achieves a slightly better performance than the baseline classifier, and adding phrase structure features improve much better, about 1.2% F1-score. The classifier with all syntactic features achieves the best F1-score, 2.19% higher than baseline classifier. However, in later experiment on evaluation dataset after the shared task, we

observed that dependency features actually harmed the performance for full articles dataset.

Feature set	Precision	Recall	F1
Sequence (Baseline)	82.20	81.61	81.90
Sequence + Dependency	82.28	82.09	82.19
Sequence + Phrase structure	83.14	83.04	83.09
All	84.19	83.99	84.09

Table 4: Results of scope finding system with different feature sets on abstract data set

Three experiments are designed to evaluate the benefit of abstract dataset for full articles dataset. The first one is performed on full articles data set, of which we split two-thirds for training, and the other one third for testing. The second experiment is trained on abstract data set, and evaluated on full articles data set. In the third experiment, we take abstract data set and one third of full articles as training data, and evaluate on the remaining full articles data set. The results are shown below in Table 5.

Training data	Testing data	Prec.	Recall	F1
Part Art.	Part Art.	53.14	51.80	52.46
Abs.	Full Art.	54.32	54.64	54.48
Mix	Part Art.	59.59	59.74	59.66

Table 5: Results of scope finding system with gold-standard hedge cues

Results in Table 5 reveal that more abstract and full article dataset are added to the classifier as training data, better performance the system achieve. Thus we use the combination of abstract and full articles as training data for the final evaluation.

Table 6 presents the results of our scope finding system with or without dependency features, using both gold-standard hedge cues and predicated hedge cues generated by our hedge cue finding system.

Comparing the results in Table 4, 5, and 6, we observe that the performance of scope finding classifier on full article dataset is much lower than on abstract dataset, and dependency features are beneficial for the abstract dataset, but useless for full article dataset. We ascribe this phenomenon to the lack of enough full articles training data and the different properties of

abstract and full articles data sets. Deep research is expected to continue.

Hedge cues	Dep. features	Prec.	Recall	F1
Predicted	with	57.42	47.92	52.24
	without	58.13	48.11	52.65
Gold standard	with	59.43	58.28	58.85
	without	60.20	58.86	59.52

Table 6: Results of scope finding system with/without dependency features using both gold-standard and predicated hedge cues

4 Conclusion

In this paper, we describe a machine learning system for detecting hedges and their scope in natural language texts. These two tasks are formalized as sequence labeling problems, and implemented using conditional random fields approach. We use a greedy forward procedure to select features for the classifier, and exploit rich syntactic features to achieve a better performance. In the in-domain evaluation, our system achieves the third score in biological data set for the first task, and achieves 0.5265 F1 score for the second task.

Acknowledgments

The authors would like to thank Buzhou Tang for useful discussions of the paper. This work is supported by the National High-tech R&D Program of China (863 Program, No. 2007AA01Z194).

References

- David B. Aronow, Fangfang Feng, and W. Bruce Croft. 1999. Ad Hoc Classification of Radiology Reports. *Journal of the American Medical Informatics Association*, 6(5):393–411.
- Wendy W. Chapman, Will Bridewell, Paul Hanbury, Gregory F. Cooper, and Bruce G. Buchanan. 2001. A Simple Algorithm for Identifying Negated Findings and Diseases in Discharge Summaries. *Journal of Biomedical Informatics*, 34:301–310.
- Richárd Farkas, Veronika Vincze, György Móra, János Csirik, and 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.
- Yang Huang, and Henry J. Lowe. 2007. A novel hybrid approach to automated negation detection in clinical radiology reports. *Journal of the American Medical Informatics Association*, 14(3):304–311.
- John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289.
- David McClosky. 2009. *Any Domain Parsing: Automatic Domain Adaptation for Natural Language Parsing*. Ph.D. thesis, Department of Computer Science, Brown University.
- Ben Medlock, and Ted Briscoe. 2007. Weakly supervised learning for hedge classification in scientific literature. In *Proc. of ACL 2007*, pages 992–999.
- Roser Morante, and Walter Daelemans. 2009. Learning the scope of hedge cues in biomedical texts. In *Proceedings of the Workshop on BioNLP*, pages 28–36.
- Pradeep G. Mutalik, Aniruddha Deshpande, and Prakash M. Nadkarni. 2001. Use of general-purpose negation detection to augment concept indexing of medical documents: a quantitative study using the UMLS. *Journal of the American Medical Informatics Association*, 8(6):598–609.
- Kenji Sagae, and Jun’ichi Tsujii. 2007. Dependency parsing and domain adaptation with LR models and parser ensembles. In *Proceedings of the CoNLL-2007 Shared Task*, pages 82–94.
- György Szarvas. 2008. Hedge classification in biomedical texts with a weakly supervised selection of keywords. In *Proc. of ACL 2008*, pages 281–289, Columbus, Ohio, USA. ACL.
- György Szarvas, Veronika Vincze, Richárd Farkas, and János Csirik. 2008. The BioScope corpus: annotation for negation, uncertainty and their scope in biomedical texts. In *Proc. of BioNLP 2008*, pages 38–45, Columbus, Ohio. ACL.
- Yoshimasa Tsuruoka, Yuka Tateishi, Jin-Dong Kim, Tomoko Ohta, John McNaught, Sophia Ananiadou, and Jun’ichi Tsujii. 2005. Developing a Robust Part-of-Speech Tagger for Biomedical Text. *Advances in Informatics - 10th Panhellenic Conference on Informatics*, LNCS 3746, pages 382–392.