PAPER

Voting-Based Ensemble Classifiers to Detect Hedges and Their Scopes in Biomedical Texts

Huiwei ZHOU[†a)], **Xiaoyan LI**[†b)], **Degen HUANG**[†c)], **Yuansheng YANG**[†d)], *Nonmembers*, *and* **Fuji REN**[††], *Member*

SUMMARY Previous studies of pattern recognition have shown that classifiers ensemble approaches can lead to better recognition results. In this paper, we apply the voting technique for the CoNLL-2010 shared task on detecting hedge cues and their scope in biomedical texts. Six machine learning-based systems are combined through three different voting schemes. We demonstrate the effectiveness of classifiers ensemble approaches and compare the performance of three different voting schemes for hedge cue and their scope detection. Experiments on the CoNLL-2010 evaluation data show that our best system achieves an F-score of 87.49% on hedge detection task and 60.87% on scope finding task respectively, which are significantly better than those of the previous systems.

key words: *hedges, voting, classification, machine learning*

1. Introduction

Speculative language, also known as hedging, is usually used in science text, especially in the biomedical domain. When researchers are not completely certain about the conclusions, they use speculative language to express this uncertainty. Hedged information is necessary for biomedical researchers to express impressions or hypothesized explanations of experimental results.

The term hedging is originally introduced by Lakoff [1]. However, researches on hedge detection from Natural Language Processing (NLP) perspective are just proposed in recent years. Vincze et al. [2] construct a corpus annotated for negations, speculations and their linguistic scopes. It provides a common resource for the training, the evaluation and the comparison of biomedical NLP systems.

Vincze et al. [2] report that 17.69% of the sentences in the abstracts section of the BioScope corpus and 22.29% of the sentences in the full papers section contain hedge cues. Light et al. [3] estimate that 11% of sentences in MEDLINE abstracts contain speculative fragments. Szarvas [4] reports that 32.41% of gene names mentioned in the hedge classification dataset [5] appear in the speculative sentences. This means that quite a few false positives of the gene interaction extraction system could be due to hedging if hedge detection had been neglected.

Manuscript received January 13, 2011.
Manuscript revised May 28, 2011.

[†]The authors are with Dalian University of Technology, Dalian City, China.
[††]The author is with the University of Tokushima, Tokushima-shi, 770–8501 Japan.
a) E-mail: zhouhuiwei@dlut.edu.cn
b) E-mail: lixiaoyan@dlut.edu.cn
c) E-mail: huangdg@dlut.edu.cn
d) E-mail: yangys@dlut.edu.cn

DOI: 10.1587/transinf.E94.D.1989

An example sentence containing hedged information is shown as follows:

(a) *It will be important to explore this connection further, since recent studies <xcope><cue>suggest</cue> an interaction between adenosine signaling and the NF-kappaB signaling pathway, which is the mammalian counterpart of the Toll pathway</xcope>.*

The word "*suggest*" indicates that the following statements are not supported by fact. The scope of the hedge cue "*suggest*" is the statement "*an interaction between adenosine signaling and the NF-kappaB signaling pathway, which is the mammalian counterpart of the Toll pathway*". But the statement "*It will be important to explore this connection further*" is factual information. Therefore, detections of hedge cues and their linguistic scope are both important in biomedical text mining.

The CoNLL-2010 Shared Task [6] formulates speculative language detection as two subtasks. Task 1 aims to identify sentences containing uncertainty and Task 2 aims to resolve the in-sentence scope of hedge cues. This paper aims to solve the two subtasks. As for task 1, Medlock and Briscoe [5] propose an automatic classification of hedging in biomedical texts using weakly supervised machine learning. Further, Medlock [7] illuminates the hedge identification task including annotation guidelines, theoretical analysis and discussion. He argues for separation of the acquisition and classification phases in semi-supervised machining learning method and presents a probabilistic acquisition model. Tang et al. [8] build a cascade subsystem to detect hedges in the CoNLL-2010 Shared Task. They first train a Conditional Random Field (CRF) model and a large margin-based model respectively. And then they train another CRF model using the result of the first phase. Their system achieves 86.36% F-score on biological corpus for hedge detection. It is the best result on Task 1. Zhou et al. [9] integrate a keyword-based system with a CRF-based system by introducing keyword features to the CRF-based system. Their system achieves a state-of-the-art F-score 86.32% in Task 1.

As for Task 2, Morante and Daelemans [10] present a meta-learning system that finds the scope of hedge cues in biomedical texts. They use three classifiers and a meta-learner that uses the predictions of the three classifiers to predict the scope classes. Morante et al. [11] develop a scope detector by using only one memory-based system that relies on information from syntactic dependencies. Their system scores the highest F-score (57.32%) of Task 2.

Recently, the combination of classifiers is used to achieve better performance [12], [13]. Jung et al. [12] propose an efficient matching scheme with a gradual voting strategy. Fishel and Nivre [13] study two techniques for combining data-driven dependency parser: voting and stacking. Their experimental results show that both methods lead to significant improvements over the best component system, while voting works better than stacking. A combination aggregates the results of many classifiers, overcoming the possible local weakness of the individual classifier, producing a more robust recognition. Stacking for hedge detection is adopted in detail in Tang et al. [8] and stacking for hedge scope detection is used in Morante and Daelemans [10] as mentioned above.

Aiming to further improve the performance of the uncertain information detection, we focus on the voting technique, which combines many individual classifiers to exploit the unique advantage of each classifier. First, we construct six basic classifiers based on three learning algorithms: Conditional Random Field (CRF) [14], Support Vector Machine (SVM) [15] and Max-Margin Markov Network (M^3-Net) [16] times two directions (forward and backward) for each task. Then three different voting schemes for system combination in hedge detection and hedge scope detection are compared.

2. Methods

2.1 Preprocessing

To extract features for the machine learners, we convert the XML training data to a token-per-token representation, in which a sentence consists of a sequence of tokens and each token starts on a new line.

In our preprocessing, GENIA Tagger[†] [17] is applied to get stems, Part-of-Speech (POS) tags, and BIO chunk tags. Dependency features are extracted by GDep parser[††] [18]. Table 1 shows a preprocessed sentence with the information per token: word, stem, POS tag, chunk tag, dependency label, cue tag, and scope tag. Hedge cues are given IOB2 representation. Scope tags representation follows the way of Morante and Daelemans [10], where F-scope indicates the first token of a scope sequence, L-scope indicates the last token of a scope sequence, and NONE indicates others.

Table 1 Preprocessed sentence.

token	Stem	POS	Chunk	Label	C	S
This	This	DT	B-NP	SUB	O	N
indicates	indicates	VBZ	B-VP	ROOT	B	F
that	that	IN	B-SBAR	VMOD	I	N
D-mib	D-mib	NN	B-NP	SUB	O	N
acts	acts	VBZ	B-BP	SBAR	O	N
at	at	IN	B-PP	VMOD	O	N
a	a	DT	B-NP	NMOD	O	N
step	step	NN	I-NP	PMOD	O	N
upstream	upstream	RB	B-ADVP	NMOD	O	N
of	of	IN	B-PP	AMOD	O	N
N	N	DT	B-NP	NMOD	O	N
activation	activation	NN	I-NP	PMOD	O	L
.	.	.	O	P	O	N

B:B-cue, I:I-cue, F:F-scope, L:L-scope, N:NONE

2.2 Task 1: Hedge Detection

We treat the detection of sentences containing uncertain information as token classification task in which we learn classifiers to predict whether a token is a cue or not.

2.2.1 Hedge Detection Classifiers

An important issue of classifiers combination is that each individual classifier should be complementary. This can be achieved by employing heterogeneous learning algorithms, as well as using different sets of the features. This paper investigates the heterogeneous classifiers ensemble strategy by using three machine learning algorithms: CRF, SVM and M^3-Net.

CRF: Conditional random field (CRF) is undirected graphical models trained to maximize a conditional probability [14]. The use of graphical models allows the structure of the labels to be exploited very effectively. CRF is discriminative model and can thus capture many correlated features of the inputs. This allows flexible feature designs for hierarchical tag sets. However, they do not have the generalization guarantees of SVM and the possibility to use the Kernel function. Moreover, they cannot give theoretical bound on the generalization error compared with those of margin-based classifiers.

SVM: Support Vector machine (SVM) takes a strategy that maximizes the margin between critical examples and the separating hyperplane [15]. The margin-maximizing properties of the learning algorithm ensure the high generalization of SVM even with training data of a very high dimension. Furthermore, by introducing the Kernel principle, SVM can handle non-linear feature spaces, and carry out the training in high-dimensional spaces with considering combinations of more than one feature. However, SVM doesn't handle interactions between the labels in the multi-label case. In sequence labeling, SVM ignores structure in the problem, assigning labels independently to each object, losing much useful information.

M^3-Net: Max-Margin Markov Networks (M^3-Net) is a new framework which unifies CRF and SVM, and combines the advantages of both [16]. The approach defines a log-linear Markov network over a set of label variables which allows us to capture correlations in structured data. Also, the margin-based optimization approach gives theoretical generalization guarantees. Taskar et al. [16] adapts structured sequential minimal optimization (SMO) algorithm for solving quadratic programming (QP) problems to train M^3-Net. However, the polynomial number of constraints in the QP problem associated with the M^3-Net can still be very large, making the structured SMO algorithm slow to converge over

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

[††]Available at http://people.ict.usc.edu/ sagae/parser/gdep/

the training data. This currently limits the scalability and applicability of M^3-Net to real-world structured data. Among the many kinds of Kernel functions available, we use the linear kernel with considering time cost.

The heterogeneity of the data influences the performance of CRF, SVM and M^3-Net. In our experiments, each machine learning algorithm is used to create two classifiers, one that parses the input sentence from left to right (forward), and the other from right to left (backward). Aggregately six cue classifiers are used in our voting-based combination experiments.

The following features are used for each component classifier.

- Word ($i = -n, \ldots, -1, 0, +1, \ldots, +n$)
- Stem ($i = -n, \ldots, -1, 0, +1, \ldots, +n$)
- POS ($i = -n, \ldots, -1, 0, +1, \ldots, +n$)
- Chunking tag ($i = -n, \ldots, -1, 0, +1, \ldots, +n$)
- Keyword feature ($i = -n, \ldots, -1, 0, +1, \ldots, +n$)

Hedge cues that appear in the training dataset and their synonyms in WordNet[†] are selected as keywords for hedge cue detection. To find the complete cues, keywords are matched through a maximum matching method (MM) [19].

The above linguistic features are base features, which are effective in hedge detection. Base features, by themselves cannot determine the current token correctly. In this paper, we introduce the following two additional features for hedge detection.

- Co-occurrence keyword feature

Speculation keywords usually co-occur in the sentences. Consider the sentence "*Therefore, we asked whether the observed iPfam coverage is larger than would be expected by chance.*" Here, "*whether*" and "*would*" are speculation keywords and their co-occurrence might be a clue for their speculation context. It is a binary feature which is set to "Y" if there are co-occurring speculation keywords. Otherwise, it is set to "N".

- Combined features

Combined features include $W_{i-1}W_i, P_0C_{-1}K_1$ and $C_0K_{-1}CO_1$, where $-1 \leq i \leq 1$, W denotes the word, P the POS tag, C the chunk tag, K the keyword feature and CO the co-occurrence keyword feature.

2.2.2 Voting-Based Ensemble Classifiers for Hedge Detection

Each individual cue classifier takes part in the decision of labeling an input token. In the simple majority voting [20], the decision of IOB2 labels is taken according to the number of votes given by all classifiers. Sagae and Lavie [21] report that they achieve a higher accuracy by applying weighted voting of systems. It is well-known that weighted voting scheme can maximize the margin between critical samples and the separating hyperplane, and produces a decision function with high generalization performance.

Three different voting schemes, therefore, are used in this paper: (1) majority voting; (2) weighted voting by the accuracy of the component classifier; (3) POS weighted voting by the accuracy of the component classifier on all tokens which have the same POS. We use six initial classifiers mentioned previously- three learning algorithms times two directions. Before applying weighted voting method, we need to decide on the weights to be given to individual classifiers. In all weighted voting experiments, the training corpus is used to train the component classifiers and the development corpus is used to learn weights. In our weighted voting experiments, 4-fold cross-validation on the CoNLL-2010 training dataset is used to get the voting weights. The voting weight of the classifier i is calculated as:

$$w(i) = \frac{\sum_{l_j \in \{B,I\}} C_i(l_j)}{\sum_{l_j \in \{B,I\}} Count(l_j)} \quad (1)$$

where $Count(l_j)$ is the total number of token whose class label is l_j in the development data, and $C_i(l_j)$ is the number of correctly tagged token whose label is l_j in the development data by the classifier i. Since the number of label O in the training corpus is far more than the number of label B-cue and I-cue, it is difficult to distinguish the hedge detection capability of the classifiers if label O is used to calculate the voting weight. Therefore we only use two classes (B-cue, I-cue) to calculate the voting weight.

2.3 Task 2: Hedge Scope Detection

Only sentences assigned cues in the hedge detection phase are selected for hedge scope detection.

2.3.1 Voting-Based Hedge Scope Detection

We also adopt the three algorithms - CRF, SVMs and M^3-Net to implement hedge scope detection system.

The features used in hedge scope detection systems are listed below.

- Word
- Stem
- POS
- Chunk tag
- Hedge cues feature (H)

Hedge cues labels that are doped out in Task 1 are selected as an important feature.

- Distance from the current word to the closest preceding and following hedge cue (DH)
- Stem of the closest preceding and following hedge cue (SH)
- POS of the closest preceding and following hedge cue (PH)
- Dependency feature

[†]Available at http://wordnet.princeton.edu/

Dependency label of current word, POS tag and dependency label of the head of cue in the dependency tree are used in our experiments.

- Phrase structure feature (PS)

The phrase structure information plays an important role in hedge scope detection. Enju[†] is used to get the phrase structure feature. The parent phrase immediately dominating the cue is given IOB2 label as phrase structure feature.

- Context feature

Context of the word, stem, POS, chunk tag and hedge cue feature in the window [−4, 4] and Context of DH, SH, PH, dependency feature and syntactic feature in the window [−2, 2] are used in our experiments.

- Combined feature

Combined features including $C_{i-1}C_i, C_{i-1}C_iC_{i+1}$, $P_0C_{-1}H_1$ and $C_0H_{-1}DH_1$, where $-1 \leq i \leq 1$, C denotes the chunk tag, P the POS of the word, H the hedge cue feature and DH the distance to the closest preceding and following hedge cue.

The three voting schemes used in hedge detection system are also used in hedge scope detection system.

2.3.2 Post-Processing

In CoNLL-2010 share task corpus, scopes are annotated as continuous sequences of tokens that include the cue. However, sometimes the classifiers only predict the first or the last token of the scope. Therefore, we need to process the output of the classifier to get the complete sequence of the scope. The following post processing rules are adopted.

1. If one token has been predicted as F-scope and one as L-scope, the sequence will start at the token predicted as F-scope, and end at the token predicted as L-scope.
2. If one token has been predicted as F-scope and none has been predicted as L-scope, the sequence will start at the token predicted as F-scope and end at the end of the sentence.
3. If one token has been predicted as L-scope and none has been predicted as F-scope, the sequence will start at the hedge cue and finish at the token predicted as L-scope.
4. If one token has been predicted as F-scope and more than one has been predicted as L-scope, the sequence will start at the token predicted as F-scope and end at the first token predicted as L-scope.
5. If one token has been predicted as L-scope and more than one has been predicted as F-scope, the sequence will start at the first token predicted as F-scope and finish at the token predicted as L-scope.
6. If an L-scope is predicted before an F-scope, the sequence will start at the token predicted as F-scope, and finish at the end of the sentence.

3. Results and Discussion

3.1 Experimental Settings

Our experimental results are all based on the CoNLL-2010 BioScope dataset. The BioScope corpus consists of two parts: biological paper abstracts and biological full papers. The test set includes 5003 sentences and 1043 of them contain uncertain information. The evaluation of hedge detection task is carried out using the sentence-level F-score of the uncertainty class, and the results are calculated with the official scorer provided by the task organizers.

In our experiments, CRF++-0.54[††] implementation is employed to CRF, YamCha-0.33[†††] implementation is employed to SVM method and pocket_m3n_0.11[††††] implementation is employed to M^3-Net.

We set $d = 3$ for the dimension of the polynomial kernel function of SVMs.

3.2 Hedge detection performance

Table 2 shows the effects of the additional features proposed in Sect. 2.2.1. The base features are the same as those used in the hedge detection system of Zhou [9]. The hedge tags are predicted based on CRF algorithm in forward parsing under the condition $C = 1$ (the hyper-parameter of CRF) and $n = 4$ (window size), which is exactly the same setting in Zhou [9]. From these results, we can conclude that all additional features proposed in this paper are effective in improving the performance.

Table 3 shows the relationship between the window size and the hedge detection performance. Using all the base features and the additional features, the best F-score 86.71 is obtained when the window size is ±2.

Table 4 shows the relationship between the hyper-parameter $C \in R^+$ of CRF and the hedge detection performance. This parameter trades the balance between overfitting and underfitting. With larger C value, CRF tends to

Table 2 Effects of the additional features in hedge detection.

Feature set	Prec.	Recall	F-score
base features (Zhou [9])	87.21	85.44	86.32
+co-occurrence keyword feature	87.73	85.06	86.38
+co-occurrence keyword ,combined feature	87.74	85.19	86.45

Table 3 Window size n vs. Performance in hedge detection.

Window size	Prec.	Recall	F-score
1	85.93	85.82	85.88
2	88.01	85.44	86.71
3	88.43	84.18	86.25
4	87.74	85.19	86.45

[†]Available at http://www-tsujii.is.s.u-tokyo.ac.jp/enju/
[††]Available at http://crfpp.sourceforge.net/
[†††]Available at http://chasen.org/ taku/software/yamcha/
[††††]Available at http://sourceforge.net/projects/pocket-crf-1/files/

Table 4 Hyper-parameter C vs. Performance in hedge detection.

Classifier	C	Prec.	Recall	F-score
CRF-F	1.0	88.01	85.44	86.71
	2.0	87.90	86.08	87.01
	3.0	88.24	86.46	87.34
	4.0	88.02	86.46	87.23
CRF-B	1.0	87.53	85.32	86.41
	2.0	87.73	85.95	86.83
	3.0	87.98	86.20	87.08
	4.0	88.08	86.08	87.07

Table 5 Hedge detection results of six individual classifiers and their ensemble through three voting schemes.

Classifier	Prec.	Recall	F-score
CRF-F	88.24	86.46	87.34
CRF-B	87.98	86.20	87.08
SVM-F	88.29	84.94	86.58
SVM-B	87.28	85.95	86.61
M^3-NET-F	87.79	85.57	86.67
M^3-NET-B	87.22	84.68	85.93
Majority voting	88.46	86.33	87.38
Weighted voting	88.06	86.84	87.44
POS weighted voting	88.69	86.33	87.49

F: Forward, B: Backward

overfit the given training corpus. As shown in Table 4, the results are significantly influenced by parameter C, which can be determined by a cross validation. As a result, the case of $C = 3$ gives the best F-score.

Table 5 compares the performance of six individual classifiers and three voting-based ensemble classifiers. By applying the ensemble, we can see that all voting schemes are effective in improving the performance. Majority scheme takes advantage of combination, and gets a higher F-score than any single classifier. Weighted voting scheme achieves a little improvement than majority voting scheme because weighted voting gives appropriate weights to individual classifiers. POS weighted voting scheme achieves the best F-score of 87.49, which benefits from grouping weights. It can be concluded that POS is an important feature for grouping weights.

Table 6 shows the different weights grouped by the word POS. It can be seen that six classifiers have different weights for different POS. The weights of CC (coordinating conjunction) are very low no matter what classifier is selected. This means it is difficult to detect the hedge cues whose POS tag is CC. The test set includes 888 coordinating conjunctions and 67 of them are hedge cues. The coordinating conjunctions such as "either", "or", "and", "but" and so on do not have speculative meaning. Through the analysis of the sentences containing coordinating conjunctions, we find that whether a conjunction is a cue or not should be determined by considering the whole sentence. For example, consider the following two sentences.

(b) *However, the increase in ISGF3 activity ultimately correlates with the accumulation of ISGF3 gamma induced by IFN-alpha <cue>or</cue> IFN-gamma.*

(c) *Furthermore, CD28 coligation fails to enhance IL-2 promoter-reporter or RE/AP construct expression in CD2-activated LPMC.*

Table 6 Weights of POS weighted voting scheme for hedge detection.

POS	Estimated weights					
	CRF-F	CRF-B	SVM-F	SVM-B	M^3-Net-F	M^3-Net-B
JJ	.7424	.6494	.6234	.64	.7227	.6625
RB	.9167	.8269	.8148	.8182	.8958	.8302
VBG	.9538	.9688	.9385	.9524	.9538	.9385
VBD	.9211	.925	.9	.8974	.9474	.9
NN	.6667	.8235	.8125	.8	.7222	.8
VBN	.8571	.7447	.7872	.7959	.881	.7347
VB	1.0	.9286	.8667	.9231	1.0	.9286
VBP	.9835	.9835	.978	.9834	.9725	.9834
CC	.1034	.5333	.3333	.2857	.1034	.5
MD	.9362	.8667	.8911	.881	.9404	.8656
VBZ	.9859	.9467	.9589	.9452	.9859	.9467

Table 7 Hedge detection performance comparison on the CoNLL-2010 Share Task test data.

System	Prec.	Recall	F-score
Ours	88.69	86.33	87.49
Tang [8]	85.03	87.72	86.36
Zhou [9]	86.49	85.06	85.77
Li [22]	90.40	81.01	85.45

Table 8 Scope detection results of six individual classifiers and their ensemble through three voting schemes.

Classifier	Prec.	Recall	F-score
CRF-F	61.16	57.31	59.17
CRF-B	61.05	57.21	59.07
SVM-F	61.05	57.21	59.07
SVM-B	61.36	57.50	59.37
M^3-Net-F	61.57	57.70	59.57
M^3-Net-B	61.47	57.60	59.47
Majority voting	61.67	57.79	59.67
Weighted voting	61.78	57.89	59.77
POS weighted voting	61.98	58.08	59.97

The word "or" in sentence (b) is a cue, which in sentence (c), "or" is not used in a speculative context. Exploiting neighboring words features to label the current word is a traditional method in the Natural Language Processing. But in fact, features in a narrow window are not rich enough to analyze the meaning of the word in the whole sentence. However, using features of the whole sentence would cause the increase of computational cost as well as the problem of data sparseness. This conflict is a problem that we should resolve in the future.

Table 7 summarizes the top three systems from the competition of CoNLL-2010 Share Task [6]: Tang et al. [8], Zhou et al. [9] and Li et al. [22]. It is obvious that our system outperforms the best system by an increase of 1.13 in F-score.

3.3 Hedge Scope Detection Performance

For hedge scope detection system, we use the hedge cues extracted by the POS weighted voting system (the best F-score of 87.49 in Table 5). Table 8 shows the performance of each single classifier and voting-based ensemble classifiers.

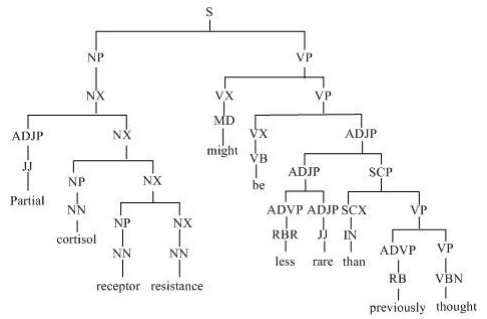

Fig. 1 Phrase tree parsed by Enju Parser.

Table 9 Effects of phrase structure feature (PS) for scope detection.

Classifier	Without PS feature			With PS feature		
	Prec.	Recall	F-score	Prec.	Recall	F-score
CRF-F	59.92	56.15	57.97	+1.24	+1.16	+1.2
CRF-B	59.71	55.95	57.77	+1.34	+1.26	+1.3
SVM-F	59.71	55.95	57.77	+1.34	+1.26	+1.3
SVM-B	60.12	56.34	58.17	+1.24	+1.16	+1.2
M^3-Net-F	59.81	56.05	57.87	+1.76	+1.65	+1.7
M^3-Net-B	60.02	56.24	58.07	+1.45	+1.36	+1.4

All results of ensemble classifiers are better than those of individual classifiers. POS weighted voting system achieves a state-of-the-art F-score 59.97 in Task 2.

It needs to be pointed out that the scope of a hedge cue is related to its POS and the phrase structure of the sentence in which it occurs. Consider the uncertain sentence "*Partial cortisol receptor resistance <xcope><cue>might</cue> be less rare than previously thought</xcope>.*" whose phrase tree parsed by Enju parser is shown in Fig. 1. Enju is an accurate HPSG parser for English. We use it to analyze syntactic structures of sentences in our experiment since it contains a parsing model for the biomedical domain.

The sentence in Fig. 1 contains a hedge cue "*might*". The scope of the modal verb "*might*" extends to the parent phrase "VP" immediately dominating it. In this paper, we introduce a new type of feature called phrase structure feature (PS), which is effective in improving the performance. Its contributions to the performance are shown in Table 9.

Table 10 shows the weights of POS weighted voting for hedge scope detection. The low accuracy of the scope of speculation CC is caused by the low accuracy of hedge cue detection as shown in Table 6.

The weights of VB (Verb) and VBN (Verb, past participle) are very low for all individual classifiers. By analyzing the error of hedge cues which have VB and VBN tag, we find the problem is caused by a verb in passive voice like "be (VB) + past participle (VBN)". The scope of a verb in passive voice extends to the whole sentence such as the scope of "*thought*" in "*<xcope>Activation of NF-kappaB is <cue>thought</cue> to be required for cytokine release from LPS-responsive cells, a critical step for endotoxic effects</xcope>*". In this case, most hedge cues are predicted as F-scope by our classifiers mistakenly.

The weight of RB (Adverb) is also low, since it is difficult for a classifier to determine the scope of the speculation adverbs. Some scopes of an adverb extend to the whole sentence such as the scope of "*<xcope>Cosignaling via the LT-beta and TNF-alpha receptors is <cue>probably</cue> involved in the modulation of HIV-1 replication and the subsequent determination of HIV-1 viral burden in monocytes</xcope>*". However, some scopes of an adverb start with the cue and end with the last token of the highest level "NP" which dominates the adverb, such as in "*Thus, the novel enhancer element identified in this study is <xcope><cue>probably</cue> a target site for both positive and negative factors</xcope>*".

To correct the errors above, scopes are reconstructed from the POS weighted classifier output by using the following additional post-processing rules:

1. The scope of a verb in passive participle voice is extended to the whole clause.
2. If the detected cue is an adverb which modifies a verb, the scope of the adverb is extended to the whole sentence. Otherwise, the scope of the adverb starts with the hedge cue and ends with the last token of the highest level "NP" which dominates the adverb.

Their contributions to the performance are shown in Table 11. From the results we can conclude that all additional rules are effective in improving the performance.

Table 10 Weights of POS weighted voting scheme for hedge scope detection.

POS	Estimated weights					
	CRF-F	CRF-B	SVM-F	SVM-B	M^3-Net-F	M^3-Net-B
JJ	.5833	.5595	.6071	.5952	.5953	.5714
RB	.4	.4	.425	.425	.375	.4
VBG	.9155	.9155	.9155	.9155	.8873	.9014
VBD	.7083	.7083	.6875	.7083	.7083	.7083
NN	.8235	.8235	.7059	.7059	.8235	.8235
VBN	.3529	.3529	.4706	.4706	.2941	.3529
VB	.3636	.3636	.3636	.3636	.3636	.3636
VBP	.8235	.8235	.8396	.8342	.8449	.8182
CC	.0968	.0968	.1613	.0968	.0968	.0968
MD	.7259	.7259	.736	.731	.731	.7208
VBZ	.5246	.5246	.541	.5246	.4754	.4918

Table 11 Hedge scope detection performance improvement by additional post-processing.

Additional post-processing	Different from the model without additional post-processing		
	Prec.	Recall	F-score
rule1	+0.52	+0.49	+0.5
rule2	+0.42	+0.39	+0.4
Rule1+rule2	+0.93	+0.87	+0.9

Table 12 Hedge scope detection performance comparison on the CoNLL-2010 Share Task test data.

System	Prec.	Recall	F-score
Ours	62.91	58.95	60.87
Roser Morante et al. [11]	59.62	55.18	57.32
Rei and Briscoe [23]	56.74	54.60	55.65
Velldal et al. [24]	56.71	54.02	55.33

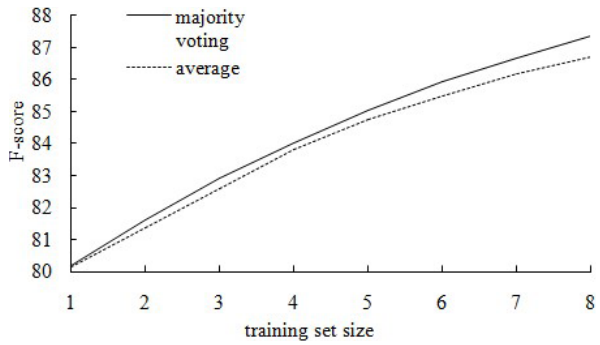

Fig. 2 Training data vs. hedge detection performance for majority voting.

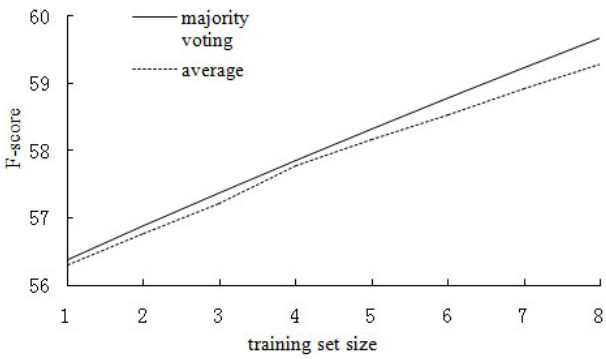

Fig. 3 Training data vs. hedge scope detection performance for majority voting.

We compare our system with the top three systems from the competition of CoNLL-2010 Share Task in Table 12. It is obvious that our system outperforms them by a significant increase of 3.55 in F-score.

3.4 Training Data vs. Ensemble Performance

Since the BioScope corpus only includes 5003 sentences and 1043 of them contain uncertain information, the contribution of ensemble in our experiments is not very obvious. However, the advantage of the ensemble approach is its expansibility. The performance of ensemble system can be easily improved by adding more training data to the current corpus. The effect of the training data size on the majority voting is investigated in this paper.

We divide the BioScope training data into eight parts. One part is used for training the six individual hedge detection classifiers (three learning algorithms two directions), which are then combined by majority voting. Next we increase the size of the training dataset by one part incrementally to train the other basic classifiers and then combine the six classifiers trained by the same size training set. Figure 2 shows the hedge detection results using majority voting for different training set sizes. Under majority voting, the F-score of the hedge detection increase significantly as the training dataset size increases. To estimate the asymptotic value of the F-score for hedge detection as the training set n increases, non-linear function $f_{hedge} = i + jm^n$ is fitted to the results shown in Fig. 2. The variable f_{hedge} is the F-score for hedge detection, n is the training dataset size, and i, j, and m are parameters requiring estimation. The statistical tool SPSS[†] is used to carry out the estimation and the following function is obtained.

$$f_{hedge} = 92.58 - 14.0391(0.8832)^n \quad (2)$$

This function suggests that as the training set increases, the F-score starts at about 80.18 for a training dataset of size 1 and reaches an asymptote at about 92.58.

Figure 3 shows the same information for hedge scope detection using majority voting. The following estimation is obtained for hedge scope detection.

$$f_{hedge} = 75.67 - 19.8240(0.9736)^n \quad (3)$$

This function suggests that as the training dataset increases, the F-score of hedge detection starts at about 56.37 for a training dataset of size 1 and reaches an asymptote at about 75.67.

The average results of hedge and their scope detection for different training set sizes are also shown in Fig. 2 and Fig. 3 respectively. All results of ensemble classifiers are steadily better than the average results on both hedge detection and their scope finding for different training set sizes. The bigger the training set is, the more significant the ensemble system improvement is. It is worthwhile to note that the F-score of the ensemble classifiers with very large training data is not 100. Some hedges may be dropped during statistical detection. This is different from the pure rule-based approach.

3.5 The Upper Bounds for Ensembles

Given an ensemble of n classifiers $\{C_1(x), \ldots, C_n(x)\}$, where n is an even integer, the majority voting classifier is defined as:

$$C(x) = \begin{cases} 1 & \sum_{i=1}^{n} C_i(x) \geq n/2 \\ 0 & Otherwise \end{cases} \quad (4)$$

The majority voting threshold k is set to be 3 in our experiments. The token is tagged as B-cue or I-cue when not less than half of the classifiers agree for the hedge detection; while the token is tagged as F-scope or L-scope when not less than half of the classifiers agree for the hedge scope detection.

For multinomial classes with $|Y|$ class values, the threshold under which a voting system makes a correct prediction depends on the distribution of the incorrect votes. This means that to get a good voting result, selecting a set of basic classifiers that are more different in their predictions can minimize the required threshold. The number of correct votes required may be as low as $n/|Y|$ or as high as $n/2$. Therefore the threshold may be 2 when the number of

[†]http://www.spss.com/

the three class votes is 2 respectively.

The upper bounds of the F-score for majority voting can be estimated by minimizing the required threshold $n/|Y|(k = 2)$ using the Eq. (5):

$$max(C(x)) = F\left(\sum_{i=1}^{6} I(C_i(x) = Y_c) \geq 3+ \sum_{i=1}^{6} I(C_i(x) = Y_c) = \sum_{i=1}^{6} I(C_i(x) = Y_1) = \sum_{i=1}^{6} I(C_i(x) = Y_2) = 2\right) \quad (5)$$

Where Yc is the correct prediction. x is the instance whose classification to be confirmed. Experiments on the CoNLL-2010 evaluation data show that our voting-based ensemble system achieves an upper bound F-score of 91.87 on hedge detection task and 62.77 on scope finding task respectively. Though the upper bound predicted by the Eq. (5) is tight, the contribution of ensemble is relatively large. This owes to the complement of the individual classifiers.

3.6 Overall Performance Comparison

From prior experiments on hedge detection (Table 5), it can be seen that CRF achieves the best performance among the three algorithms both in forward parsing and backward parsing. However, for hedge scope detection (Table 8), M^3-Net models perform better than CRF and SVM. Generally, the performance of SVM is moderate or steady among the three algorithms in both tasks. This is due to the high generalization ability of SVMs.

From the experiments on hedge detection for different kinds of the word POS (Table 6), it is easy to note the extremely good results of the CRF models on the tokens with POS tags VB*. As for hedge scope detection (Table 10), SVM outperform the CRF and M^3-Net on the tokens with POS tags VB*. Overall, the performance of six basic classifiers is comparable and influenced by the heterogeneity of the data. The proposed classifiers ensemble methods lead to improvement in performance of hedge and their scope detection.

Concerning the time needed to train the models, CRFs certainly behave most favorable. YamCha uses two fast classification algorithms -PKE (Polynomial Kernel Expanded) and PKI (Polynomial Kernel Inverted) to make the classification speed faster than the original SVMs. However, the SMO-style M^3-Net algorithm employed here remains problematic in training time cost. The M^3-Net models are trained using linear kernel here since we could not carry out the training in realistic time for the kernel of quadratic polynomial.

4. Conclusions

This paper focuses on the voting scheme to detect hedges and their scopes in biomedical texts. More specifically, three voting schemes are adapted to voting-based ensemble classifiers. The experimental results show that voting may result in improvements over their component classifiers by combining their individual advantages. Weighted voting strategies perform better than the simple majority voting method. Furthermore, grouping weights can achieve even higher performance. The best voting scheme for hedge detection and hedge scope detection achieves F-scores of 87.49% and 60.87% respectively, which far surpass the best published results.

References

[1] G. Lakoff, "Hedges: A study in meaning criteria and the logic of fuzzy concepts," J. Philosophical Logic, vol.2, no.4, pp.458–508, 1972.
[2] V. Vincze, G. Szarvas, R. Farkas, G. Mra, and J. Csirik, "The bioscope corpus: Biomedical texts annotated for uncertainty, negation and their scopes," BMC Bioinformatics, vol.9, Suppl 11.S9, 2008.
[3] M. Light, X.Y. Qiu, and P. Srinivasan, "The language of bioscience: Facts, speculations, and statements in between," Proc. BioLINK, pp.17–24, 2004.
[4] G. Szarvas, "Hedge classification in biomedical texts with a weakly supervised selection of keywords," Proc. Association for Computational Linguistics, pp.281–289, 2008.
[5] B. Medlock and T. Briscoe, "Weakly supervised learning for hedge classification in scientific literature," Proc. ACL-07, 45th Annual Meeting of the Association of Computational Linguistics, pp.992–999, 2007.
[6] R. Farkas, V. Vincze, G. Mra, J. Csirik, and G. Szarvas, "The CoNLL 2010 shared task: Learning to detect hedges and their scope in natural language text," Proc. CoNLL2010 Shared Task, 2010.
[7] B. Medlock, "Exploring hedge identification in biomedical literature," Journal of Biomedical Informatics, vol.41, Suppl.4, pp.636–654, 2008.
[8] B.Z. Tang, X.L. Wang, X. Wang, B. Yuan, and S.X. Fan, "A cascade method for detecting hedges and their scope in natural language text," Proc. Fourteenth Conference on Computational Natural Language Learning (CoNLL 2010): pp.25–29, Shared Task, Uppsala, Sweden, Association for Computational Linguistics, July 2010.
[9] H.W. Zhou, X.Y. Li, D.G. Huang, Z.Z. Li, and Y.S. Yang, "Exploiting multi-features to detect hedges and their scope in biomedical texts," Proc. Fourteenth Conference on Computational Natural Language Learning (CoNLL 2010): Shared Task, pp.106–113, Uppsala, Sweden, Association for Computational Linguistics, July 2010.
[10] R. Morante and W. Daelemans, "Learning the scope of hedge cues in biomedical texts," Proc. Workshop on BioNLP, Association for Computational Linguistics, pp.28–36, 2009.
[11] R. Morante, V.V. Asch, and W. Daelemans, "Memory-based resolution of in-sentence scopes of hedge cues," Proc. Fourteenth Conference on Computational Natural Language Learning (CoNLL 2010): Shared Task, Uppsala, Sweden, Association for Computational Linguistics, pp.48–55, July 2010.
[12] J.H. Jung, H.S. Lee, and D.J. Park, "A fast block matching technique using a gradual voting strategy," IEICE Trans. Inf. Syst., vol.E93D, no.4, pp.926–929, April 2010.
[13] M. Fishel and J. Nivre, "Voting and stacking in data-driven dependency parsing," Proc. 17th Nordic Conf. Computational Linguistics NODALIDA, pp.219–222, 2009.
[14] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," Proc. Eighteenth International Conf. Machine Learning, pp.282–289, 2001.
[15] T. Joachims, "Estimating the generalization performance of an SVM

efficiently," Proc. Seventeenth International Conf. Machine Learning, pp.431–438, 2000.
[16] B. Taskar, C. Guestrin, and D. Koller, "Max-Margin markov networks," Proc. Neural Information Processing Systems Conf., 2003.
[17] Y. Tsuruoka, Y. Tateishi, J.D. Kim, T. Ohta, J. McNaught, S. Ananiadou, and J.I. Tsujii, "Developing a robust part-of-speech tagger for biomedical text," Advances in Informatics, pp.382–392, 2005.
[18] E. Riloff, J. Wiebe, and T. Wilson, "Learning subjective nouns using extraction pattern bootstrappingm," Proc. 7th Conf. Computational Natural Language Learning, pp.25–32, 2003.
[19] Y. Liu, Q. Tan, and K. Shen, "The word segmentation rules and automatic word segmentation methods for Chinese information Processing," QingHua University Press and GuangXi Science and Technology Press, 1994.
[20] L. Lam and C.Y. Suen, "Application of majority voting to pattern recognition: An analysis of its behavior and performance," IEEE Trans. Systems Man Cybern., vol.27, no.5, pp.553–568, 1997.
[21] K. Sagae and A. Lavie, "Parsing combination by reparsing," Proc. NAACL, pp.129–132, 2006.
[22] X.X. Li, J.P. Shen, X. Gao, and X. Wang, "Exploiting rich features for detecting hedges and their scope," Proc. Fourteenth Conference on Computational Natural Language Learning (CoNLL 2010): Shared Task, Uppsala, Sweden, Association for Computational Linguistics, pp.78–83, July 2010.
[23] M. Rei and T. Briscoe, "Combining manual rules and supervised learning for hedge cue and scope detection," Proc. Fourteenth Conference on Computational Natural Language Learning (CoNLL 2010): Shared Task, pp.56–63, Uppsala, Sweden, July 2010.
[24] E. Velldal, L. Øvrelid, and S. Oepen, "Resolving speculation: MaxEnt cue classification and dependency-based scope rules," Proc. Fourteenth Conference on Computational Natural Language Learning (CoNLL 2010): Shared Task, pp.48–55, Uppsala, Sweden, July 2010.

Huiwei Zhou was born in 1969. She is a lecturer in the Dalian University of Technology. Her main research interests include computational linguistics and data mining.

Xiaoyan Li was born in 1985. She is a graduate student. Her main research interests include computational linguistics and data mining for Biotechnology.

Degen Huang was born in 1965. He is a professor in the Dalian University of Technology. His main research interests include natural language processing, machine learning and machine translation. He is now working at the School of Computer Science and Technology, Dalian University of Technology. He is now a senior member of CCF, and an associate editor of Int. J. Advanced Intelligence.

Yuansheng Yang was born in 1946. He is a professor in the Dalian University of Technology. His main research interests include computer algorithms and natural language processing.

Fuji Ren Ph. D. and professor. His main research interests include Natural Language Processing, Knowledge Engineering, Sentience Computer, Machine Translation, Machine-Aided English Writing, Automatic Abstracting, Dialogue machine translation, Information Retrieval.