Recognizing Biomedical Named Entities Based on the Sentence Vector/Twin Word Embeddings Conditioned Bidirectional LSTM

Lishuang Li, Liuke Jin, Yuxin Jiang and Degen Huang

School of Computer Science and Technology,
Dalian University of Technology, Dalian 116024, Liaoning, China
{lils, Huangdg}@dlut.edu.cn
dllg_lkjin@mail.dlut.edu.cn,512415325@qq.com

Abstract. As a fundamental step in biomedical information extraction tasks, biomedical named entity recognition remains challenging. In recent years, the neural network has been applied on the entity recognition to avoid the complex hand-designed features, which are derived from various linguistic analyses. However, performance of the conventional neural network systems is always limited to exploiting long range dependencies in sentences. In this paper, we mainly adopt the bidirectional recurrent neural network with LSTM unit to identify biomedical entities, in which the twin word embeddings and sentence vector are added to rich input information. Therefore, the complex feature extraction can be skipped. In the testing phase, Viterbi algorithm is also used to filter the illogical label sequences. The experimental results conducted on the BioCreative II GM corpus show that our system can achieve an F-score of 88.61%, which outperforms CRF models using the complex hand-designed features and is 6.74% higher than RNNs.

Keywords: LSTM; twin word embeddings; sentence vector; Viterbi algorithm

1 Introduction

With the rapid development of computational and biological technology, biomedical literatures are growing exponentially, and abundant literatures about biomedical knowledge also provide an opportunity for text mining techniques in this field. As a fundamental step, the biomedical named entity recognition (Bio-NER) plays a critical role in many tasks such as coreference resolution and relation extraction in the biomedical field. Over the past years, though various methods have been proposed for Bio-NER, there is still a large gap on recognition performance between the biomedical and general field.

Currently, the most widely used methods to recognize biomedical named entity can focus on dictionary-based methods, rule-based methods and statistical machine learning methods [1]. Compared with the other two methods, the machine learning me-

thods are more robust and there is an advantage that they can identify the potential biomedical entities which are not previously included in standard dictionaries. There have been many attempts to develop machine learning techniques such as Hidden Markov Model (HMM) [2], Maximum Entropy (ME) [3], Conditional Random Field (CRF) [4], Support Vector Machine (SVM) [5] and etc.

However, these shallow machine learning methods are required to extract the manual features as the intermediate representation of each word in the text. Therefore, the recognition performance may be affected by some common drawbacks as followings. First, the construction of the feature set mainly relies on some experience and domain knowledge. Besides, selecting an optimal subset of features needs tremendous experiments. Furthermore, some complex features with syntactic information may be obtained from other NLP modules, like Part-of-Speech, and the inevitable cascading errors can lead to the final recognition errors. Meanwhile, enormous manual efforts may lead to over-design of the system and reduce the ability of generalization.

Aiming to overcome the problems described above, deep learning has been applied on the entity recognition in recent years. Collobert et al. [6] proposed a unified neural network architecture for various natural languages processing tasks which also achieved a better result in the NER task. Chen el al. [7] proposed deep belief network (DBN) to extract unsupervised and multi-level feature representation for entity recognition and classification, outperforming SVM, CRF and ANN classifiers. In order to integrate longer range of contextual effects and flexibly use the context information, Li et al. [8, 9] adopted the combined and extended recurrent neural networks (RNNs) which had better performance than CRF models with some simple features. However, some limitations still existed in their system. For example, the back propagated error in long sentence either blows up or decays exponentially so that long time lags are inaccessible in RNNs. Therefore, Long Short Term-Memory (LSTM) as a RNN architecture is motivated to deal with long range dependencies.

In this paper, we extend the bidirectional LSTM (BLSTM) on biomedical named entity recognition. Firstly, the twin word embeddings are used to rich input information. Then, the sentence vector can be obtained by calculating the differences of two embeddings to get the whole sentence information, which can accurately encodes the input information. Finally, in the testing phase, Viterbi algorithm is adopted to filter the illogical label sequences. The experimental results on the BioCreative II GM corpus show that our Sentence vector/Twin word embeddings conditioned BLSTM (ST-BLSTM) without any manual features can achieve an F-score of 88.61% which is better than (or close to) other state-of-the-art Bio-NER systems.

2 Methodology

We explore a so-called ST-BLSTM architecture, in which the twin word embeddings and sentence vector are introduced to the BLSTM. The system architecture for named entity recognition based on ST-BLSTM can be summarized in Fig. 1. Firstly, the word embeddings are obtained by lookup tables and the vectors in the word-context window are concatenated together to feed into the recurrent neural network. Then, we

establish a recurrent neural network with ST-BLSTM unit to acquire the hidden layer. And the recurrent connection is also added into the output layer to associate previous prediction probabilities. What's more, Viterbi algorithm is considered in the testing phase to further improve the recognition capability.

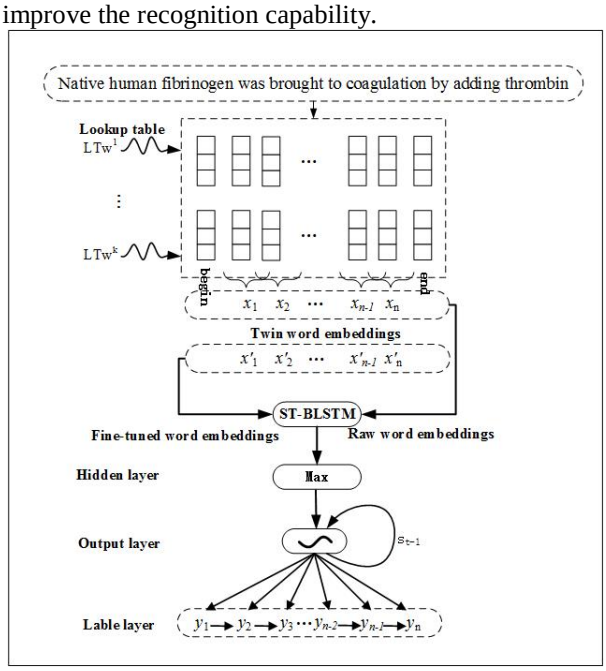

Fig. 1. Bio-NER architecture based on ST-BLSTM

2.1 LSTM

A standard architecture of LSTM mainly consists of an input layer, a recurrent LSTM layer and an output layer. Based on this structure, the input, output and stored information can be partially adjusted by the gates, which enhance the flexibility of the model. Such structures are more capable to learn a complex composition of word vectors than simple RNNs. While numerous LSTM variants have been described, here the forward pass for the LSTM model used in this paper is as follows:

$$i_t = \sigma(x_t \cdot w_{xh}^i + h_{t-1} \cdot w_{hh'}^i + b_h^i) \tag{1}$$

$$f_t = \sigma(x_t \cdot w_{xh}^f + h_{t-1} \cdot w_{hh'}^f + b_h^f) \tag{2}$$

$$o_t = \sigma(x_t \cdot w_{xh}^o + h_{t-1} \cdot w_{hh'}^o + b_h^o) \tag{3}$$

$$\tilde{c}_t = \tanh(x_t \cdot w_{xh}^c + h_{t-1} \cdot w_{hh'}^c + b_h^c) \tag{4}$$

$$c_t = i_t \odot \tilde{c}_t + f_t \odot c_{t-1} \tag{5}$$

$$h_t = o_t \odot \tanh(c_t) \tag{6}$$

Where σ denotes the logistic sigmoid function and $\odot$ denotes the element-wise multiplication. x is the input embeddings at time t, and i , f , o and c are respectively input gate, forget gate, output gate and the proposed values, all of which are the same size as the hidden vector h. W_{xh}, W_{hh} and b_h are the input connections, recurrent connections and bias values respectively. $\tilde{c}_t$ is the true cell value at time t. Intuitively, the forget gate controls the extent to which the previous memory cell is forgotten, the input gate controls what proportion of the current input to pass into the memory cell, and the output gate controls the exposure of the internal memory state. Therefore, the hidden vector from an LSTM unit is partial view of the unit's internal memory cell. Since the value of the gating variables varies for each vector element, the model can learn to represent information of long range dependencies.

2.2 BLSTM

One shortcoming of conventional RNNs is that they are only able to make use of the previous context. Bidirectional RNNs (BRNNs) [10] can do this by processing the data in both directions with two separate hidden layers, which are then fed forwards to the same output layer. In order to efficiently make use of the past features and future features, we construct bidirectional LSTM. Since there are no interactions between the two types of state neurons, the BLSTM network can be unfolded into a general feed forward network. In our implementation, we respectively do forward and backward for the whole sentences and reset the hidden states to random values at the beginning of each sentence.

2.3 ST-BLSTM

Since the original word embeddings are trained by the unsupervised learning approaches, the bias caused by the context definition may impact the quality of word embeddings. In this paper, fine tuning is added into the training process to retrain the word embeddings. Furthermore, we construct the twin word embeddings to rich the input information and the sentence vector is introduced to extend the neural network, whose memory cell is shown in Fig 2.

Twin Word Embeddings. The supervised fine-tuning process can further improve the performance of BLSTM and the retrained word embeddings can also be obtained in the fine-tuning process. The retrained word embeddings contain richer information associated with Bio-NER and the pre-trained word embeddings learned from large-scale unlabeled corpus obtain the potential feature information. In order to take into account the advantage of both feature information, we use two independent word embeddings to extend the BLSTM network. Since they share the same initial values,

we call them twin word embeddings. The only difference between them is that one kind of word embeddings is fine-tuned as parameter matrix, i.e. x'_t, while the other kind of word embeddings keeps constant over the whole process, i.e. x_t. In this work, we use the new LSTM architecture that is precisely specified below.

$$i_t = \sigma(x_t \cdot w^i_{xh} + x'_t \cdot w^i_{x'h} + h_{t-1} \cdot w^i_{hh'} + b^i_h) \tag{7}$$

$$f_t = \sigma(x_t \cdot w^f_{xh} + x'_t \cdot w^f_{x'h} + h_{t-1} \cdot w^f_{hh'} + b^f_h) \tag{8}$$

$$o_t = \sigma(x_t \cdot w^o_{xh} + x'_t \cdot w^o_{x'h} + h_{t-1} \cdot w^o_{hh'} + b^o_h) \tag{9}$$

$$\tilde{c}_t = \tanh(x_t \cdot w^c_{xh} + x'_t \cdot w^c_{x'h} + h_{t-1} \cdot w^c_{hh'} + b^o_h) \tag{10}$$

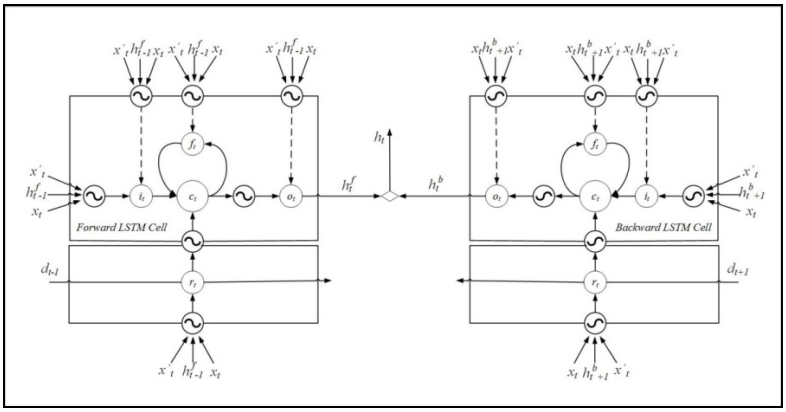

Fig. 2. Memory cell of ST-BLSTM

Sentence Vector. Since it is easy to ignore the implicit meaning of a sentence only using the word-level embeddings, the sentence-level feature representation applied into the hidden layer is considered in our system. Calculating the difference of the twin word embeddings, we can generate the sentence vector d_0 by averaging or maximizing all the word embeddings in the sentence. Besides, we use reading gate r_t to control what information should be retrained for future time steps. Then Equation (5) is modified so that the cell value c_t also depends on the sentence vector, which can accurately encode the input information.

$$r_t = \sigma(x_t \cdot w^r_{xh} + x'_t \cdot w^r_{x'h} + h_{t-1} \cdot w^r_{hh'} + b^r_h) \tag{11}$$

$$d_0 = \max(\sum_{t=1}^{T}(x'_t - x_t)) \tag{12}$$

$$d_t = r_t \odot d_{t-1} \tag{13}$$

$$c_t = i_t \odot \tilde{c}_t + f_t \odot c_{t-1} + \tanh(d_t) \tag{14}$$

2.4 Extension at the Output Layer

Considering the recurrent connection at the output layer can also improve the performance of recognition [8], the probability information from the previous state together with the result of the hidden layer are applied into the current prediction as Equation (15).

$$s_t = soft(h_t \cdot w_{hs} + s_{t-1} \cdot w_{ss'} + b_s) \tag{15}$$

$$soft(z_m) = e^{z_m} \Big/ \sum_k e^{z_k} , \tag{16}$$

where W_{hs} and W_{ss} are the weight matrices between the hidden layer and output layer, and between the previous output node and current output node, respectively. h_{t-1} represents the output values in the hidden layer from the previous time step, and s_t produces a probability distribution over labels. b_s represents the bias of each layer.

2.5 Training

All the neural network models used in this paper are trained by treating each sentence as a mini-batch. The objective function is the cross entropy error between the predicted probability $\boldsymbol{p_i}$ and the actual label vector $\boldsymbol{y_i}$ as Equation (17). The forward and backward networks in the BLSTM are structured to share the same set of word embeddings. Adadelta [11] is used for gradient descent and optimizing the parameters. Besides, dropout [12] is adopted in our experiments to address the overfitting problem.

$$loss = -\frac{1}{n}\sum_{i=1}^{k} y_i \log p_i \tag{17}$$

2.6 Viterbi Algorithm in the Testing Phase

During the testing phase, Viterbi algorithm is executed to make sure that the illogical label sequence will not be selected. Since the generated label y_i does not involve the label y_{i-1} before it, the illogical label chains maybe exist in the prediction result. For example, it is obvious that the label **I** should not follow the label **O**. In this paper, we use the similar method as shown in Chen et al.'s [13], the initial probabilities of illogical entity label is assigned 0, while the others reset to 1. And the transition probabilities of the illogical entity label path should be 0. Thus, the partial probabilities of path containing "O I" will be 0 and this path will be discarded.

3 Experiments

Our experiments are carried out on three different datasets including the BioCreative II GM, JNLPBA2004 and BioCreative V DNER. Firstly, we experimentally demon-

strate that the improvements based on our ST-BLSTM are effective on the BioCreative II GM corpus. Then, the comparison with other approaches is conducted on the three corpora. In experiments, all the deep networks are based on the common Theano neural network toolkit [1] and the RNN models are trained with the same hyper-parameters. All the experiments are based on a set of 200 dimensional word embeddings. Besides, we use F-score as our assessing criteria to evaluate our method. The definition of Precision (*P*), Recall (*R*) and F-score (*F*) are shown as Equation (18-20). *TP* is short for true positives, *FP* represents false positives, and *FN* stands for false negatives.

$$P = TP/(TP + FP) \tag{18}$$

$$R = TP/(TP + FN) \tag{19}$$

$$F - score = 2 * P * R/(P + R) \tag{20}$$

3.1 Data Set

Table 1. Three biomedical datasets

Corpus	Training set	Development set	Test set
BioCreative II GM (sentence)	15000	-	5000
JNLPBA2004 (abstract)	2000	-	404
BioCreative V DNER (abstract)	500	500	500

We test our system on three biomedical datasets as shown in Table 1. And BioCreative II GM is mainly composed of sentences, while JNLPBA2004 and BioCreative V DNER give the abstracts. Table 1 lists the size of sentences or abstracts for training, development and test sets, respectively. ***BILOU*** tagging scheme is selected to find the entity boundary in our experiment. ***B*** refers to the beginning word of a gene name, ***I*** and ***L*** respectively indicate inside tokens and the last token in a gene name if it contains more than one word, ***O*** refers to the words which are not included in a gene name, and finally ***U*** represents the unit-length chunks.

3.2 BioCreative II GM Corpus

The Results of Improvements Based on ST-BLSTM.
Our experiments are carried out on the BioCreative II GM corpus and the models' performance is reported in Table 2. The effects of improvements are analyzed as followings.

[1] http://deeplearning.net/tutorial/rnnslu.html

Table 2. Results on GM corpus with different improved approaches based on LSTM

Model	PWE	RC	Viterbi	P (%)	R (%)	F- score (%)
LSTM				83.19	73.40	77.98
LSTM	√			85.63	82.13	83.85
LSTM	√	√		85.88	83.03	84.43
BLSTM	√	√		88.68	85.76	87.20
T-BLSTM	√	√		89.83	87.07	88.43
ST-BLSTM	√	√		89.48	87.63	88.54
ST-BLSTM	√	√	√	**89.54**	**87.69**	**88.61**

Pre-trained Word Embeddings (PWE). In order to explore the impact of richer text information on LSTM architecture, we use two ways to initialize the word embedding: random and pre-trained. And the results reveal that the pre-trained word embeddings have better performance than the random word embeddings by rising 5.87% F-score (77.98% vs 83.85%).

Recurrent Connection (RC) at the output layer. Recurrent connection at the output layer can take advantage of the previous probabilistic information of labels and apply it into the calculation of current prediction. Thus, the potential links between labels can be considered to further improve the performance. The experimental results show that the F-score can increase from 83.85% to 84.43%.

BLSTM. In order to efficiently make use of the past and future features, bidirectional LSTM networks are trained in our work. From Table 2, we can see that the BLSTM can have better performance which rises 2.77% compared with the unidirectional LSTM.

Twin Word Embeddings. Based on the BLSTM, the twin word embeddings are added to rich the input information and the F-score reaches 88.43%. We can improve the performance by 1.23%.

Sentence Vector. Sentence vector is also combined with our BLSTM, which is generated by maximizing the difference of twin word embeddings in a sentence. The F-score reaches 88.54% which rises by 0.11%.

Viterbi Algorithm. In the testing phase, we also use Viterbi algorithm to filter illogical sequence of labels. At last, the best result from our architecture can be increased to 88.61%.

Comparison with Existing Systems.

We make the comparisons between our system and some state-of-the-art works in Table 3. As the best system in competition at that time, Ando [14] mainly used a semi-supervised learning method, combined classifiers with dictionary as well as the post-processing, the final F-score reached 87.21%. In Li et al.'s system [15], they extracted

rich hand-designed features such as part-of-speech, stemmed word, orthographic feature etc. and unigram, bigram, trigram types of features based on CRF model as well as the post-processing achieving an F-score of 87.28 %. In Li et al.'s method [16], they increased three kinds of distributed word representation besides the rich hand-designed features, and used the combined methods to reach a better F-score of 88.44%. However, in our approach the complex hand-designed features and domain dictionary knowledge are skipped as well as 0.17% F-score higher compared with Li et al.'s [16].

Table 3. Comparison with other Bio-NER systems

Model	P (%)	R (%)	F-score (%)
Ando et al.[14]	88.48	85.97	87.21
Li et al.[15]	90.38	84.39	87.28
Li et al.[16]	91.24	85.80	88.44
Li et al.[17]	**90.52**	**87.63**	**89.05**
Li et al.[8]	80.93	82.21	81.87
ours	**89.54**	**87.69**	**88.61**

In Li et al.'s system [8], the conventional RNNs are adopted to Bio-NER task and their best performance is 81.87% F-score which is 6.74% lower than our method. It demonstrates that our model outperforms the conventional RNN.

Though our system can outperform most of the shallow approaches, compared with Li et al.'s system [17] which performs the best until now, our F-score is 0.44% lower. The reason account for lag is that they utilized the abundant external resources to construct the dictionary, rich domain knowledge and hand-designed features.

3.3 JNLPBA2004 Corpus

Table 4. Comparison with other systems on the JNLPBA2004 corpus

Model	P (%)	R (%)	F-score (%)
Yao et al.[18]	76.13	66.54	71.01
Chang et al. [19]	-	-	71.85
Wang et al.[20]	-	-	72.23
Zhou and Su et al.[21]	75.99	69.42	72.55
ours	**74.77**	**70.85**	**72.76**

Table 4 lists the comparison with other systems on the JNLPBA2004 corpus. Yao et al. [18] used a multi-layer neural network to continuously learn the representation of features, achieving 71.01% F-score. Chang et al. [19] used some hand-designed features and word embeddings as the input of CRF model as well as the post-processing; they achieved 71.85% F-score. Wang et al. [20] verified that the Gimli method based on CRF model could achieve the best performance with 72.23% F-score among six different Bio-NER methods on JNLPBA2004 corpus. Besides, as the best system in competition at that time, Zhou and Su et al.'s method got 72.55% F-score [21]. The abundant resources knowledge and common hand-designed features,

such as abbreviation, alias and dictionary, were used, which greatly enhanced its performance. However, the experimental results show that the F-score of our ST-BLSTM model can reach 72.76% which outperforms all of them by 0.91%, 1.75%, 0.53% and 0.21%, respectively. Meanwhile, no hand-designed features and rules are used in our system.

3.4 **BioCreative V DNER Corpus**

We also apply our system on the BioCreative V DNER corpus. Table 5 shows the comparison result with CRF model. In the case of evaluating the test set, we combine training set and development set as the training set. The CRF model needs to extract the hand-designed features such as part-of-speech, stemmed word, orthographic feature etc. As shown in Table 5, our method can reach 78.91% F-score on the development set, which is 2.5% higher than CRF. And CRF achieves 76.91% on the test set, while our method is 5.69% higher than CRF instead of any manual features. For example, in development set, the standard entity "Tricuspid valve regurgitation" is recognized by our method, while the CRF model could not recognize it. The main reason is that the neural network can learn more potential characteristic information, and train more complex models; however, the shallow machine learning methods have strong dependency on the artificial features and hard to represent the complex models. Therefore our method can achieve a better result in the NER task.

Table 5. Results on the Biocreative V corpus about disease recognition

Model	Data set	P (%)	R (%)	F-score (%)
CRF	development set	71.41	82.16	76.41
ours		76.67	81.28	**78.91**
CRF	test set	72.78	81.54	76.91
ours		81.53	83.70	**82.60**

4 Discussion

From the above experimental results, we can conclude that our ST-BLSTM model outperforms most state-of-the-art Bio-NER systems and mainly includes the following important advantages:

No Hand-designed Features. We skip the step of extracting complex hand-designed features, and replace it with word embeddings trained off-line. Since high-quality word embeddings can catch a large number of precise syntactic and semantic word relationships, the deep learning architecture can fully utilize this information and extract the high-level features for the Bio-NER.

Additional Extension at the Output Layer. Considering that predicted result (i.e. probability of labeling) from the prior node can have an important impact on the current prediction, we extend the original LSTM model by adding a reconnection at

the output layer. From the experimental results, we can see that the extended method can produce positive impact on the Bio-NER.

Combining Twin Word Embeddings and Sentence Vector. Considering the fine-tuned word embeddings contain richer information associated with Bio-NER, and the pre-trained word embeddings contain the feature information learning from large-scale unlabeled corpus, we extend the bidirectional LSTM by adding twin word embeddings. For the input, the extended features are more abundant, and the multiplication gates can control more accurate information. Besides, the sentence vector could contain complementary information of twin word embeddings. The experimental results show that both twin word embeddings and sentence vector could have positive effects on the BLSTM architecture to recognize biomedical named entities.

Viterbi Algorithm in the Testing Phase. The results show that the added Viterbi algorithm in bidirectional LSTM output layer can filter the illogical label sequences effectively. This is mainly because that the algorithm is based on dynamic programming and can find the most likely label sequences.

5 Conclusion

In this paper, we propose ST-BLSTM architecture to identify biomedical entities. The twin word embeddings and sentence vector are added into the bidirectional LSTM to obtain more abundant contextual information. Simultaneously, we extend the model by adding recurrent connection at the output layer and in the testing phase the Viterbi algorithm is applied to filter the illogical label sequences. The experimental results show that our model on BioCreative II GM corpus can achieve 88.61% F-score without using any hand-designed features and external resource, higher than almost all systems. And on JNLPBA2004 and BioCreative V DNER datasets, we also can achieve a rather better recognition performance.

Acknowledgment. The authors gratefully acknowledge the financial support provided by the National Natural Science Foundation of China under No. 61173101, 61173100. The Tesla K40 used for this research was donated by the NVIDIA Corporation.

References

1. Li, L., Fan, W., Huang, D., Dang, Y., Sun, J.: Boosting Performance of Gene Mention Tagging System by Hybrid Methods. Journal of Biomedical Informatics. 45(1), 156-164 (2012)
2. Shen, D., Zhang, J., Zhou, G., Su, J., Tan, C.: Effective Adaptation of a Hidden Markov Model-based Named Entity Recognizer for Biomedical Domain. In: Proceedings of the ACL 2003 Workshop on Natural Language Processing in Biomedicine. vol. 13, pp.49-56. (2003)

3. Saha, S., Sarkar, S., Mitra, P.: Feature Selection Techniques for Maximum Entropy Based Biomedical Named Entity Recognition. Journal of Biomedical Informatics. 42(5), 905-911 (2009)
4. Sun, C., Guan, Y., Wang, X., Lin, L.: Rich Features Based Conditional Random Fields for Biological Named Entities Recognition. Computers in Biology and Medicine. 37(9), 1327-1333 (2007)
5. Lee, K., Hwang, Y., Kim, S., Rim, H.: Biomedical Named Entity Recognition Using Two-phase Model Based on SVMs. Journal of Biomedical Informatics. 37(6), 436-447 (2004)
6. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu K., Kuksa, P.: Natural Language Processing (almost) from Scratch. Journal of Machine Learning Research. 12(8), 2493-2537 (2011)
7. Chen, Y., Zheng, D., Zhao, T.: Exploring Deep Belief Nets to Detect and Categorize Chinese Entities. In: International Conference on Advanced Data Mining and Applications. pp. 468-480. (2013)
8. Li, L., Jin, L., Huang, D.: Exploring Recurrent Neural Networks to Detect Named Entities from Biomedical Text. In: Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data. pp. 279-290 (2015)
9. Li, L., Jin, L., Jiang, Z., Song D., Huang, D.: Biomedical Named Entity Recognition Based on Extended Recurrent Neural Networks. In: IEEE International Conference on Bioinformatics and Biomedicine. pp. 649-652 (2015)
10. Schuster, M., Paliwal, K.: Bidirectional Recurrent Neural Networks. IEEE Transactions on Signal Processing. 45(11), 2673-2681 (1997)
11. Zeiler, M.D.: ADADELTA: An Adaptive Learning Rate Method. arXiv Preprint arXiv. 1212. 5701 (2012).
12. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research. 15(1), 1929-1958 (2014)
13. Chen, Y., Zheng, D., Zhao, T.: Exploring Deep Belief Nets to Detect and Categorize Chinese Entities. In: International Conference on Advanced Data Mining and Applications. pp. 468-480. (2013)
14. Ando, R. K.: BioCreative II Gene Mention Tagging System at IBM Watson. In: Proceedings of the Second BioCreative Challenge Evaluation Workshop. vol. 23. pp. 101-103 (2007)
15. Li, L., Zhou, R., Huang D., Liao, W.: Integrating Divergent Models for Gene Mention Tagging. In: IEEE International Conference on Bioinformatics and Biomedicine. pp.1-7 (2009)
16. Li, L., He, H., Liu, S., Huang, D.: Research of Word Representations on Biomedical Named Entity Recognition. Journal of Chinese Computer Systems. 2, 302-307 (2016)(in Chinese)
17. Li, Y., Lin, H., Yang, Z.: Incorporating Rich Background Knowledge for Gene Named Entity Classification and Recognition. BMC Bioinformatics. 10(1) 1-15 (2009)
18. Yao, L., Liu, H., Liu, Y., Li, X., Anwar, M. W.: Biomedical Named Entity Recognition based on Deep Neutral Network. Corpus, 8(8), 279–288 (2015)
19. Chang, F., Guo, J., Xu, W., Chung, S.: Application of Word Embeddings in Biomedical Named Entity Recognition Tasks. Journal of Digital Information Management. 13(5), 321–327 (2015)
20. Wang, X., Yang, C., Guan, R.: A Comparative Study for Biomedical Named Entity Recognition. International Journal of Machine Learning & Cybernetics. 1-10 (2015)
21. Zhou, G. Su, J.: Exploring Deep Knowledge Resources in Biomedical Name Recognition. In: International Joint Workshop on Natural Language Processing in Biomedicine and ITS Applications. pp. 96–99 (2004)