Journal of Information Hiding and Multimedia Signal Processing ©2016 ISSN 2073-4212
Ubiquitous International Volume **7**, Number **4**, July **2016**

Biomedical Named Entity Recognition Based on Feature Selection and Word Representations

Ding-Xin Song, De-Gen Huang*, Li-Shuang Li and Hong-Lei He

School of Computer Science and Technology
Dalian University of Technology
No.2, Linggong Road, Hi-Tech Zone, Dalian, 116024, China
songdx@mail.dlut.edu.cn; huangdg@dlut.edu.cn; lilishuang314@163.com;lils@dlut.edu.cn
*:Corresponding author

Received November, 2015; revised Februrary, 2016

Abstract. *As an important task in biomedical text mining, biomedical named entity recognition (Bio-NER) has increasingly attracted researchers attention. Various methods have been employed to solve this problem and achieved desirable results on the annotated datasets. In this work, we focus on the feature set to reduce the training cost by feature selection and template optimization. Also, we integrate three kinds of word representation learnt in unsupervised way to summarize latent features. The experimental results show that feature selection, template optimization and word representation can promote the performance effectively. After post-processing, our methods achieve an F-score of 88.51% and perform better than most of the state-of-the-art systems.*
Keywords: Biomedical named entity recognition; Feature selection; Word representation.

1. **Introduction.** Biomedical named entity recognition (Bio-NER) is the prerequisite and key issue in the field of biomedical information extraction. Traditional research methods for Bio-NER can be mainly classified into three categories which are dictionary-based methods, rule-based methods and statistical machine learning methods [1]. Especially statistical machine learning methods have been the mainstream due to their better robustness and generalization. The used feature set is the most important factor for the success of machine learning. On the one hand, if the features are independent and correlate well with the class, the learning is easy. Conversely, it will be difficult and even impossible to learn. On the other hand, artificial features take most of the effort in a machine learning project [2]. Thus how to select features and integrate unsupervised features are important in named entity recognition.

Most researchers concern the learning algorithms and the features are designed artificially according to domain knowledge and experiences. The redundancy of features increases the calculation time and space complexity, and affects the performance. Langley et al. [3] indicated that sample complexity grows exponentially with the number of redundant features. Li et al. [4] applied SVM Recursive Feature Elimination (SVM-RFE) [5] which interacts with the SVM classifier to search the optimal feature set and is less computationally intensive than the subset selection method, to choose a subset of the most relevant features.

Recently, word representation is introduced efficiently as features into the task of NER. A word representation is often a vector associated with each word. Each dimensions value

corresponds to a feature. It is an effective way to learn word representation features with deep syntactic information from unlabeled corpora and explore them into NER system to improve the performance. Turian et al. [6] compared several word representations systematically on NER and chunking, and showed that each word representation could improve the accuracy. Kuksa et al. [7] introduced a semi-supervised method Word-Codebook Learning and obtained relative semantic information from the learnt word vectors. The experimental results showed that word representation obtained latent semantic information from unlabeled corpus and thus improved the performance.

In this paper, we explore feature selection and template optimization to reduce redundancy and improve the performance in Bio-NER, and then to further obtain deep information from unlabeled text and reduce the effort taking in artificial features. We introduce three kinds of word representations, that is, word embeddings, non-hierarchical clustering and hierarchical clustering in NER. Experiments result shows that our methods perform better than most of the state-of-art systems.

2. **Our Method.** A workflow to describe our system is shown in Fig.1.

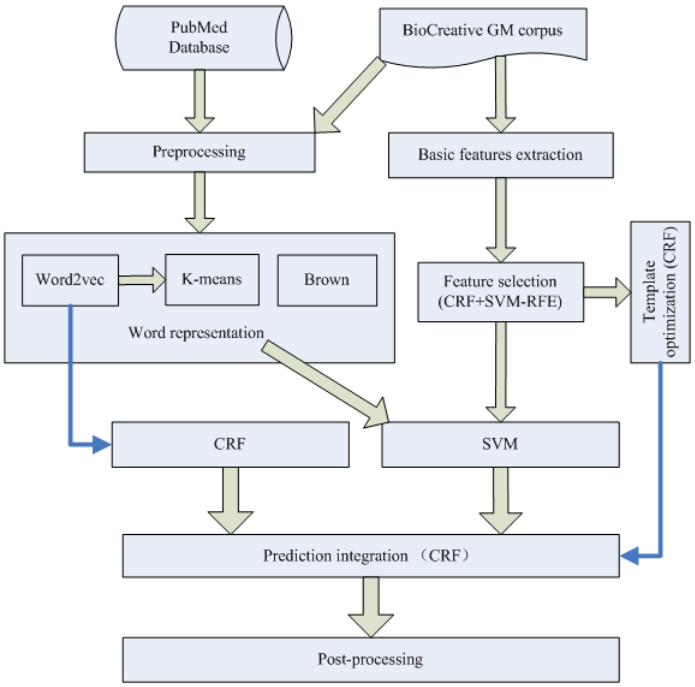

FIGURE 1. Overflow of our system

2.1. **System structure.** The corpora we used are PubMed Database and BioCreative GM corpus. Both of them are employed to train word representations, and BioCreative GM corpus is used to conduct the feature selection and template optimization experiments with SVM-RFE and CRF. In the step of integration, two different models, SVM and CRF, are trained to do the first prediction in which three kinds of word representations and selected basic features are exploited into SVM model while CRF model just adopting word embeddings. And then a new CRF model is trained by integrating two predicting results into selected three word representations, basic features and optimized template to do the final prediction. At last, the rule-based post-processing is done to improve the performance.

2.2. **Basic features.** According to the common feature in biomedical field, the local features used in this work are described as follows:

(1) Word features: the original word, the stemmed and Part-of-speech (POS) tag of the word where the latter two are obtained from GENIA tagger.

(2) Chunking: In general, biomedical named entities appear in noun phrase and often share the same critical section.

(3) Domain specific feature: It checks whether the current word is an ATCGU sequence, a nucleoside name, a nucleotide name, a nucleic acid, a short/long form of amino acid name or an amino acid with positional information.

(4) Morphological feature I: Digits in the current word are replaced by '*' and capital letters are lowercased. If the word has no digits, it is replaced by "no*" (e.g. CD4→cd*, cell→no*).

(5) Morphological feature II: Each character of the word is replaced by '-', except 5 vowels (i.e., a, e, i, o, u). For example, "cancer" is replaced by "-a--e-".

(6) Hot words: 250 words are extracted from training corpus as hot words according to their frequency.

(7) Word length: The length of the word which may be 1, 2, 3-5 or≥6.

(8) Word shape: To get the word shape of each word, capitalized characters are replaced by 'X', non-capitalized characters are replaced by 'x', digits are replaced by '1'.

(9) Brief word shape: We shorten consecutive strings appear in each word shape to one character to get the brief word shape for each word. E.g."xxxxXX"→"xX".

(10) Trigger words: Frequent words appear before the named entity in training corpus, e.g. "activate", "contain", "express".

(11) Prefix and suffix: 2-g, 3-g and 4-g prefix and suffix of the word, for example, the 2-g prefix and suffix feature of "Flur" is "Fl" and "ur".

(12) Orthographic feature: This feature is to get the word digital, capitalization, punctuation, and other characteristics.

2.3. **Feature selection.** SVM-RFE is a popular technique for feature selection which is based on heuristic search strategy and package-evaluation criteria. It uses the discriminate function of SVM as ranking criterion to sort features, and removes features with minimal impact on performance recursively according to sequential backward selection. Important features are reserved and thus optimal feature subset can be obtained.

In our work, all of the basic features are in initial features set. Then SVM-RFE is executed to sort features and remove redundant features. And the selected features are employed in the subsequent part.

2.4. **Template optimization.** Intuitively, context information can provide clues for Bio-NER. But which kind of contexts playing an important role is not explicit. Therefore it is necessary to select and optimize the context information. In this work, the context and joint information of each feature in optimal feature subset obtained by SVM-RFE in CRF template are optimized employing sequential forward selection method.

First, the context and joint information of each basic feature is added in CRF template successively. And then the performance is calculated after train and test. The context or joint information with the highest performance is selected and added in CRF template. This process is stopped until the performance decrease after adding certain information.

The context information of each feature in optimal subset is our candidate context while the joint of the word and other features in subset is our candidate joint information due to the importance of word in NER.

2.5. **Word representation.** Three kinds of word representation, including word embeddings, K-means and Brown clusters, are compared in the task of Bio-NER and described as follows.

(1) Word embeddings

Word embeddings have recently been proposed to address several NLP problems and have achieved great successes [8, 9, 10, 11]. Aiming at a much lower computational cost, Mikolov [9, 10, 11] recently developed the Word2Vec tool for computing continuous distributed representations of words. The tool implemented the Skip-gram model [9] expanded on the n-gram model architectures, which aimed to employ the current word to train other words which are in the fixed window before and after the current word in the same sentence.

We obtain word embeddings from large-scale unannotated corpus downloaded from PubMed. In our experiment, the dimension of vector is set to 50, 100, 200 and 400 respectively to evaluate the influence of dimension. The vectors of words with the same semantic or syntactic information in word embeddings is closer to each other, shown as Fig.2, which shows word embeddings can capture rich linguistic regularities.

Each word in the corpus is represented as a fixed-dimension vector, and each dimension of the word embeddings is adopted as an unsupervised word feature.

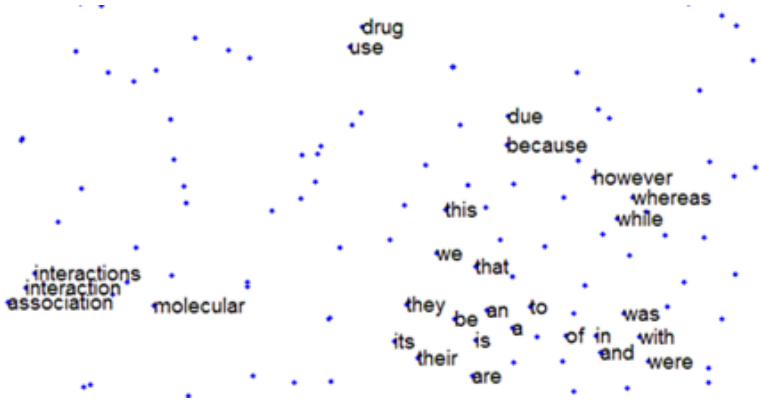

FIGURE 2. Word embeddings

(2) K-means

K-means algorithm, a non-hierarchical clustering algorithm will divide the n vectors x1, x2,. . . ,xn into k clusters, where k≤n. Different from word embeddings, the vector clustering representation puts the words that are close in the vector space into a cluster based on the K-means algorithm.

In this work, we employ K-means algorithm to perform clustering on the word vectors (the vectors in word embeddings). The max iteration C is set to 1000, and the number of clusters is set to 256, 512, 1024, 2048 and 4096. Thus each word gets an identifier from 1 to the number of clusters indicating its cluster which is viewed as additional unsupervised word feature.

(3) Brown clusters feature

Different from the non-hierarchical K-means algorithm, Brown algorithm is a classic hierarchical clustering algorithm based on n-gram language models. One shortage of Brown clusters is that it is based solely on bi-gram statistics, and does not consider words in a wider context. In this algorithm, each word gets its own cluster initially. Then, the algorithm repeatedly picks two clusters with the maximum mutual information and merges them into a single cluster until the number of the clusters is equal to the given parameter.

The result of Brown clustering algorithm is a binary tree, where each word occupies a leaf node, and where each leaf node contains a single word. A particular word can be assigned a binary string by following the traversal path from the root to its leaf, assigning 0 for each left branch, and 1 for each right branch.

The number of Brown clusters are set 32, 64, 128, 256 and 1024 respectively to select the optimal one. As the same as K-means, Brown clusters are viewed as additional unsupervised word feature.

2.6. **Post-processing.** In corpus, some punctuation marks such as parenthesis, brackets or double quotation marks are always paired. However, sometimes the machine learning models identify their first part as a part of a name entity without recognizing the second part, which causes mismatching. We adopt some rules to detect those mismatching marks and correct them.

Take "Trap(transposon- associated protein)" as example, which is a name entity in test set. "(" is identified successfully by the trained model, but ")" is failed. Thus, two wrong biomedical named entities "Trap" and "transposon- associated protein" are put into the result. After post-processing, "Trap(transposon-associated protein)" is identified correctly.

3. Experiment.

3.1. **Data and evaluation.** 5.33 million biomedical abstracts (5.6GB unannotated text) are downloaded from PubMed using protein as a keyword. This unlabeled text with BioCreative II GM corpus is used to train three kinds of word representation where BioCreative II GM corpus is supplied public by the evaluation task. The organization of this evaluation task supplies evaluate script and F-score, an popular evaluation criteria in information processing field, is used to evaluate performance. The formulation of F-score is as follows:

$$F\text{-}score = \frac{2 * P * R}{P + R} \tag{1}$$

where P is short for precision measuring the proportion of the correct identified names in recognized names and R represents Recall evaluating the degree of how many correct identified gene names in given results.

3.2. **Result of feature selection.** The used features are orderly word, stem, POS, chunking, domain specific feature, morphological feature I, morphological feature II, hot words, word length, word shape, brief word shape, trigger words, prefix-2, suffix-2,prefix-3, suffix-3,prefix-4, suffix-4,orthographic feature and are numbered 0 to 18. The order after conducting SVM-RFE is 0, 2, 4, 5, 8, 10, 13, 15, 14, 18, 17, 16, 12, 9, 6, 1, 3, 7 and 11. It shows that word, POS and morphological features play an important role in NER.

Features will be successively added to the feature subset according to the sorted order, and the variance of F-score is shown in Fig. 3. The F-score does not vary linearly with the increasement of the number of features. The F-score reaches the best until the 16th feature added with an F-score of 80.24% and is 0.33% higher than that using all features (19 basic features with F-score 79.91%) used. The number of the used features is reduced by 3. This shows that SVM-RFE can remove redundant features and further improves the performance.

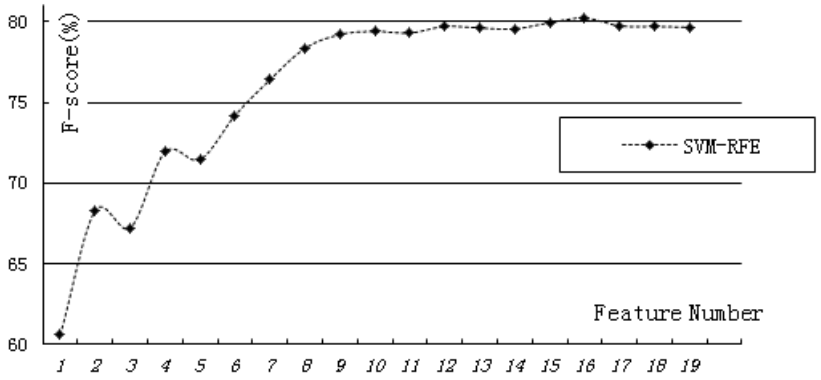

FIGURE 3. Feature selection of SVM-RFE

TABLE 1. The influence of separate context feature

Methods	P (%)	R (%)	F-score (%)
Baseline	82.62	78	80.24
Baseline+SCF of word	84.15	80.3	82.17
Baseline+SCF of stem	**84.26**	**80.56**	**82.36**
Baseline+SCF of POS	83.37	79.44	81.36
Baseline+SCF of domain specific	82.42	77.68	79.98
Baseline+SCF of morphological I	82.54	78.32	80.37
Baseline+SCF of morphological II	83.80	80.32	82.02
Baseline+SCF of word length	82.20	78	80.04
Baseline+SCF of word shape	83.06	79.68	81.33
Baseline+SCF simple word shape	82.97	79.86	81.38
Baseline+SCF of preffix-2	82.13	76.48	79.20
Baseline+SCF of suffix-2	82.73	79.36	81.01
Baseline+SCF of preffix-3	83.02	79.04	80.98
Baseline+SCF of suffix-3	83.07	80.88	81.96
Baseline+SCF of preffix-4	83.29	79.36	81.27
Baseline+SCF of suffix-4	82.40	80.16	81.26
Baseline+SCF of orthographic	82.56	77.68	80.04

3.3. **Result of template optimization.**

(1) The optimization of separate context feature

Based on feature selection, the separate context feature of each word is added, and the first iteration result is shown in Table 1 below.

The context information is added successively. In the first four iterations, the performance is improved gradually. But the performance decreases from the fifth iteration. And thus the forward sequence selection is stopped. The F-score of each iteration result is shown in Fig.4. The selected features include the context of stem, simple word shape, POS and the current word.

The F-score reaches 83.87% by template optimization and is improved by 3.63% than the original feature set with F-score 80.24%. 16 separate features, the context information of current word, stem, POS and simple word shape are adopted in this optimization.

(2) The optimization of joint information

Besides the separate context feature, we employ the combined context feature to improve the system performance. The first iteration results are shown in Table 2.

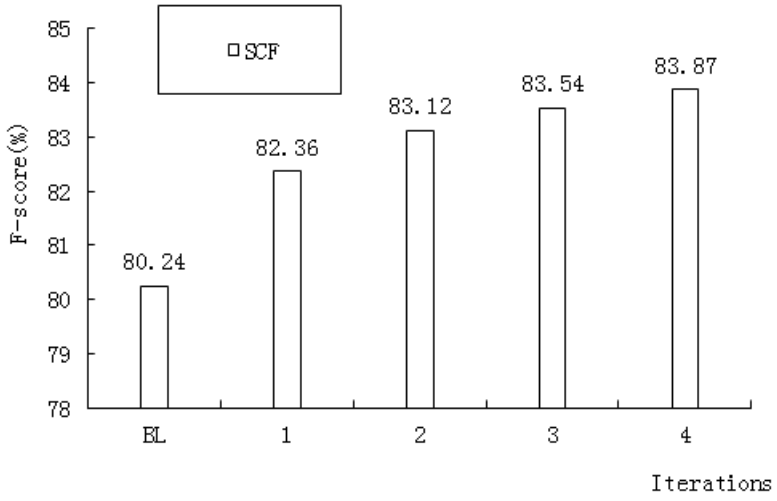

FIGURE 4. F-score of each iteration in SCF experiment

TABLE 2. The influence of the combined context feature

Methods	P (%)	R (%)	F-score (%)
Baseline	85.94	81.91	83.87
Baseline+CCF of word	**86.13**	**82.23**	**84.13**
Baseline+CCF of stem	86.07	82.08	84.02
Baseline+CCF of POS	85.98	82.11	84.00
Baseline+CCF of domain specific	84.45	78.75	81.51
Baseline+CCF of morphological I	83.67	79.56	81.56
Baseline+CCF of morphological II	83.96	81.73	82.82
Baseline+CCF of word length	83.41	79.34	81.32
Baseline+CCF of word shape	85.16	81.83	83.46
Baseline+CCF of simple word shape	84.87	82.17	83.49
Baseline+CCF of preffix-2	83.31	79.38	81.29
Baseline+CCF of suffix-2	84.23	80.56	82.35
Baseline+CCF of preffix-3	84.64	81.36	82.96
Baseline+CCF of suffix-3	84.98	81.87	83.39
Baseline+CCF of preffix-4	84.76	81.43	83.06
Baseline+CCF of suffix-4	83.92	81.15	82.51
Baseline+CCF of orthographic	83.67	79.23	81.38

The same iteration method is used as the separate context feature experiment. The forward sequence selection is stopped at the sixth iteration (the F-score of first five times is shown in Fig.5.), and the F-score is 84.45%, 4.21% higher than the initial score.

(3) The influence of optimization information

On the base of feature selection, we add the optimized template as Table 3, such as the before and after word of the current word, the POS of before and after word. The F-score added optimized template is improved by 5.36%.

3.4. **Word representation.**

(1) Experiment of three word representation methods

To investigate the influence of the dimension of word embeddings and the number

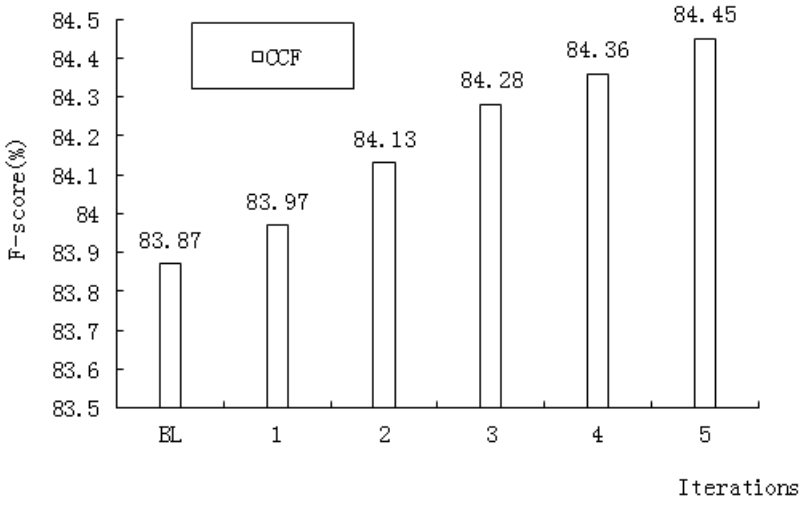

FIGURE 5. F-score of each iteration in CCF experiment

TABLE 3. The impact of feature selection and template optimization

Features	P (%)	R (%)	F-score (%)
Basic Features	85.30	78.13	81.56
Feature selection and template optimization	90.25	83.82	86.92

of clusters, we set several values and select the optimal one. The dimension of word embeddings is set to 50,100,200 and 400 respectively by Word2vec. The number of K-means cluster based on word embeddings is set to 256, 512, 1024, 2048 and 4096. In Brown cluster, the number of clusters is 32, 64, 128, 256, 512 and 1024. The results of different settings are shown in Table 4. It shows that the F-score do not rise with the increase of the dimension or the number of clusters. Finally the dimension of word embedding, the number of K-means clusters and Brown clusters are set to 50, 4096 and 64 respectively. Word embeddings lead to the decrease of F-score by 0.36% with precision decreased by 0.93%. However, the F-scores are increased by 0.39% and 0.2% respectively when K-means and Brown clusters added.

(2) The influence of combinations of word representation

We also combine the three word representation methods and the results are shown in Table 5 below. The F-score decreases when word embeddings is added, while it increases when the K-means clusters and Brown clusters are combined with basic features. However, the best performance is from the basic feature combined with K-means clusters only.

(3) The influence of context of word representation

We conduct several experiments to further explore the influence of K-means based word embeddings and Brown cluster in two different window sizes. When the current word, the two words before and after the current word are used, we define the window size is 5 which is similar to 3.As is shown in Table 6, it achieves the best with 87.64% F-score and is improved by 0.72% when K-means based word embeddings and Brown cluster of words in window size 5, the selected basic features and the its joint information are adopted.

In a word, K-means cluster based on word embeddings and Brown cluster loading rich semantic information and their combinations improve the performance, especially the latter. However, the performance decreases when adding word embeddings.

TABLE 4. The influence of dimension of word embeddings and the number of clusters

Methods	P (%)	R (%)	F-score (%)
Baseline	90.25	83.82	86.92
Baseline+WR50	89.32	83.96	**86.56**
Baseline+WR100	88.53	83.55	85.97
Baseline+WR200	88.16	83.17	85.59
Baseline+WR400	87.97	82.76	85.29
Baseline+WR-K256	89.96	84.41	87.09
Baseline+WR-K512	89.90	84.23	86.97
Baseline+WR-K1024	90.01	84.04	86.93
Baseline+WR-K2048	90.27	84.29	87.18
Baseline+WR-K4096	**90.45**	**84.39**	**87.31**
Baseline+B32	90.35	83.90	87.01
Baseline+B64	**90.61**	**83.88**	**87.12**
Baseline+B128	90.50	83.68	86.95
Baseline+B256	90.46	83.73	86.96
Baseline+B512	90.39	83.82	86.98
Baseline+B1024	90.50	83.73	86.98

TABLE 5. Comparison of three word representations' combinations

Methods	P (%)	R (%)	F-score (%)
Baseline+WR1+WR2	89.43	83.93	86.59
Baseline+WR1+WR3	89.48	83.88	86.59
Baseline+WR2+WR3	90.59	84.10	**87.23**
Baseline+WR1+WR2+WR3	89.74	83.77	86.65

TABLE 6. The influence of context of word representation

Methods	P (%)	R (%)	F-score (%)
BL+WR2+WR3	90.59	84.10	87.23
BL+WR2+WR3(3 word)	90.33	84.59	87.37
BL+WR2+WR3(5 word)	90.32	84.61	87.37
BL+WR2(3 word)+WR3	90.52	84.18	87.24
BL+WR2(3 word)+WR3(3 word)	90.26	84.53	87.30
BL+WR2(3 word)+WR3(5 word)	90.25	84.83	87.46
BL+WR2(5 word)+WR3	90.58	84.36	87.36
BL+WR2(5 word)+WR3(3 word)	90.52	84.67	87.50
BL+WR2(5 word)+WR3(5 word)	90.48	84.97	**87.64**

3.5. **Combination of SVM and CRF.**

(1) The impact of three word representations and their combinations on SVM model

The same experiment conditions of CRF are employed on SVM model again, the results are shown as Table 7. The F-score is 85.26% when basic features are adopted and it increases by integrating K-means cluster, Brown cluster and their combinations. The performance of basic feature combining both K-means cluster and Brown cluster reaches the best with F-score 86.58% improved by 1.32%.

(2) The impact of integrating predictions from CRF and SVM

TABLE 7. The impact of word representations and their combinations on SVM model

Methods	P (%)	R (%)	F-score (%)
YamCha (BL)	87.90	82.78	85.26
YamCha (BL+WR1)	88.48	77.93	82.87
YamCha (BL+WR2)	88.42	83.60	85.94
YamCha (BL+WR3)	88.82	83.74	86.21
YamCha (BL+WR1+WR2)	88.65	78.23	83.11
YamCha (BL+WR1+WR3)	89.27	79.27	83.97
YamCha (BL+WR2+WR3)	**89.15**	**84.15**	**86.58**
YamCha (BL+WR1+WR2+WR3)	89.11	79.38	83.96

To further improve the performance, we try to explore information from other models. We only exploit word embeddings with 50 dimension to construct another CRF model and evaluate it on test set. It achieves 75.10% F-score and demonstrates that word embeddings load rich information. And the prediction from this CRF model can be integrated into features. Meanwhile, the best prediction from SVM is also exploited. The result of exploring CRF and SVM predictions and post-processing is shown in Table 8 in which the F-score reaches 88.51%.

TABLE 8. The result of integrating predictions from CRF and SVM

Methods	P (%)	R (%)	F-score (%)
BL+WR2+WR3	90.48	84.97	87.64
BL+WR2+WR3+COM1+COM2	90.66	85.57	88.04
BL+WR2+WR3+COM1+COM2+PPk	91.11	86.05	**88.51**

3.6. **Comparison with other systems.** The best F-score achieved in the task of BioCreative II competition was 87.21% where semi-supervised structural method was adopted. Multi models based on rich features were adopted in the systems ranked the second and third in that task [12].

Li et al. [1] adopted the ensemble method and achieved better performance. Hsu et al. [13] exploited bi-direction parse model of CRF and dictionary filter and reached an F-score of 88.30%. Li et al. [14] proposed feature coupling generalization which could extract semantic information from unlabeled text and matched terms in dictionary combined with machine learning method. They achieved an F-score of 89.05%. Our method obtains an F-score of 88.51% without additional resources just adopting features relative with words and latent deep information. Although our method does not outperform Li et al. [14], several techniques such as multiple classifiers, dictionary matching and rich external resources were not used in our method.

4. **Discussion.** In this work, four aspects contribute to the improvement.

First, feature selection removes irrelative, redundant or weak influence features and reduces training cost. And thus it improves the performance by 0.33%.

Second, important context information in CRF template is optimized using sequential forward selection method and obtains an improvement of F-score by 5.36%.

Third, word representations, which are learnt in unsupervised way and can effectively measure the similarity between words, improves the performance by 0.72%. Especially it achieves an F-score of 75.10% when only word embeddings adopted and illustrates word

Table 9. Comparison with other systems

Methods	P (%)	R (%)	F-score (%)
BioCreativeII competition(best) (semi-supervised method combined with dictionary)	88.48	85.97	87.21
BioCreativeII competition(rank 2) (multi-classifiers combined with dictionary)	89.30	84.49	86.83
BioCreativeII competition (rank 3) (multi-classifiers)	84.93	88.28	86.57
Li [1](ensemble methods)	90.15	86.75	88.42
Hsu [13](CRF bi-direction parse model combined with dictionary)	88.95	87.65	88.30
Li [14](CRF combined with dictionary)	90.52	87.63	89.05
Our method(semi-supervised)	91.11	86.05	88.51

representations loads rich semantic and syntactic information.

The last but not least, prediction integration can utilize the predictive results efficiently which are learnt from different models with different feature sets and increases the F-score by 0.4%.

5. **Conclusion.** The proposed method improves the performance of Bio-NER, achieving an F-score of 88.51% and outperforming most of published works. Feature selection and template optimization help choose a better feature set. And word embeddings play an important role which implies a lot of syntactic and semantic information, and achieve acceptable performance. Finally, the integration of predictions from SVM and CRF further improves the performance.

Though the efforts in Bio-NER, there is room for improvement. In the future, more feature selection methods for different features should be researched. And deep learning for Bio-NER is another trend to further reduce the cost taking in feature engineering.

Acknowledgment. The authors gratefully acknowledge the financial support provided by the National Natural Science Foundation of China under No.61173101,61173100.

REFERENCES

[1] L. S. Li, W. T. Fan, D. G. Huang, Y. Z. Dang and J. Sun, Boosting performance of gene mention tagging system by hybrid methods, *Journal of biomedical informatics*, pp.156–164, 2012.

[2] P. Domingos, A few useful things to know about machine learning, *Communications of the ACM*, pp.78–87, 2012.

[3] P. Langley, *Selection of relevant features in machine learning*, Paloalto:Defense Technical Information Center, 1994.

[4] L. S. Li, L. K. Jin, J. Q. Zheng, P. P. Zhang and D. G. Huang, The Protein-Protein Interaction Extraction Based on Full Texts, *IEEE International Conference on Bioinformatics and Biomedicine*, Belfast, pp.493–496, 2014.

[5] I. Guyon, J. Weston, S. Barnhill and V. Vapnik, Gene selection for cancer classification using support vector machines, *Machine learning*, pp.389–422, 2002.

[6] J. Turian, L. Ratinov, and Y. Bengio, Word representations: A simple and general method for semi-supervised learning, *Proc. of the 48th Annual Meeting of the Association for Computational Linguistics*, Uppsala, Sweden, pp.384–394, 2010.

[7] P. P. Kuksa and Y. J. Qi, Semi-supervised bio-named entity recognition with word-codebook learning, *Proc. of the SIAM International Conference on Data Mining*, Columbus, pp.25–36, 2010.

[8] R. Collobert, J. Weston and L. Bottou (eds.), Natural language processing (almost) from scratch, *The Journal of Machine Learning Research*, pp.2493–2537, 2011.

[9] T. Mikolov, K. Chen, G. Corrado and J. Dean, Efficient estimation of word representations in vector space, *arXiv preprint arXiv*, pp.1301.3781, 2013.
[10] T. Mikolov, I. Sutskever and K. Chen (eds.), Distributed representations of words and phrases and their compositionality, *In Advances in Neural Information Processing Systems*, pp.3111–3119, 2013.
[11] T. Mikolov, W. Yih and G. Zweig, Linguistic regularities in continuous space word representations, *Proc. of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics*, Human Language Technologies, Atlanta, pp.746–751, 2013.
[12] L. Smith, L. K. Tanabe and R. J. Andor (eds.), Overview of BioCreative II gene mention recognition, *Genome biology*, vol.9, no.S2, pp.1–19, 2008.
[13] C. N. Hsu, Y. M. Chang and C. J. Kuo (eds.), Integrating high dimensional bi-directional parsing models for gene mention tagging, *Bioinformatics*, pp.286–294, 2008.
[14] Y. P. Li, H. F. Lin and Z. H. Yang, Incorporating rich background knowledge gene named entity classicisation and recognition, *BMC Bioinformatics*, vol.10, no.1, pp.233, 2009.