International Journal of Hybrid Information Technology
Vol.8, No.8 (2015), pp.279-288
http://dx.doi.org/10.14257/ijhit.2015.8.8.29

Biomedical Named Entity Recognition based on Deep Neutral Network

Lin Yao[1,2,3], Hong Liu[1], Yi Liu[2], Xinxin Li[4] and Muhammad Waqas Anwar[5]

[1]School of Electronics Engineering and Computer Science, Peking University
[2]Pku-hkust Shenzhen-hongkong Institution
[3]School of Software, HIT
[4]Computer Science Department, HITSGS
[5]Department of Computer Science, COMSATS Institute of Information Technology
yaolin@insun.hit.edu.cn

Abstract

Many machine learning methods have been applied on the biomedical named entity recognition and achieve good results on GENIA corpus. However most of those methods reply on the feature engineering which is labor-intensive. In this paper， huge potential feature information represented as word vectors are generated by neutral networks based on unlabeled biomedical text files. We propose a Biomedical Named Entity Recognition (Bio-NER) method based on deep neural network architecture which has multiple layers and each layer abstracts features based upon the features generated by lower layers. Our system achieved F-score 71.01% on GENIA regular test corpus , F-score values for 5-fold cross-validation is 71.01% and this result is closed to the state-of-the-art performance with only POS (Part-of-speech) feature and represents the deep learning can effectively performed on biomedical NER.

Keywords: *Deep learning, Biomedical named entity recognition, Neutral networks*

1. Introduction

In the biomedical field, Gigabyte even Terabyte level data has been produced each day. Such huge data drives the development of the biomedical area research in so many ways. However, the ability of biomedical researchers to analyze, and apply biomedical Big Data is often limited. Relevant software and tools for biomedical information processing could enhance the use of those huge Data. Biomedical NER is a crucial initial step for biomedical information processing. And it is fundamental technology for identifying entities and their interactions. But the biomedical NER is more difficult than general NER because of the complex situations such as irregular expression, hardly distinguished boundaries and daily changing group members. The difficulty and potential importance of this task attract many researchers [1-5].

Many supervised learning techniques have been used to address biomedical NER problem, such as HMM (Hidden Markov Models)[6], MEMMs (Maximum Entropy Markov Models) [7, 8], SVM (Support Vector Machines)[9, 10], and CRF (Conditional Random Field) [11-13]. CRF is applied to address entity recognition in biomedical by Settles [9]. The method achieves an F-score of 69.9% on GENIA corpus with only several kinds of features. The HMM is carried on the GENIA corpus and achieves a precision of 66.5% and a recall of 66.6% [9]. **SVM is** applied by Ki-Joong Lee to present a two-phase named entity recognizer on the GENIA corpus and get an F-score of 74.8% for the boundary identification and an F-Score of 66.7% for the semantic classification [10]. Skip-chain conditional random fields (CRFs) model has been applied on this task by considering long-range dependencies. This approach achieves F-score of 73.2% on GENIA corpus[14]. Li presents a two phases Bio-NER model on GENIA corpus. They divided the task into two steps: named entity detection(NED) and named entity classification (NEC)[15]. The first step is to distinguish non named-entities (NNEs) but not identifying their types. Six classifiers are constructed in this phrase. In the NEC phrase, the multi agent strategy is used and they achieve an F-score of 76.06 %. However the previous methods are either based on specific hand-crafted feature templates or a stacking method integrated with different training methods. The building **o**f feature templates is completely empirical processes and requires many trial works.

Motivated by the works of Collobert [16] , we construct a neural network model for biomedical named entity recognition task. Our works present that the deep learning can effectively be performed on biomedical NER. Our architecture achieved close to state-of-the-art performance on GENIA corpus which is a popular standard corpus has been adopted by many research groups as assessment.

This article is organized as follows. Section 2 describes the proposed architecture based on deep Neutral Network. In Section 3, we present our performance followed by our analysis. Conclusions are made in Section 4.

2. Architecture

Our method is based on a CNN (convolutional neural network), which has been successful adopted in some NLP (Natural Language Processing) tasks [17-19]. The versatile convolutional neural network architectures were proposed by Bengio [17] for probabilistic language model. Since then NNs were reintroduced later for multiple NLP tasks. We choose it for Bio-NER task. The neural network architecture is shown in Figure1. Many traditional tagging approaches need to choose different features depending on different tasks regarding a lot of prior and professional knowledge. The feature engineering is important but manual and labor-intensive. Compared with other over-engineering system, the deep learning approach reduces dependency on linguistic ingenuity. Figure 1 describes the word beta which is in the right middle of sliding window is evaluated at moment t. The input for our neural network model is the words within the sliding windows represented as real-valued vectors. After transformation of linear layers and sigmoid layer, the node score for each label of word beta are generated. As shown in the last procedure of Figure1, finally a score lattice for a sentence is output. The nodes of each column indicate the label scores at particular time. The labels transition scores are listed on the edge. Viterbi algorithm is applied to obtain the optimal label sequence.

2.1. Extracting Word Feature Vectors

The input length of CNN is fixed and need to be adapted to text data. As shown in Figure 1, first, a word dictionary W will be constructed using huge raw data from biomedical papers. Each word in the dictionary is represented by a fixed-dimension vector. The words are transformed into the vectors for the CNN input. The vectors represent the features carrying sentence information and semantic similarities among words. The vector representations of words are stored in the matrix $M \in \square^{D\times|W|}$, where D is vector dimensionality to represent a word, and $|W||$is the size of vocabulary. Usually we consider the vocabulary of words as finite. The resulting word vector file also can be used as features in other biomedical information processing and machine learning applications. Parameters M are initialized randomly and trained on a large amount of unlabeled biomedical text paper files with general neural network.

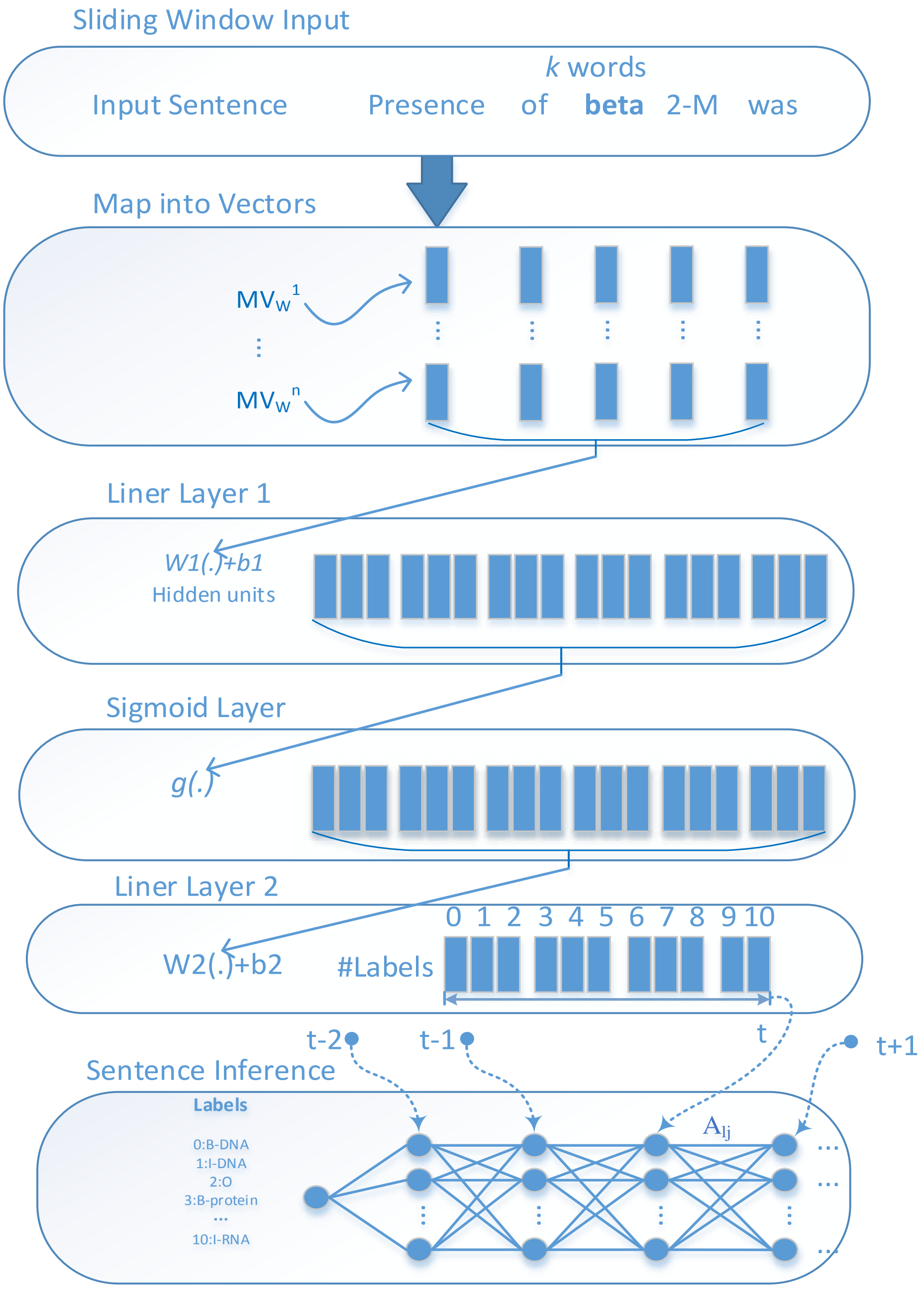


Figure 1. The Neural Network Architecture

Generally, the real-valued vector representation M can be obtained by two ways. One is initializing the vector for each word i with zero in all positions but 1 for the M_i position, and automatically optimize them as parameters during the training phase of the network [18]. The second way is to view word representation as a part of training a neural network language model [20-23]. In this paper, we optimize them on specific Bio-NER task. In this paper, we adopt the second one. Compared with different language models [23-26], we choose the skip-gram neural network language model which is not the fastest one but more suitable for rare words training, since there are many rare words only appear in biomedical literatures.

Also, if features other than words might be contributed to the task, they can be added as some dimension of word vectors. For example, a feature can be used to indicate if a word is a temporal word or it is capital. In our works, to verify the affection of deep NN on Bio-NER, there is only one feature been concerned. We will describe it later.

2.2. Extracting Sentence Level Features

For biomedical NER task, each word in sentences should be given a proper label to indicate if it is a biomedical NER or not. The sentences will be the system's inputs. The outputs are proper label sequences for each sentence. Since the length of sentences is variable, but input of NN is fixed, we choose sliding window approach like many other machine-learning problems. The window size is *k* which is decided at the beginning and system may have the different precision because of it. Using sliding window approach, the dependency information among the label of each word and its neighboring words is concerned. Therefore its neighbors in the window will pass together through the layer. When the word at position *p* is studied, it and its neighbors of positions in the range of $\left[\left(p-(k-1)/2\right),\left(p+(k-1)/2\right)\right]$ will be passed into the Mapping layer. As we mentioned earlier, each word was translated into a *D*-dimensional vector through this layer. Therefore the input size of the linear layer 1 is fixed as $D \times k$.

2.3. Label Criterion

A deep neural network can be represented as architecture with several layers. Each layer abstracts features base on the features generated by lower layers. Dependent on the design, each layer can be either linear function or other transformation. A function $f_\theta(.)$ describing the three layers in our architecture can be denoted as follows:

$$f(x) = M^2 g(\mathbf{M}^1 x + b^1) + b^2 \tag{2.1}$$

where the matrices $M^1 \in \mathbb{R}^{H \times DK}, b^1 \in \mathbb{R}^{1 \times H}, M^2 \in \mathbb{R}^{|L| \times H}, b^2 \in \mathbb{R}^{1 \times |L|}, g(.)$ is sigmoid function. *H* is the number of hidden unit and can be tuned for better performance $|L|$ is the size of all possible label tags set.

Using stochastic gradient ascent, over a training set T, the v-dimensional parameter matrix $\theta(\theta_1, \theta_{2,}, \ldots, \theta_v)$ is trained by maximizing the likelihood:

$$\sum_{(x,y)\in T} \log p(y \mid x, \theta) \tag{2.2}$$

This task is multi-class classification problem. $f(x,l,\theta)$ is used to denote the score for each lth label in the given example x which corresponds to a training word window. $f(x,l,\theta)$ can be interpreted as conditional probability $p(l \mid \mathrm{x}, \theta)$. By applying the softmax regression operation, $p(l \mid \mathrm{x}, \theta)$ should be described as:

$$p(l \mid x, \theta) = \frac{e^{f(x,l,\theta)}}{\sum_j e^{f(x,j,\theta)}} \tag{2.3}$$

Defining the log-add operation as

$$\underset{i}{\log add}\, z_i = \log\left(\sum_i e^{z_i}\right) \tag{2.4}$$

Therefore, the log-likelihood for one training example (x, y) can be expressed as:

$$\log p(y \mid x, \theta) = f(x, y, \theta) - \underset{j}{\mathrm{logadd}}\, f(x, \mathrm{j}, \theta) \tag{2.5}$$

$f\left(x_{[1:\mathrm{T}]}, l, t, \theta\right)$ is defined as the output score for the sentence $x_{[1:T]}$ and for the tag *l* at the time *t* by the system with parameters θ .

As one of sequence labeling task, the Bio-NER needs to concern the score of each label path. There are dependencies among those tags in the same sentence. For example, left boundary words of Entity category A cannot be followed by a different category B's inside words. Therefore to interpret the output we shall consider not only each word in sentence but also the dependencies between labels. To label each word in the sentence **x** is quite like searching a path with maximum score path in the graph. The score of along the path in a sentence is the sum of two parts. One is the node scores mentioned early, the other is the transition scores A_{lj} which is denote the probability transforms from label l to j. All parameter including A_{lj} and θ are denoted as $\tilde{\theta}$.

For a sentence $\mathrm{x}_{[1:T]}$, the score for a path with labels $l_{[1:T]}$is denoted as follows:

$$S\left(x_{[1:T]}, l_{[1:T]}, \tilde{\theta}\right) = \sum_{t=1}^{T}\left(A_{l_{[t-1]}l_{[t]}} + f\left(x_{[1:T]}, l_{[t]}, t, \theta\right)\right) \tag{2.6}$$

Do the similar operations as early. Taking it to exponential makes it positive and respecting it with all the paths to normalize. Therefore the log conditional probability of taking the real labeled path $y_{[1:T]}$ is interpreted as follows:

$$\log p\left(y_{[1:T]} \mid x_{[1:T]}, \tilde{\theta}\right) = S\left(x_{[1:T]}, y_{[1:T]}, \tilde{\theta}\right) - \underset{\forall l_{[1:T]}}{\log add}\, S\left(x_{[1:T]}, l_{[1:T]}, \tilde{\theta}\right) \tag{2.7}$$

During the training procedure, all the parameters $\tilde{\theta}$ are trained over all the training examples $(x_{[1:T]}, y_{[1:T]})$ to maximize the $\sum_{(\mathrm{x},\mathrm{y}\in\mathrm{T})} \log p\left(y_{[1:T]} \mid x_{[1:T]}, \tilde{\theta}\right)$. Later in the inference procedure the Viterbi algorithm is chosen to find $\underset{l_{[1:T]}}{\arg\max}\, S\left(x_{[1:T]}, l_{[1:T]}, \tilde{\theta}\right)$.

2.4. Stochastic Gradient

Gradient descent algorithms are the simplest optimization algorithms for minimizing a formula, but considering the huge cost of computation, we choose an optimizing method, stochastic gradient descent[27]. In each iteration step, an example (x, y) is drawn at random, and a new value of θ is computed.

$$\theta \leftarrow \theta + \varepsilon \Delta \log p\left(y \mid x, \theta\right) \tag{2.8}$$

$\Delta \log p(y \mid x, \theta)$ denotes the gradient of $\log p\left(y_{[1:T]} \mid x_{[1:T]}, \theta\right)$ with respect to θ and the chosen learning rate ε is a small positive constant.

3. Experiments

3.1. Task Description

Bio-DNA task is to recognize the entities ranging from protein/gene names to disease/virus names and label them with particular labels in plain biomedical text. Shown in the Figure 2, every word in a sentence is regarded as a token which is associated with a label. Labels such as B-C, I-C or O indicate not only the category of the Named Entity (NE) but also the location of the token within the Named Entity. In the label denotation, C is the category labels; B and I are location labels, standing for the beginning of an entity and inside of an entity respectively. There are 5 categories labels: protein, DNA, RNA, cell_type, cell_line. O indicates that a token is not part of a NE. In GENIA corpus, the test file is given as BIO notation. Totally 11 labels are included using BIO notation as shown in Figure 2. Each token in the biomedical text will be assigned with one of the 11 labels in the recognition results.

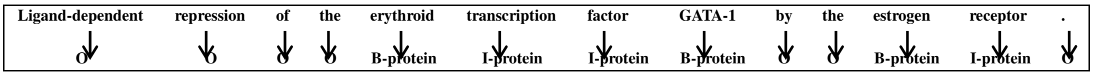


Figure 2. Example of Biomedical Named Entity Recognition

3.2. Experiment Result and Analysis

The word representation is trained using a skip-gram neural network language model with unlabeled training data. Unlabeled data are collected from the PUBMED database using tools biopython[1] and 'drug', 'interaction', 'protein', 'DNA', 'Cell type' are chosen as keywords for searching. We download totally 339084 associated papers from pubmed database and 294993 papers have abstracts. Totally 431 MB text file is used as unlabeled data set. Word2vec[2] tool is used for implement our skipgram language model. Totally 205924 words with 600 dimension vectors are included in the dictionary$W$.

Except these features, we also consider the POS feature using part-of-speech tagger tools[3] which is specifically designed for biomedical text since the characteristics of biomedical text are quite different from those of newspaper articles.

In our experiment, GENIA corpus is applied. Precision, recall and F-score are selected as our evaluation measurement. The precision is the number of NEs a system correctly detected divided by the total number of NEs identified by the system. The recall is the number of NEs a system correctly detected divided by the total number of NEs contained in the input text. $F - score = 2 precision * Re call / (Precision + Re call)$ stands for the harmonic performance of a system.

The precision, recall and F-score of our methods are presented in the Table X. Some categories like protein and RNA have much higher performance than others. The recognition of Protein category has highest F-score. This situation also happened in other system. Some researchers believed it is due to a lack of the train data. The amounts of each entity category in the training data are presented in the Table 1.

However after comparing Table 1 and Figure 3, we found the amount of training data is not the main reason. The cell_type category has the small training data set but with the highest precision and second highest F-score. On the other hand, the DNA category has second largest of training data set. Figure 4 shows that there are 12% 'B-DNA' category words are wrongly labeled into 'B-protein' which is much more than other biomedical categories.

Table 1. Performance of Major Entity Categories

Category	recall	precision	F-score
protein	0.8062	0.6389	0.7129
cell_line	0.6160	0.5008	0.5525
DNA	0.6761	0.6427	0.6590
cell_type	0.7356	0.7344	0.7350
RNA	0.6102	0.6050	0.6076
Overall	**0.7610**	**0.6486**	**0.7003**

[1] https://biopython.org/wiki/Main_Page
[2] https://code.google.com/p/word2vec/
[3] http://www.nactem.ac.uk/tsujii/GENIA/tagger/

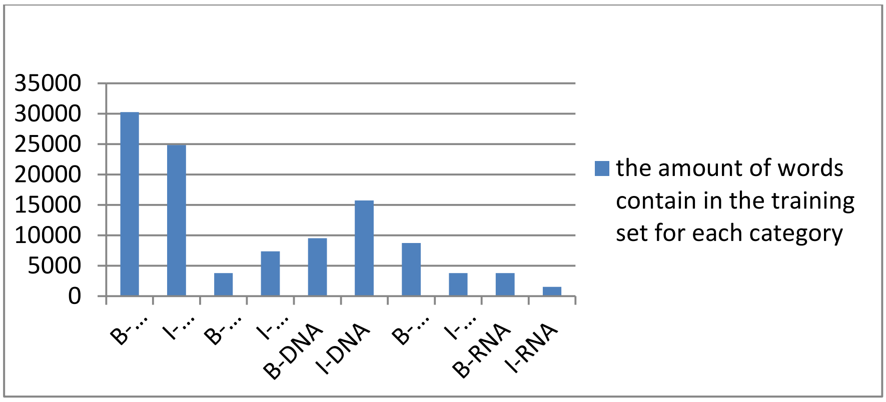


Figure 1. Training Data for Major Entity Categories

After further researching on the training data, we found two main reasons. One is that the Biomedical Named Entities are usually composed of several nested named entities. For DNA and protein categories there are many overlapping nested named entities. For example, there are two training examples in the figure 5. Words ‘Epstein-Barr’, ‘viruses’, ‘EBV’, ‘protein’, ‘cell’ and punctuations ‘(’, ‘)’ are embedded in both entities but they belong to two different categories.

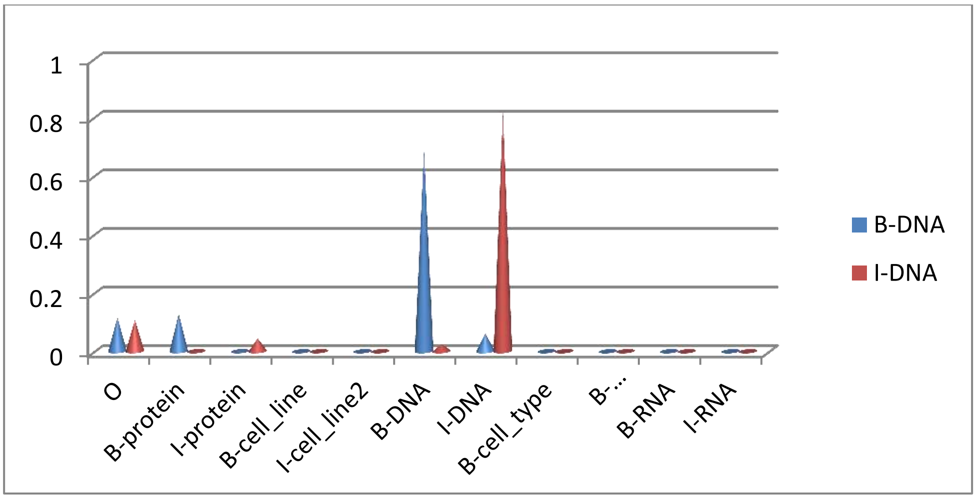
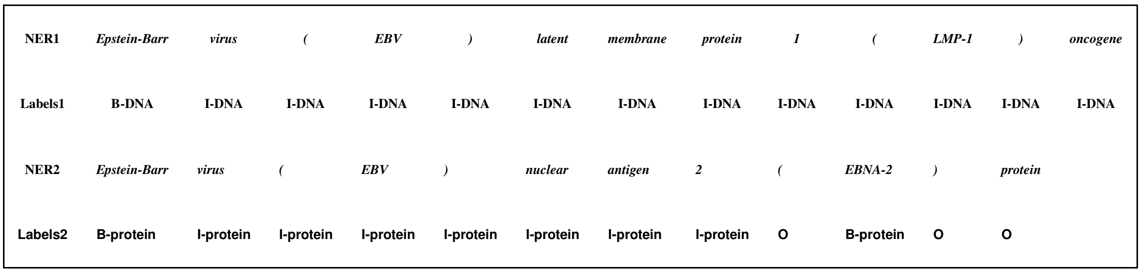


Figure 4. Distribution of Errors

After analyzing the data we find that these words may appear at different position according different categories. For example the word ‘cells’ usually locates at the end of cell-type entities but in the middle of cell-line entities. Since the current BIO notation cannot present such information, to utilize this kind of information, the BMESO notation is applied in the middle steps of processing.



NER1	*Epstein-Barr*	*virus*	*(*	*EBV*	*)*	*latent*	*membrane*	*protein*	*1*	*(*	*LMP-1*	*)*	*oncogene*
Labels1	B-DNA	I-DNA	I-DNA	I-DNA	I-DNA	I-DNA	I-DNA	I-DNA	I-DNA	I-DNA	I-DNA	I-DNA	I-DNA
NER2	*Epstein-Barr*	*virus*	*(*	*EBV*	*)*	*nuclear*	*antigen*	*2*	*(*	*EBNA-2*	*)*	*protein*	
Labels2	B-protein	I-protein	I-protein	I-protein	I-protein	I-protein	I-protein	I-protein	O	B-protein	O	O	

Figure 5. Examples of NERs and Labels

BMESO notation is quite like the BIO notation. However it can give more detail description of the each word position in the entities. B still indicates the beginning of an entity. E indicates the end of an entity. Words between the beginning word and end word will be noted with M. If the entity is single word will be marked with S. The experiments show that the BMESO can improve the performance.

The second reason is lack of labeled training set and some entities in the test file do not appear in the training set. In the biomedical field, each day there are some new words been generated with some vanished at the same time. Therefore such case is restricted by the task itself and public data sets.

Then we get the final results on GENIA test file and list them on the following Table 2.

Table 2. Comparisons with some State-of-the-art Systems

Teams	**recall**	**precision**	**F-score**
Our result	**76.13**	**66.54**	**71.01**
Sasaki et al.'s[28]	79.85	68.58	73.78
Liao et al.'s[14]	73.6	72.8	73.2
Sun et al. 's[29]	72.3	70.2	71.2
ABNER[30]	72.0	69.1	70.5
Saha et al.'s[31]	67.66	68.12	67.89

Compared with other teams, our result is very close to the start-of-art results with the least amount feature template and without building lexicon. The dictionary-based approach is beneficial when the composition of lexicon is relative stable. However biomedical dictionary is changing every day. And design feature template is a labor-intensive work, and it will be different because of changing of tasks and corpora. The two type solutions have limit flexible.

4. Conclusion

We have presented a multiple-layer neural network and apply it on Biomedical NER task. We have achieved close to state of the art performance using the new model. The results reveal this neural network models without extra hand-crafted features bring great performance.

In future work we will focus on other ways to further improve the performance of neural network. First, we realize the left boundary word is really important and can affect the recognition of whole entity which may include several words. Therefore once the first word is wrongly labeled, the following words will be incorrect also. The reverse recognition binding with forward recognition will be explored next. Second, statistical features may be integrated.

References

[1] H. Dai, Y. Chang, R. Tzong-Han Tsai and W. Hsu, "New Challenges for Biological Text-Mining in the Next Decade", Journal of Computer Science and Technology, vol. 25, no. 1, **(2010)**, pp.169-179.

[2] M. Krallinger, A. Morgan, L. Smith, F. Leitner, L. Tanabe, J. Wilbur, L. Hirschman and A. Valencia, "Evaluation of Text-Mining Systems for Biology: Overview of the Second Biocreative Community Challenge", Genome biology, vol. 9, no. 2, **(2008)**.

[3] H. Dai, C. Huang, R. Lin, R. Tsai and W. Hsu, "Biosmile Web Search: A Web Application for Annotating Biomedical Entities and Relations", Nucleic Acids Research, vol. 36, **(2008)**, pp. 390-397.

[4] D. Rebholz-Schuhmann, M. Arregui, S. Gaudan, H. Kirsch and A. Jimeno, "Text Processing through Web Services: Calling Whatizit", Bioinformatics, vol. 24, no. 2, **(2008)**, pp. 296-300.

[5] L. Si, T. Kanungo and X. Huang, "Boosting Performance of Bio-Entity Recognition by Combining Results from Multiple Systems", Proceedings of the 5th international workshop on Bioinformatics ACM, **(2005)**, pp. 76-83.

[6] A. Vlachos, "Evaluating and Combining Biomedical Named Entity Recognition Systems", BioNLP 2007: Biological, translational, and clinical language processing, **(2007)**, pp. 199-206.

[7] 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, in: Advances in Informatics", Springer, **(2005)**, pp. 382-392.

[8] J. Finkel, S. Dingare, H. Nguyen, M. Nissim, C. Manning and G. Sinclair, "Exploiting Context for Biomedical Entity Recognition: From Syntax to the Web", Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications. Association for Computational Linguistics, **(2004)**, pp. 88-91.

[9] K. Lee, Y. Hwang, S. Kim and H. Rim, "Biomedical Named Entity Recognition Using Two-Phase Model Based on Svms", Journal of biomedical informatics, vol. 37, no. 6, **(2004)**, pp. 436-447.

[10] G. Zhou, J. Zhang, J. Su, D. Shen and C. Tan, "Recognizing Names in Biomedical Texts: A Machine Learning Approach. Bioinformatics", vol. 20, no. 7, **(2004)**, pp. 1178-1190.

[11] L. Li, R. Zhou and D. Huang, "Two-Phase Biomedical Named Entity Recognition Using Crfs. Computational Biology and Chemistry", vol. 33, no. 4, **(2009)**, pp. 334-338.

[12] C.-N. Hsu, Y.-M. Chang, C.-J. Kuo, Y.-S. Lin, H.-S. Huang and I.-F. Chung, "Integrating High Dimensional Bi-Directional Parsing Models for Gene Mention Tagging", Bioinformatics, vol. 24, no. 13, **(2008)**, pp. 286-294.

[13] R. T. Tsai, C.-L. Sung, H.-J. Dai, H.-C. Hung, T.-Y. Sung and W.-L. Hsu, "Nerbio: Using Selected Word Conjunctions, Term Normalization, and Global Patterns to Improve Biomedical Named Entity Recognition", BMC bioinformatics, vol. 7, no. 5, **(2006)**.

[14] Z. Liao and H. Wu, "Biomedical Named Entity Recognition Based on Skip-Chain Crfs", Industrial Control and Electronics Engineering (ICICEE), 2012 International Conference on. IEEE, **(2012)**, pp. 1495-1498.

[15] L. Li, W. Fan and D. Huang, "A Two-Phase Bio-Ner System Based on Integrated Classifiers and Multi-Agent Strategy", **(2013)**.

[16] R. Collobert, "Deep Learning for Efficient Discriminative Parsing", International Conference on Artificial Intelligence and Statistics, **(2011)**.

[17] R. D. Y. Bengio and P. Vincent, "A Neural Probalilistic Language Model", In NIPS, vol. 13, **(2001)**.

[18] R. C. A. J. Westion, "A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning", In ICML, vol. **(2008)**.

[19] J. W. R. Collobert, L. Bottou, M. Karlen, K, Kavukcuoglu and A. P. Kuksa, "Natural Language Processing (Almost) from Scratch", JMLR, **(2011)**.

[20] R. E. D. Yoshua Bengio, P. Vincent and C. Janvin, "A Neural Probabilistic Language Model", J. Mach. Learn. Res., vol. 3, **(2003)**, pp. 1137-1155.

[21] H. Schwenk, "Continuous Space Language Models", Computer Speech & Language, vol. 21, no. 3, **(2007)**, pp. 492 -518.

[22] T. Mikolov, M. Karafiat, L. Burget, J. Cernocky and S. Khudanpur, "Recurrent Neural Network Based Language Model", Eleventh Annual Conference of the International Speech Communication Association (INTERSPEECH), **(2010)**, pp. 1045-1048.

[23] A. Mnih and Y. W The, "A Fast and Simple Algorithm for Training Neural Probabilistic Language Models", Proceedings of the 29th International Conference on Machine Learning (ICML-12), **(2012)**, pp. 1751-1758.

[24] R. Collobert, "Deep Learning for Efficient Discriminative Parsing", In International Conference on Artificial Intelligence and Statistics (AISTATS), **(2011)**.

[25] L. A. R. Joseph Turian and Y. Bengio, "Word Representations: A Simple and General Method for Semi-Supervised Learning", Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, **(2010)**, pp. 384-394.

[26] W. T. Y. Tomas Mikolov and G. Zweig, "Linguistic Regularities in Continuous Space Word Representations", Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, **(2013)**, pp. 746-751.

[27] L. Bottou, "Stochastic Gradient Learning in Neural Networks", Proceedings of Neuro-Nimes, vol. 91, **(1991)**.

[28] Y. T. Y. Sasaki, J. McNaught and S. Ananiadou, "How to Make the Most of Ne Dictionaries in Statistical Ner. Proc. Workshop Current Trends in Biomedical Natural Language Processing", **(2008)**, pp. 63-70.

[29] Y. G. C. H. Sun, X. L. Wang and L. Lin, "Rich Features Based Conditional Random Fields for Biological Named Entities Recognition", Computers in Biology and Medicine, vol. 37, **(2007)**, pp. 1327-1333.

[30] Abner: A Biomedical Named Entity Recognizer, **(2013)**, pp. 46-51.

[31] S. N. S. K. Saha, S. Sarkar and P. Mitra, "A Composite Kernel for Named Entity Recognition", Pattern Recognition Letters, vol. 3, **(2010)**, pp. 1591-1597.