International Journal of Computer Science and Applications, Vol. 5, No. 2, pp 1- 11
©Technomathematics Research Foundation

BIOMEDICAL NAMED ENTITY RECOGNITION BASED ON CLASSIFIERS ENSEMBLE*

Haochang Wang[1,2] Tiejun Zhao[1] Hongye Tan[1] Shu Zhang[1]

{hcwang, tjzhao, hytan, zhangshu}@mtlab.hit.edu.cn

[1] MOE-MS Key Laboratory of Natural Language Processing and Speech in Harbin Institute of Technology, Harbin, 150001, China
[2] College of Computer and Information Technology, Daqing Petroleum Institute, Daqing, 163318, China

Abstract

In this paper, we present classifiers ensemble approaches for biomedical named entity recognition. Generalized Winnow, Conditional Random Fields, Support Vector Machine, and Maximum Entropy are combined through three different strategies. We demonstrate the effectiveness of classifiers ensemble strategies and compare its performances with standalone classifier systems. In the experiments on the JNLPBA 2004 evaluation data, our best system achieves an F-score of 77.57%, which is better than most state of the art systems. The experiment show that our proposed classifiers ensemble method especially the stacking method can lead to significant improvement in performances of biomedical named entity recognition.

Keywords: biomedical named entity recognition; classifiers ensemble; meta-learning; stacking; cascade generalization

1 Introduction

With the explosion of information in the biomedical domain, there is a strong demand for automated biomedical information extraction techniques. Recognizing the named entity (NE) such as proteins, DNAs, RNAs, cells etc. has become one of the most fundamental tasks in the biomedical knowledge discovery. While many algorithms have been proposed for this task, biomedical named entity recognition (NER) remains a challenging task and an active area of the research, as there is still a large gap about 10 points in the F-score between the best algorithms for biomedical named entity recognition and those for general newswire named entity recognition.

It is more difficult for biomedical NER in the following facts:

1) New NEs continue to be created, there does not exist a complete dictionary for most types of biomedical NEs.
2) The same word or phrase can refer to different entities depending upon their contexts. Conversely, many biological NEs have various spelling forms.
3) Some modifiers are often used before basic NEs, and sometimes biomedical

* This research is supported in part by the National High-Tech Research and Development Program projects 2006AA010108 and 2006AA01Z150.

NEs are very long. These factors highlight the difficulties for identifying the boundaries of NEs.

4) NEs may be cascaded. One NE may be embedded in another NE. More efforts must be made to identify this kind of NEs.

5) Abbreviations are frequently used in biomedical domain. Since there are few evidences in abbreviation for certain NE class, it is difficult to classify them correctly.

To tackle these challenges, it is necessary to explore effective methods and rich features.

There have been many attempts to develop techniques to identify NE in the biomedical literature. They roughly fall into three approaches, that is, heuristic rule-based approach, dictionary-based approach, and statistical machine learning-based approach. However, the state-of-the-art techniques for biomedical NER do not achieve satisfactory results. The problem suggests that individual biomedical NER system may not cover entity representations with enough rich features and no single type of algorithm is practical to achieve the best perform ance.

Classifiers ensemble has been a fruitful research direction in machine learning in recent years. It is an effective method for machine learning and can improve the classification performance of a standalone classifier. A combination aggregates the results of many classifiers, overcoming the possible local weakness of the individual classifier, producing a more robust recognition. In this paper, we conducted experiments with four different learning algorithms: Generalized Winnow, Conditional Random Fields (CRFs), Support Vector Machines (SVM), and Maximum Entropy (ME). We compared the performances of three different classifiers ensemble strategies: arbitration rules, stacked generalization (class-stacking and class-attribute-stacking), and cascade generalization.

We also explore various features for biomedical NER including local features, full text features, and external resources features. The experiments show that our system achieves promising performances.

The remaining part of this paper is organized as follows: Section 2 gives a short description to four classifiers: Generalized Winnow, CRFs, SVM, and ME which server as base classifiers. Section 3 introduces classifiers ensemble strategies. Feature selection is described in detail in section 4. The experimental setup and results are presented and discussed in section 5, and conclusion and future directions are drawn at last.

2 Classifiers

An important issue in classifiers ensemble is that the classifiers should not be strongly correlated in their "mis classification". This can be achieved by using different feature sets or different training sets to homogeneous classifiers, as well as using a different classification principle for each of the individual classifiers i.e. using heterogeneous classifiers. Heterogeneous classifiers usually use different assumptions about the structure of the data and the stochastic model that generates it. This leads to a different estimate of the posteriori probabilities. This paper deals with the heterogeneous classifiers ensemble strategy. We conducted experiments with four different types of classifiers.

Generalized Winnow: Winnow family of algorithms is particularly suitable for solving classification problems arising from Natural Language Processing (NLP) applications, due to their robustness to irrelevant features. Generalized Winnow was originally proposed by Zhang[1], and was used in the text chunking task. The basic idea is

to modify the original Winnow algorithm so that it solves a regularized optimization problem. The advantages of the Generalized Winnow comparing with the original Winnow are its ability to handle linearly non-separable data and its ability to provide reliable confidence estimates. Such confidence estimates are required in the statistical sequential modeling approach to the biomedical NER problem.

CRFs: CRFs are probabilistic frameworks for labeling and segmenting sequential data, which were firstly introduced by Lafferty et al.[2] The approach has achieved empirical success in many NLP problems. CRFs are undirected graphical models, trained to maximize the conditional probability of the output given the inputs. They have several advantages over both generative models like Hidden Markov Models (HMMs) and classifiers applied at each sequence position, including the ability to relax strong independence assumptions made in those models, and the ability to integrate a wide variety of arbitrary, non-independent features of the input.

SVM: SVM is a popular machine learning approach based on the structural risk minimization of statistical learning theory [3]. SVM is a kind of binary classifiers that search for an optimal separating hyper-plane between positive and negative samples and make decisions based on support vectors which are selected as the only effective examples in the training sets. The most attractive characteristics of SVM are the absence of local minima, the sparseness of the solution, and the use of the kernel-induced feature spaces. The SVM training process always seeks a global optimized solution and avoids over-fitting, so it has the ability to handle a large number of features and a relatively small dataset. We apply "Pairwise method" to extend SVM to a multi-class classifier.

ME: The ME framework is a powerful learning model, which has been successfully employed for many natural language processing tasks[4]. The ME principle seeks the distribution that maximizes the entropy of the distribution subject to the known constraints. The idea is to be "maximally noncommittal" about what we do not know, while still agreeing with what we do know. The advantage of ME is that it is robust and statistically efficient, while still allowing for easy representation and incorporation of different features.

3 Ensemble Approaches

The main aim for classifiers ensemble in our study is to improve recognition accuracy. Classifiers ensemble are effective methods for machine learning and can improve the classification performance of individual classifiers. There are several approaches for ensemble of classifiers. In this paper, three distinct strategies are tested to combine multiple predictions from separate base classifiers. The strategies are (1)arbitration rules, (2)stacked generalization including class-stacking and class-attribute-stacking, (3)cascade generalization.

The approach of arbitration rules uses heuristic rules to judge which prediction to be selected if the participating base learners cannot reach a consensus decision. Figure 1 depicts how the final prediction is made with input predictions of base learners using arbitration rule. Let x be an instance whose classification to be confirmed, and $C_k(x)$, $k = 1, 2, ..., K$ be the predicted classes of x from the Kth classifier models M_k, $k = 1, 2, ..., K$. We first compute the classification predicted by each of the single classifiers. The arbitration rule is then applied to judge which prediction is selected if the participating base classifiers cannot reach a consensus decision.

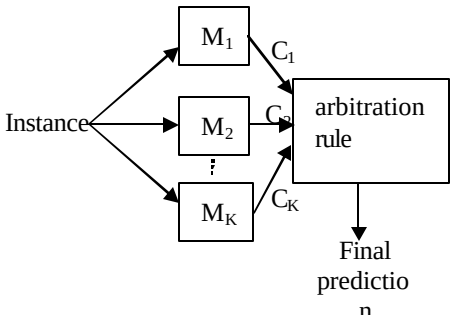

Figure 1. Classifiers ensemble using arbitration rule

Meta-learning is loosely defined as learning from the output of concept learning systems[5]. Meta-learning studies how a learning system can increase the efficiency through experiences, one common approach to meta-learning is known as stacked generalization also called stacking[6]. In stacking method the transformation of the training set conveys information about the predictions of the base-classifiers. After transforming the original training set, each example contains the original predictions of the base-classifiers, and may also contain the feature vectors.

The following notations are defined before introducing the details of the stacking strategy. Let x be an instance whose classification to be confirmed, and $C_k(x)$, $k = 1, 2, ..., K$ be the predicted classes of x from the Kth classifier models M_k, $k = 1, 2, ..., K$. $class(x)$ and $attrvec(x)$ denote the correct classification and attributes vector of example x, respectively.

Given a data set $D = \{(class(x_i), attrvec(x_i)), i = 1,\ldots,I\}$, we randomly split the data into J almost equal parts $D_1,\ldots,D_j$, and define D_j and $D^{(-j)} = D - D_j$ to be the test and training sets for the jth fold of a J-fold cross-validation.. Given K learning algorithms, which we call level-0 generalizers, the kth algorithm will be invoked on the data in the training set $D^{(-j)}$ to induce a classifier model $M_k^{(-j)}$, for $k = 1,\ldots,K$. These classifier models are called level-0 models. For each instance x_i in D_j, the test set for the jth cross-validation fold, let $C_{ki}(x_i)$ denote the prediction of the classifier model $M_k^{(-j)}$ on x_i.

At the end of the entire cross-validation process, the data set assembled from the outputs of the K classifier models is $D_{CV} = \{(class(x_i), C_{1i}(x_i),\ldots, C_{Ki}(x_i)), i = 1,\ldots,I\}$. D_{CV} is the level-1 data. In level-1, a learning algorithm, i.e. the generalizer is used to derive a model M_{Stack} from D_{CV}. Figure 2 illustrate the cross-validation process. To complete the training process, the final level-0 models M_k, $k = 1,\ldots,K$, are derived using all the data in D.

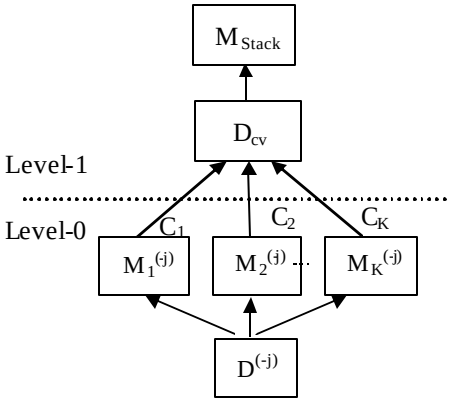

Figure 2. J-fold cross-validation process in level-0 and the level-1model produced process

Now let us consider the classification process, in which the models M_k, $k = 1,\ldots,K$, are used in conjunction with M_{Stack}. Given a new instance, models M_k , $k = 1,\ldots,K$, produce a vector $(C_1(x),\ldots, C_K(x))$. This vector is the input to the level-1 model M_{Stack}, whose output is the final classification result for that instance[7]. This classification process is shown in figure 3.

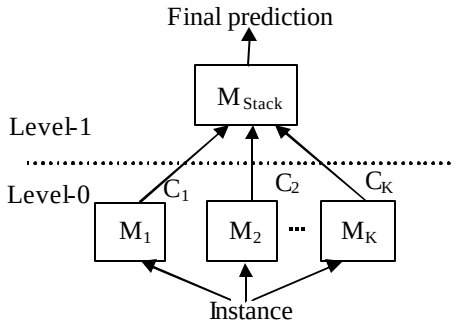

Figure 3. Classification process of stacking

For the training set, we have both the predictions of the base classifiers and the true class. The matrix containing the predictions of the base classifiers as predictors and corresponding true classes for training cases will be called the meta-data set. The classifier trained on this matrix will be called the meta-classifier.

We experiment with two stacking schemes which determined by the content of training examples for the meta-classifier.

Class-stacking The meta-level training instances consist of the correct classification and the predictions from base classifiers, i.e., $T = \{(class(x),C_1(x),C_2(x), ...,C_K(x)) \mid x \in D\}$.

Class-attribute-stacking The meta-level training instances consist of the correct classification and the predictions from base classifiers with the addition of the attribute

vectors, i.e., $T = \{(class(x), C_1(x), C_2(x), \ldots, C_K(x), attrvec(x)) \mid x \in D\}$.

Cascade generalization[8] is defined as combining the learning algorithms in sequence essentially. The meta-level training instances are obtained by adding the predicted classifications of classifiers to *attrvec(x)* in sequence, i.e., $T_k = \{(class(x), C_{k-1}(x), attrvec(x)) \mid x \in D\}$, $C_{k-1}(x)$ is obtained by level-$(K-2)$ classifiers which is shown in figure 3.

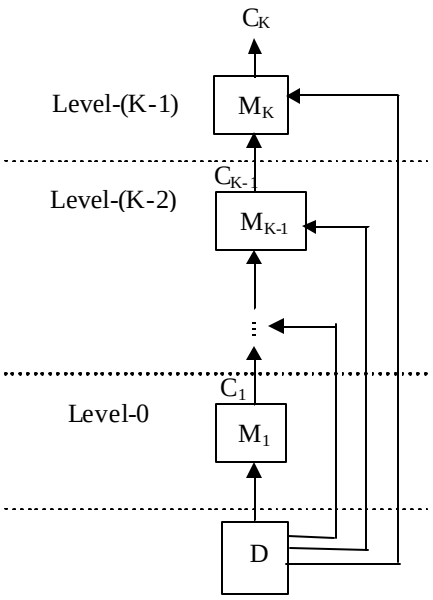

Figure 4. Classifiers ensemble using cascade generalization

4 Feature Selection

Feature selection is of utmost importance for the applications of statistical machine learning models. The main aim of selecting features is to find textual attributes that contribute to improving the recognition accuracy. In order to deal with the special phenomena in the biomedical texts, we make extensive use of a diverse set of features, including local features, full text features and external resources features. Here these types of features are described in detail and their effectiveness to the NER system is also discussed.

Local features are the immediate context of each word. We integrate variety of evidential local features including token features, unknown token features, orthographical features, word class features, POS features, shallow parsing features, key word features, frequency features, and N-gram features etc.

Unknown token features are binary features and state whether the current token was seen or not in the training data. In the training phase, we randomly set unknown token features.

Orthographical features: as many biomedical NEs contain numbers, upper case

characters, non-English characters and combinations of them, we define binary features for certain types of combinations. These features contribute a lot to performance enhancement despite they are simple surface clues . Generally, orthographical features are manually designed and aimed to group words by similar forms. They are likely to be served as indicators of unknown words and provide information to detect the boundaries of NEs. In our work, we manually designed orthographical features based on the characteristics of biomedical names.

Word class features: Certain kinds of named entities, which belong to the same class, are similar to each other. We introduce word class features as follows: first, given an token, capital letters, small letters, numbers and non-English characters are converted to "A", "a", "0" and "_" respectively; and then, consecutive same characters are squeezed into one character. This method will group similar names into the same NE class.

POS features: many biomedical NE are descriptive and very long. Therefore, POS may provide useful evidence about the boundaries of biomedical NE. General-purpose part-of-speech taggers do not usually perform well on biomedical texts because lexical characteristics of biomedical documents are considerably different from those of newspaper articles, which are often used as the training data for a general-purpose tagger. In our work, GENIA tagger2.0.2[9] is adopted which is trained not only on the general corpus but also on the GENIA corpus and the PennBioIE [10] corpus, so the tagger works well on various types of biomedical documents.

Shallow parsing features: we adopt GENIA tagger2.0.2 to get chunk information. Shallow parsing features may provide useful evidences about the boundaries of biomedical NEs.

Morphological features: Some prefixes and suffixes can provide good clues for classifying NEs. From our experience, the acceptable length for prefixes and suffixes is 1-4 characters. Experiments show that the performance of an NER system can be greatly enhanced with token prefix and suffix information. Such information can predict whether an unseen token looks like an entity-type or not. We also use more frequent prefix and suffix lists for feature extraction.

Key word features: a key word describes the function and characters of a compound, so the features provide large amount of information for NER. Furthermore, key words can be considered as the core of NE. This type of features provides important clues for discriminating NE classes.

Frequency features: while the information that a token was seen in a gazetteer is an unreliable indicator of whether it is a NE, less frequent words are less likely to be ambiguous than more frequent ones. Additionally, more frequent words are likely to be seen often in the training data and the system should be better at classifying them, while less frequent words are a common source of error and their classification is more likely to benefit from the use of external resources. We assigned each word in the training and testing data a frequency category corresponding to its frequency in the corpus.

N-gram features: we extract bi-gram and tri-gram from the training data. All the n-gram that uses frequencies over the pre-defined threshold are added as a kind of features in the statistic lists and used by the system.

Full text features are the context of the entire document. This type of features differs from the local features in that it cannot be derived from the token and its local environment. The system makes use of alias feature as one of full text features .The intuition is that relevant NE will be referred to in many ways throughout a given text. During the process of NER, the NEs already recognized from the previous sentences of

the document are stored in a list. When the system encounters an NE candidate, a name alias algorithm is invoked to first dynamically determine whether the NE candidate might be alias for a previously recognized NE in the list.

External resources features: using external resources features is based on the fact that local features and full text features can't provide sufficient evidence for confident recognition and classification. External resources were used to provide evidential clues. The features described here are mainly external gazetteers including common gazetteer, species names gazetteer, the list of chemical names endings, mineral name gazetteers. We also used the gene gazetteer, tissue name gazetteer, but the system performance dropped a little, so we removed these gazetteers.

5 Experimental Setup and Results

5.1 Individual Classifier Experiments

In our experiments, we evaluated our system on the JNLPBA 2004 public evaluation data sets[†]. In the JNLPBA 2004 shared task, the training data came from the GENIA Version 3.02 corpus which contains 2,000 MEDLINE abstracts. 404 MEDLINE abstracts from the GENIA project were used for the testing purpose. 5 classes of biomedical NE were tagged, that is, Protein, DNA, RNA, Cell_line and Cell_type[11]. Results are given as F-scores using a modified version of the CoNLL evaluation script and are defined as $F = (2PR)/(P + R)$, where P denotes Precision and R Recall. P is the ratio of the number of correctly found NE chunks to the number of found NE chunks, and R is the ratio of the number of correctly found NE chunks to the number of true NE chunks. The recognized NE is considered as right NE while every fragment composing NE has been correctly detected.

Table 1 shows the results of the four different individual classifiers on the test corpus. The CRFs model achieves the best recognition results. The ME runs faster than other three models, followed by Generalized Winnow. SVM model takes the most CPU time. As different individual classifiers have different sensitivity to different features and have different efficiency, we select different feature sets for these classifiers. Mostly local features and part of external resource features are applied to CRFs models; local features, full text features and part of external resources features are used by Generalized Winnow model; only token features are applied to SVM models as SVM have low efficiency; all features are applied to ME model. No post-processing is added to these models.

Table 1: Results of the four individual classifiers on the JNLPBA 2004 test data

Algorithm	P(%)	R(%)	F(%)
Generalized Winnow	67.99	72.48	70.16
CRFs	70.02	72.35	71.17
SVM	64.04	62.32	63.17
ME	65.12	71.19	68.02

5.2 Classifiers Ensemble Experiments

The performance of the standalone classifier was about 70% as mentioned above. Based on their performance we experiment with three different classifiers ensemble strategies:

[†] http://research.nii.ac.jp/~collier/workshops/JNLPBA04st.htm

(1)arbitration rules, (2)stacking generalization including class-stacking and class-attribute-stacking, (3)cascade generalization.

For arbitration rule ensemble strategy, we use the majority vote rule to combine the results from the Generalized Winnow, CRFs, SVM and ME models. If these four systems cannot reach a consensus decision, the preference is given the output class which has the highest confidence score. To evaluate the stacking method, we use Generalized Winnow, CRFs, SVM and ME as the base learning algorithms, which are all trained on the training set of the JNLPBA training data with 4-fold cross-validation, the 4-fold cross-validation is the internal operation of stacking as mentioned in section 3. At the meta-level, we experiment with CRFs model. In the cascade generalization approach, we combine SVM, CRFs, ME and Generalized Winnow models in sequence.

We analyze the results in more detail to see how the performance of the system improves through classifiers ensemble strategy. Experimental results are presented in table 2. CRFs classifier that yields the best classification result among the standalone classifiers is chosen as the baseline. All results from the ensemble strategies are significantly better than the baseline system. The implications of these experimental results are discussed as follows.

Arbitration rules ensemble strategy takes advantage of combination, and gets 73.18% F-score on the test data-set., but it can't significantly improve the system performance. Class-stacking approach achieves a little improvement than arbitration rule ensemble strategy because CRFs meta-classifier can automatically assign appropriate weights for standalone classifiers and use knowledge about how base classifiers behave with respect to each other. The most apparent outcome of these experiments is the superior performance of class-attribute-stacking ensemble strategy which achieves an F-score of 77.57%. Cascade generalization is also very effective and gets an F-score of 76.24%. We can explain that they can take advantage of all the evidences available and learn the relationship or correlation between individual classifiers predictions and the correct prediction to improve the performance of the system.

Table 2: Results of different classifiers ensemble system on the JNLPBA 2004 test data

ID	Experiment	P	R	F
1	arbitration rule	71.55	74.88	73.18
2	class-stacking	72.36	75.20	73.75
3	class-attribute- stacking	75.57	79.68	77.57
4	cascade meta-learning	74.76	78.42	76.24

Table 3: System performance comparison on the JNLPBA 2004 test data

System	P	R	F
Stacking	75.57	79.68	77.57
Zho[12]	69.4	76.0	72.6
Fin[13]	68.6	71.6	70.1
Set[14]	69.3	70.3	69.8

We compare our system with the top three systems from the competition of JNLPBA 2004 task in table 3. It is obvious that our system outperforms other systems by an increase in F-score of at least 5%.

6 Conclusion

In this paper, we evaluated techniques for classifiers ensemble strategy to recognize

biomedical NEs and showed some encouraging results. The best ensemble strategy achieves an F-score of 77.57% surpassing the best published results. The experiments results demonstrate that the classifiers ensemble strategy, especially the class-attribute-stacking method, is a suitable method for biomedical NER. Feature selection special for biomedical NER is another important factor to the success of our system.

In future work, we will further improve the accuracy and efficiency of our biomedical NER system. In addition we plan to find alternative methods for classifiers ensemble to improve the system performances.

References

1. Tong Zhang, Fred Damerau, David E. Johnson, Text chunking based on a generalization of Winnow[J], Journal of Machine Learning Research, 2002. (2): 615-637
2. J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data[A], In Proc. of ICML[C], 2001. 282-289.
3. B. Boser, I.Guyon, and V. Vapnik. An training algorithm for optimal margin classifiers. In Fifth Annual Workshop on Computational Learning Theory, pages 144--152, Pittsburgh, ACM. 1992.
4. Berger AL, Della Pietra SA, Della Pietra VJ. A maximum entropy approach to natural language processing. Computational Linguistics, 1996,22(1):39- 71.
5. A. L. Prodromidis, P. K Chan, S J. Stolfo. Meta-learning in distributed data mining systems: issues and approaches. Advances in distributed data mining. ? M? AAAI Press, Kargupta and Chan (eds.), 1999
6. D. Wolpert., Stacked generalization[J]. Neural Networks. 1992 , 5 (2) : 241-259
7. Ting K, Witten I. Issues in stacked generalization. Journal of Artificial Intelligence Research,1999,10:271~ 289
8. J. Gama, P. Brazdil. Cascade generalization[J]. Machine learning, 2000, 41(3):315-343.
9. Yoshimasa Tsuruoka, Yuka Tateishi, Jin-Dong Kim et al.. Developing a Robust Part-of-Speech Tagger for Biomedical Text[A]. Advances in Informatics - 10th Panhellenic Conference on Informatics[C], LNCS 3746, pp. 382-392, 2005
10. S. Kulick, A. Bies, M. Liberman et al.. Integrated Annotation for Biomedical Information Extraction[A]. HLT/NAACL 2004 Workshop: Biolink 2004[C], pp. 61-68.
11. KIM Jin-Dong,OHTA Tomoko,TSURUOKA Yoshimasa, et al. Introduction to the Bio-Entity Recognition Task at JNLPBA[A]. In Proceedings of the Joint Workshop on Natural Language Processing in Biomedicine and its Applications(JNLPBA-2004) [C], Geneva, Switzerland, 2004, 70-75.
12. GuoDong Zhou and Jian Su. Exploring Deep Knowledge Resources in Biomedical Name Recognition[A]. Proceedings of the Joint Workshop on Natural Language Processing in Biomedicine and its Applications(JNLPBA-2004) [C]. Geneva, Switzerland, 2004.

13. Jenny Finkel, Shipra Dingare, Huy Nguyen et al.. Exploiting Context for Biomedical Entity Recognition: From Syntax to the Web[A]. Proceedings of the Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA-2004) [C]. Geneva, Switzerland, 2004.
14. Burr Settles. Biomedical Named Entity Recognition Using Conditional Random Fields and Novel Feature Sets[A]. Proceedings of the Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA-2004) [C]. Geneva, Switzerland, 2004