Improved convolutional neural network for biomedical word sense disambiguation with enhanced context feature modeling

REN Kai [1,2*], WANG Shi-Wen[3]
[1]Computer School, Wuhan University, Wuhan, 430072, China
[2]College of Computer Science, South-Central University for Nationalities,Wuhan, China
[3]Confucius Institute, New Jersey City University, 2039 Kennedy Blvd
Jersey City, NJ, 07305,USA
*Corresponding Author, email:rk8123@gmail.com

Journal of Digital Information Management

ABSTRACT: *Polysemy is a common phenomenon in the biomedical domain. Ambiguous words directly influence the accuracy of computer semantic analyses. Thus, word sense disambiguation (WSD) is often conducted in advance. Most current biomedical WSD methods rely on manual selection of features for WSD. To identify latent context features from a deep layer and reduce the negative influence of manual selection of features in WSD, this paper proposes the Convolutional Neural Network (CNN) method for biomedical WSD with enhanced text feature modeling. First, this program automatically conducts crawling of a large scale of relevant corpus from MEDLINE for training and obtains relevant context feature vectors. These feature vectors are subsequently adopted as input data in CNN. Finally, the CNN classification method is used for WSD. By testing 203 commonly used ambiguous words from MSH-WSD corpus, the author finds that the average accuracy of the proposed method is 94.65%, which is a significant improvement relative to that of previous methods. This result proves that CNN is an efficient WSD method to be used in the biomedical domain. Given that context feature representation and WSD are important pre-works in extraction and retrieval of biomedical information, WSD can reduce the negative effect of ambiguous words on accuracy of such pre-works.*

Subject Categories and Descriptors
I.2.7 [Artificial intelligence]: Natural Language Processing - Text analysis; **F.1.1[Computation by abstract devices]:** Modles of Computation - Self-modifying machines

General Terms
Algorithms, Performance,Languages

Keywords: Convolutional neural network, Biomedicine, Semantic representation, Word sense disambiguation

Received: 22 July 2016, Revised: 8 September 2016, Accepted: 15 September 2016

1. Introduction

Word sense disambiguation (WSD) is employed to identify which sense of a word is used in a sentence when the word has multiple meanings. WSD lays the foundation for natural language processing, including information extraction, machine translation, and information search. Ambiguous words are commonly used in the biomedical domain. When a computer automatically processes a medical document, these ambiguous words might be difficult to be comprehended by the computer.

Disambiguation of biomedical word senses is an issue not to be ignored by a computer used in processing biomedical texts.

Several conventional approaches are employed in WSD. First are the unsupervised methods. These methods directly use raw, unannotated corpus in learning and training. Well-trained algorithms can be used in a test corpus to directly obtain WSD results. Many effective unsupervised methods have been widely applied in raw, unannotated corpus. With considerable human labor that is being saved, favorable results can be obtained. Second are knowledge-based methods. These methods rely mainly on dictionaries, Wikipedia, WorldNet, Hownet, and other external resources. Given that external resources are not universal, they are possibly ineffective external resources for a specific domain. In this way, the effects of machine learning cannot be guaranteed. Moreover, it takes a large amount of calculation resources to learn a large amount of external resources. Third are supervised methods. These methods use some sense-annotated corpus for training. Well-trained algorithms can be used in test set. These methods are often useful in obtaining favorable results. However, the requirement of some sense-annotated corpus limits the applications of these methods.

Most of the ambiguous biomedical words are fixed, making manual or automatic annotation a possibility. On the basis of MSH-WSD corpus, this paper conducts supervised WSD research to enhance the effects of context feature representation and improve WSD accuracy in the biomedical domain.

2. Related Works

WSD has long been a research hotspot in the natural language processing (NLP) domain. Based on theory of machine learning, three commonly used approaches to WSD exist, namely unsupervised methods, knowledge-based methods, and supervised methods. A series of research on these approaches have been conducted.

The unsupervised methods generally classify ambiguous words into several classes through clustering or classification. Every class is mapped to a standard sense, thereby realizing WSD. Apart from their role in WSD, unsupervised methods can be used to discover new words. On the basis of the current research on unsupervised WSD, Pedersen [1] compared several high-order matrix representation vectors for WSD and used sense clusters to realize unsupervised WSD. Brody et al. [2] conducted WSD using context-based Bayesian model, which is a classical WSD model. Many other scholars have explored a series of improvements based on this model. Chasin et al. [3] compared the algorithms based on graphs with the unsupervised WSD methods based on topic models and then summarized the predominating unsupervised WSD methods. The author previously used unsupervised method based on kernel function fuzzy C-means clustering [4] used to disambiguate ambiguous biomedical terms, and favorable results were obtained. Most of the above methods are WSD methods directly derived from traditional domains. No WSD methods specific for the biomedical domain have been established and thus breakthroughs in biomedical WSD effects are limited.

Knowledge-based WSD methods rely on external resources, such as Wikipedia and professional dictionaries, rather than on annotated corpus for WSD. For example, Navigli et al. [5] improved WSD effects by using page classification features of Wikipedia. However, Wikipedia is not focused on a specific domain. It possibly provides information relevant to the biomedical domain, but the knowledge provided is possibly insufficient. Another method is a graph-based method proposed by Agirre et al. [6]. This method originates from the ontological knowledge of WordNet. This method also does not target the biomedical domain. McInnes et al. [7] used Unified Medical Language System (UMLS) as external resource to improve WSD accuracy in the biomedical domain. This method can be regarded as a WordNet-based WSD method in this domain. This research was a leading one considering its era background. Jimeno-Yepes et al. [8] built a standard corpus dataset based on online resources to eliminate ambiguities in biomedical words. Their research provided a standard corpus used to evaluate future research works. All of the above are representative knowledge-based WSD methods. The use of external knowledge resources can significantly improve WSD effects but not all ambiguous words to be handled with can find appropriate resources to complete WSD. Such is the limitation of these methods.

Supervised methods train models based on ambiguous words with annotated senses. Thus, the models trained can effectively classify and recognize ambiguous words of unannotated corpus, thereby realizing WSD. Many new findings on supervised methods have been recently obtained. Zhong and Ng. [9] have proposed a multi-featured WSD system based on Support Vector Machine (SVM). Given its high adaptability, this method presents a wide range of applications in engineering. Stevenson et al. [10] improved WSD accuracy by using linguistic and MeSH features. These two investigations focused on feature refinement, which to some extent improved WSD accuracy. However, feature selection is done manually, resulting in poor migration nature of algorithms. Other representative scholars in supervised methods include McInnes et al.[11], who combined MetaMap automatic annotation and UMLS, and Jimeno Yepes et al. [12], who conducted a series of in-depth research on textural features. All of the above methods have made some breakthroughs in feature representation, effectively improving WSD accuracy. The most remarkable research finding is the result of Jimeno Yepes et al. [13], who combined word embedding and Long Short-Term Memory (LSTM), which is a publicly acknowledged supervised WSD method showing the highest accuracy. This study compares the proposed method and the most recognized method used to analyze whether improved embedding can

increase WSD accuracy.

In previous research, context features were mostly set artificially, inevitably causing uncertainty. This paper uses the supervised WSD based on word2vec context feature embedding. By using the word2vec representation method, context features of words to be disambiguated can be mapped to a high-dimensional space. Favorable results were achieved by Yoon Kim [14], who used Convolutional Neural Network (CNN) for textual classification based on sentences. The present study attempts to improve the CNN classification method for WSD. Moreover, this paper investigates abbreviation disambiguation in the biomedical domain. An experiment is specifically designed to disambiguate ambiguous abbreviations.

This paper consists of five sections. Section 2 presents the status of WSD research in the biomedical domain. Section 3 introduces the relevant algorithms and models involved in the experiment and collection of experimental data. Section 4 presents the analysis and discussion of results. Section 5 summarizes the entire paper and presents the conclusions of this research.

3. Methodology

3.1 Algorithms

3.1.1 Word embedding

In a distributed semantic representation model, a vector should be built to express context information. The context is confirmed by a context window or by an N-gram model. In previous research, context feature information is usually formulized by frequency. A method based on Term-Frequency-Inverse Document Frequency is the most commonly used method.

Some improvements based on previous context representation methods were recently made. Many matrix decomposition-based methods, including Latent Semantic Analysis and word2vec representation technology, have become popular. word2vec is a word embedding toolkit that can train vector space models faster than the previous approaches. It can utilize either of the two model architectures, namely, Continuous Bag-of-Words (CBOW) and Skip-Gram, to generate distributed representations of words [15]. From the perspective of algorithms, these two models are quite similar to each other. In CBOW architecture, the model predicts target words based on keywords and context words. By contrast, Skip-Gram architecture predicts source words based on target words. The difference between them lies in their processing, wherein CBOW algorithm conducts smoothing of numerus distributed information, and such processing is helpful for small-sized datasets. By contrast, Skip-Gram algorithm regards every context-target word combination as an observation, and such processing is more effective in large datasets. In this study, datasets are relatively small and closed, so the CBOW algorithm is adopted for word vector representation.

Semantic distribution, which is usually called word embedding, is currently represented by neural network techniques. The technique word embedding conducts modeling based on the complex relationship between the context and the target words via the neural network. Different types of neural networks with different parameters exist. During neural network representation, modeling of the complex context is conducted to include more semantic information. Moreover, along with increase in length, parameters increase linearly rather than geometrically; thus, they are highly practical. This paper considers the particularity of abbreviation in corpus and incorporates it into the word embedding features.

3.1.2 CNN

CNN was proposed by Fukushima [16] in 1980. Local perception and weight sharing are the core features of CNN. In a general feed-forward artificial neural network, every node in a hidden layer is connected to all nodes in the output layer; by contrast, in CNN, every node in a hidden layer is simply connected to a node with a fixed size or window size. In the sub-network starting from the fixed area to the hidden layer, the weights are shared by all areas in the hidden layer. The equation from the input layer to the hidden layer is as follows:

$$x_i = \lfloor e(\omega_{i-\lfloor \omega in/2 \rfloor});...; e(\omega_i);...;e(\omega_{i-\lfloor \omega in/2 \rfloor}) \rfloor \quad (1)$$

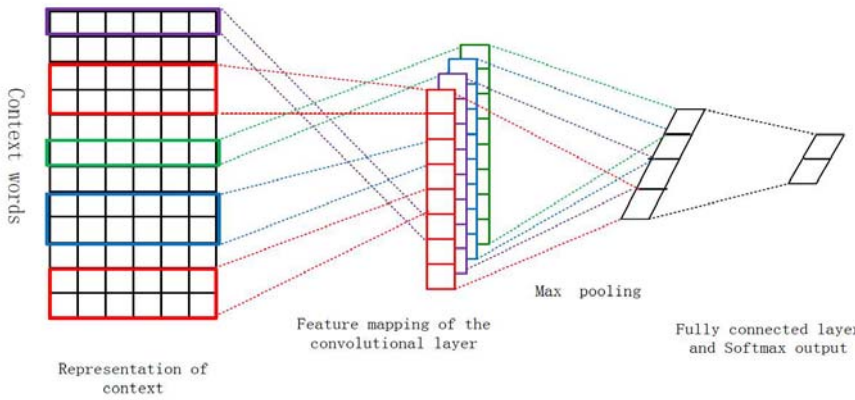

Figure 1. Convolutional neural network model for context classification

$$h_i^{(1)} = tanh\,(W_{xi} + b) \quad (2)$$

Investigation on word and document semantic vector representation methods based on the neural network showed that after several hidden layers, the CNN usually turns to the pooling technique to compress hidden layers of different lengths into hidden layers of a fixed length. The commonly used pooling techniques include mean pooling and max pooling. This paper adopts the max pooling [17], which is expressed as follows:

$$h^{(2)} = \max_{i=1}^{n} h_i^{(1)} \quad (3)$$

Through the convolution kernel, the CNN model can conduct modeling of local information of every part in the context. Full-text semantics from different local information can be integrated by using the pooling layer, and the model's overall complexity is *O* (n). The CNN model presents wide applications. In NLP domain, Collobert et al. [18] applied CNN in semantic role labeling, succeeding in effectively the system performance. In 2014, Kalchbrenner et al. [19] and Kim independently used the CNN model for text classification. Moreover, Zeng et al. [20] have proposed relationship classification using the CNN model. Applications of the neural network model contributed to the improvement of the above research findings.

This paper makes some improvements on the Kim's model and conducts WSD based on an improved CNN model. Figure 1 shows the schematic of the CNN model. In Eq. (1), X is the K-dimension word vector. Every dimension corresponds to every word in a sentence. The length of the sentence determines the length of the vector. In convolutional layer of CNN, a window parameter, h, exists, and the continuous h words in every group forms a feature of the convolutional layer. All of these features when combined form a feature matrix. In the subjacent layer, max-over-time pooling, also known as max-pooling, is adopted. Max-pooling adopts the maximum value of the features as the feature value. The core idea of max-pooling can is that it representing the feature of every convolutional layer with the most significant feature. Through pooling, output sentences with different lengths are made uniform.

In the CNN model, each filter generates one feature. The number of model filters is related to window length. This paper describes the process by which a feature is extracted from a filter. Multi-module filters correspond to multiple features. These features output the probability of every class from the second layer from the bottom via a fully connected softmax layer. The improved model finally realizes a multi-label classification system based on full text features.

3.1.3 Text classification

Traditional text classification methods mainly focus on three aspects: feature representation, feature selection, and selection of appropriate machine learning algorithm. Improvement in these three aspects can increase the accuracy of text classification.

Bags-of-words, parts of speech labels, noun phrases [21], and tree kernels [22] are most commonly used in feature representation. Different features describe data from different perspectives. Combinations of different features are required to better describe texts. However, these features are all faced with the problem on data sparsity. Reduction of noise features is considered a useful approach to improve text classification. The most commonly used method is removal of stopwords. Advanced text classification methods select features based on information gain, mutual information [23], and other features. Ng et al. [24] have recently added L1 regularization into the optimization objective to automatically learn sparsity feature. This method has played an important role in large-scale applications of text classification.

Nearly all classifier algorithms, such as nearest neighbor classifier and decision tree classifier, have found applications in context classification. In terms of large-scale text classification tasks, linear classifiers present the widest applications. These classifiers include Logistic Regression, Naïve Bayes, and Support Vector Machine (SVM). This study adopts CNN for text classification and compares it with the traditional method based on SVM. The results show that the neural network method based on deep learning significantly increases the accuracy of text classification.

3.2 Experiment design

3.2.1 Experimental environment

OS : Ubuntu 14.04LTS

CPU : Intel Core i7-6700k @4.0GHz * 8

RAM : 16GB 2400 MHz DDR4

GPU : NVIDIA GeForce GTX 960 4G

Python : 2.7.6

Theano : 0.7.0

CUDA : 7.5.17

3.2.2 Corpus preparation

The MSH-WSD Data Set of the National Library of Medicine (NLM) contains 203 ambiguous words, 106 of which are abbreviations, 88 are ordinary ambiguous terms, and 9 are disambiguated words of mixed types. This study uses corpus for experiments. Moreover, this study adopts python programming by NLM for the MEDLINE interface, and every ambiguous word is regarded as a keyword (appearing in the title and abstract). Titles and abstracts of the latest 4,000 retrieval results are collected from MEDLINE. The size of the texts collected as training corpus is approximately 800M. The Word2vec is used to generate vector .bin document to represent the vector for CNN training and learning. In the experiment, the author adopts the collected text corpus as training corpus of word2vec. The following parameters are adopted for word2vec training: cbow 1 -size 200 -window 8 -negative

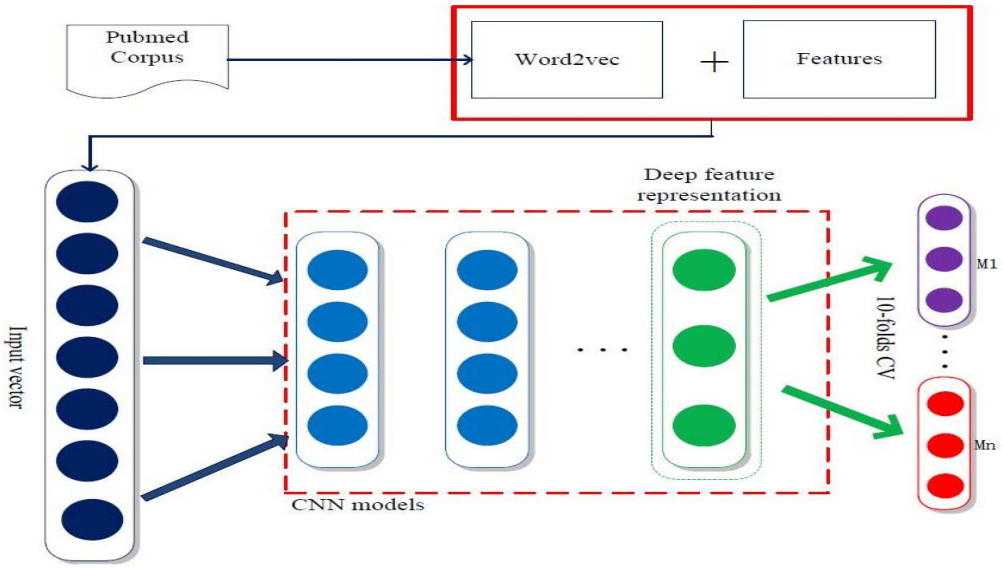

Figure 2. Framework of CNN model-based abbreviation disambiguation

25 -hs 0 -sample 1e-4 -threads 20 -binary 1 -iter 15. A vector .bin document can then be obtained to serve as input in the CNN. The dimensionality of the parameters is set at 200, which should be consistent with the number of input in the CNN.

3.2.3 Experimental workflow

Figure 2 shows the overall experimental workflow. The words to be disambiguated in the original corpus are combined with other semantic features through word2vec word embedding. All features are integrated into a vector and adopted as input data for CNN. The input data are optimized using multi-layer CNN. Semantically different vectors are separated. The separated semantic vectors are mapped to the relevant semantic classes to complete WSD.

3.2.4 Parameter settings of the CNN

The corpus containing ambiguous words is preprocessed. The author prepares two types of learning corpus. The first type is the corpus consisting of sentences containing ambiguous words. The second type is the corpus consisting of titles and full abstracts. The corpus types are divided into training set and test set. A 10-fold cross validation is adopted. Ninety percent of the corpus is used for training, whereas 10 percent in used for testing. Every ambiguous word undergoes 10 rounds of validations. The mean value is adopted as the classification accuracy.

In terms of the CNN, this experiment uses the parameters as followed: w=200,filter-hs=[3,4,5],hidde-nunits=[100,2], dropout-rate=[0.5],shuffle-batch=True,n-epochs=25,batch-size=50,lr-decay=0.95,conv-non-linear="relu",spr-norm-lim=9,non-static=True

A GPU is used for training of words, each word training lasts for approximately 20 minutes. The GPU operation greatly increases the calculation speed. The difference in the calculation time when the full text and the sentences containing ambiguous words were used as corpus is approximately 10-fold.

Methods	***AEC***	***MRD***	***2-MRD***	***UMLS SenseRelate***	***CNN Sentence S200***	***CNN FulltextS200***
Accuracy (%) (Abbreviations)	*90.90*	*87.59*	*85.01*	*83.00*	***94.91***	*96.40*

Table 1. Results for the abbreviations corpus

Methods	***SVM W50 S100***	***SVM W50 S150***	***SVM W50 S300***	***SVM W50 S500***	***LSTM W50 S100***	***LSTM W50 S500***	***CNN Full text W50 S200***
Acccuracy (%) (MSH-WSD)	*94.30*	*94.40*	*94.59*	*94.64*	*94.64*	***94.92***	*94.65*

Table 2. Results for the MSH-WSD corpus

4. Results And Discussions

4.1 Test settings

In the MSH-WSD corpus containing 203 ambiguous words, the development set and the test set are not separated. To avoid parameter changes and influence of various random factors, this study adopts a 10-fold cross validation featuring random distribution and divides the corpus into training set, development set, and test set. Moreover, initialization of the word vectors and the CNN parameters follows the same random initialization rules. In this manner, plans and parameters to test the data set are unified, and the influence of manual intervention on final results is avoided. This experiment is divided into two parts. The first part compares the 106 abbreviations in the MSH-WSD corpus. The second part compares all ambiguous words in the MSH-WSD corpus.

4.2 Analysis of the results for WSD of abbreviations

The experiment involving the corpus containing 106 ambiguous abbreviations is divided into two types. The first type uses the method similar to that proposed by Kim, in which the corresponding vectors of sentences containing ambiguous words are regarded as neural network input. The second type uses the corresponding vector of the full context containing ambiguous word as neural network input. The corresponding accuracy of the above methods is 94.91% and 96.40% (Table 1). In the experiment, the computing time required by the two methods differed by 10-fold. The method adopting the full text as input is considerably more time consuming that the method adopting sentences as input. However, this experiment showed that even if the method used sentences as input in CNN, the accuracy of WSD is obviously higher than that of the other methods. Comparative analysis shows that increase in context-based input windows positively influences the experimental results but the calculation time thus cost will also be greatly increased if a relative balance is to strike between the two.

In the data set containing 106 abbreviations, the accuracy of the proposed method significantly increases to 96.40% compared with that of several traditional methods, including AEC, MRD, 2-MRD, and UMLS SenseRelate. The improvement allows the author to learn about better forms of feature embedding from the MEDLINE corpus. Moreover, the CNN can better summarize latent features in the context. Compared with the manually derived features, these automatically derived latent features are more reasonable.

4.3 Comparative analysis of sentence input and full text input

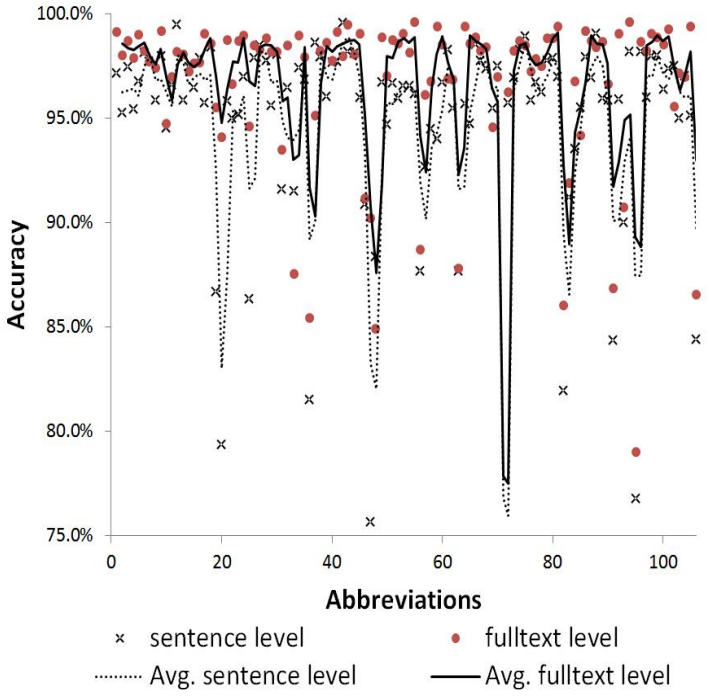

Figure 3. Sentence level vs. full-text level

Figure 3 compares the CNN based on the full text level input and the sentence level input. The continuous and dotted lines represent the CNN based on full text level input and sentence level input, respectively. The overall accuracy of CNN based on the full-text input is obviously higher than that of the CNN based on sentence input. Numerically, the full text level embedding increases the accuracy of WSD by 1.57% compared with that of the sentence level embedding. The results of the comparative experiments suggest that Increasing the quantity input data can improve the abbreviation disambiguation effects. When the value of input data is large, a high computing

capacity is required. The full text method requires 10 times the amount of time required by the sentence level method. Given the limited computing capability of the experimental environment, this paper does not attempt to further expand the comparative experiments.

4.4 WSD results for CNN full text and analysis

Most of the existing studies do not evaluate the disambiguation of abbreviations alone. Generally, they use the overall MSH-WSD set containing 106 abbreviations, 88 ordinary ambiguous terms, and 9 disambiguated words of mixed types for the test. The author compares different methods based on overall MSH-WSD set, and the results are shown in Table 2.

Table 2 shows that the proposed CNN method is obviously superior to the SVM-based method, but the final results are slightly different from the results obtained by the Recurrent Neural Network method with the LSTM algorithm. Partial optimization of abbreviations conducted in this research is attributed to such difference. Special features of ordinary terms are not included. Additionally, given the limited computing capacity of the experimental environment, this experiment adopts the 200-dimension vector. By contrast, the state-of-the-art results obtained by the LSTM method adopt the 500-dimension vector. The results for the 500-dimension vector are obviously superior to those for the 100-dimension vector. Figure 4 shows the fitting analysis of the test results obtained by the SVM, LSTM, and CNN methods at vector spaces of different dimensions. The results obtained by the LSTM method show the highest accuracy, and that results obtained by the proposed CNN method are close to those of the LSTM method and significantly higher than those of the SVM method. It can be inferred that if the vector dimension increases, the final results of the proposed method improve. In this case, the author needs to spend more time on calculation.

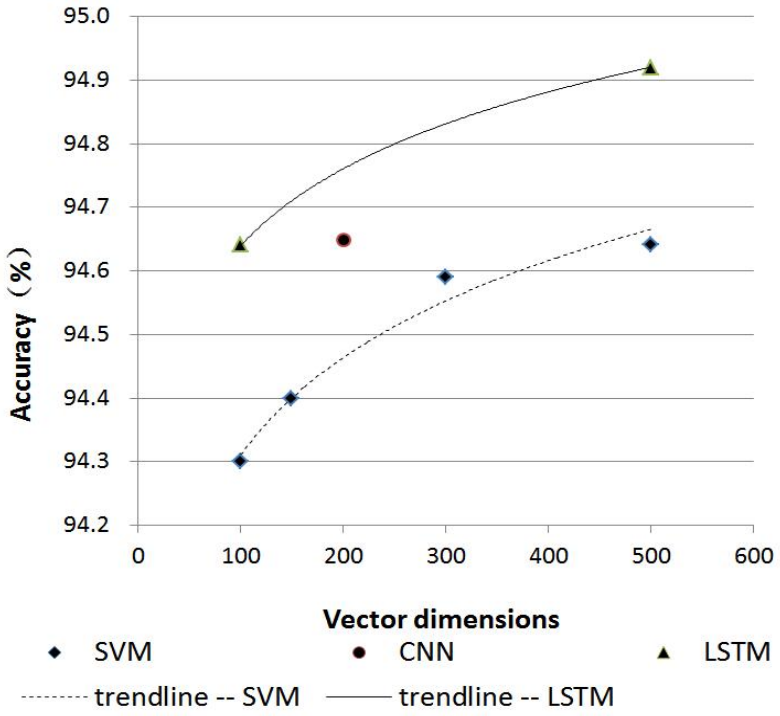

Figure 4. Results of SVM, LSTM and CNN methods in different vector dimensions

The proposed method shows a result that is nearly similar to the best results obtained by LSTM. Comparison between Table 1 and Table 2 shows that the proposed method demonstrates better abbreviation WSD effects than the full text level method. This result is attributed to the fact that word embedding in this study accounts for the expanded form of abbreviations but does not accommodate the feature optimization of ordinary terms. In future research, the author will attempt to add semantic features of some ordinary terms in the experiment to increase the disambiguation accuracy and approach the disambiguation accuracy of abbreviations, ultimately improving the overall WSD accuracy in biomedical domain.

5. Conclusion

To address the issue on ambiguous words in the biomedical domain, this paper extracts a large amount of relevant corpus from MEDLINE to realize word vector feature representation based on word2vec toolkits. The CNN model based on word embedding is adopted to classify the context and then map the context to the standard sense to disambiguate the words. Based on experimental results, the following conclusions are drawn:

(1) Word feature vector representation based on extracted corpus can contribute to WSD. The latent context feature

information of word vectors is essential in distinguishing different word senses. Moreover, the experiment suggests that the pre-setting of the word vector dimensionality considerably influences the final results and that a larger preset word vector dimensionality can increase the accuracy of WSD effects.

(2) The accuracy of WSD based on CNN significantly improves compared with that of the previous methods based on graph clustering and SVM. The CNN model can effectively extract context features from a deep layer. If the computing capacity permits, expansion of the input vectors can significantly improve the accuracy of WSD.

This paper mainly uses neural network to enhance features and improve WSD accuracy in biomedical domain. The author will further optimize the representation of context features and the network structure to increase WSD accuracy. Moreover, the author will try to use unannotated corpus for unsupervised WSD to expand the practical applications of the proposed method.

Acknowledgement

This work was supported by the Fundamental Research Funds for the Central Universities, South-Central University for Nationalities (CZQ14012 ,CZY15023).

References

[1] Pedersen, T. (2010). The effect of different context representations on word sense discrimination in biomedical texts. In: *Proceedings of the 1st ACM international health informatics symposium (IHI'10)*, pages 56-65. Arlington, VA, USA: ACM, November. 2010.

[2] Brody, S., Lapata, M. (2009). Bayesian word sense induction. In: *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2009)*, p. 103-111. Athens, Greece: Association for Computational Linguistics, March.

[3] Chasin, R., Rumshisky, A., Uzuner, O., Szolovits, P. (2014). Word sense disambiguation in the clinical domain: a comparison of knowledge-rich and knowledge-poor unsupervised methods. *Journal of the American Medical Informatics Association*, 21(5) 842-849.

[4] REN, K., REN, Y. (2015). Kernel Fuzzy C-Means Clustering for Word Sense Disambiguation in BioMedical Texts. *Journal of Digital Information Management*, 13 (6) 411-420.

[5] Navigli, R., Faralli, S., Soroa, A., De Lacalle, O., Agirre, E. (2011). Two birds with one stone: learning semantic models for text categorization and word sense disambiguation. In: *Proceedings of the 20th ACM international conference on Information and knowledge management (CIKM 2011)*, pages 2317-2320. Glasgow, United Kingdom: ACM, October. 2011.

[6] Agirre, E., Soroa, A., Stevenson, M. (2010). Graph-based Word Sense Disambiguation of biomedical documents. *Bioinformatics*, 26 (22) 2889-2896.

[7] McInnes, B. T., Pedersen, T., Liu, Y., Melton, G. B., Pakhomov, S. V. (2011). Knowledge-based method for determining the meaning of ambiguous biomedical terms using information content measures of similarity. *In:* Proceedings of the American Medical Informatics Association Annual Symposium Proceedings (AMIA 2011). p. 895-904. Washington, DC, USA :American Medical Informatics Association. October.2011.

[8] Jimeno-Yepes, A. J., McInnes, B. T., Aronson, A. R. (2011). Exploiting MeSH indexing in MEDLINE to generate a data set for word sense disambiguation. *BMC bioinformatics*, 12 (1) 223.

[9] Zhong, Z., Ng, H. T. (2010). It makes sense: A wide-coverage word sense disambiguation system for free text. *In:* Proceedings of the Annual Meeting of the Association for Computational Linguistics 2010 System Demonstrations, p. 78-83. Uppsala, Sweden: Association for Computational Linguistics, July.2010.

[10] Stevenson, M., Guo, Y., Gaizauskas, R., & Martinez, D. (2008). Disambiguation of biomedical text using diverse sources of information. *BMC bioinformatics*, 9(11) 1-11.

[11] McInnes, B. T., Pedersen, T., & Carlis, J. (2007). Using UMLS Concept Unique Identifiers (CUIs) for word sense disambiguation in the biomedical domain. In: Proceedings of the American Medical Informatics Association annual symposium proceedings (AMIA 2007). p. 533–537. Chicago, IL,USA: American Medical Informatics Association, November.2007.

[12] Yepes, A. J. J., Plaza, L., Carrillo-de-Albornoz, J., Mork, J. G., Aronson, A. R. (2015). Feature engineering for MEDLINE citation categorization with MeSH. BMC bioinformatics, 16(1) 1-12.

[13] Yepes, A. J. (2016). Higher order features and recurrent neural networks based on Long-Short Term Memory nodes in supervised biomedical word sense disambiguation. *arXiv preprint arXiv:1604.02506.*

[14] Kim, Y. (2014). Convolutional neural networks for sentence classification. *In*: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), p. 1746–1751. Doha, Qatar : Association for Computational Linguistics, October.2014.

[15] Mikolov, T., Chen, K., Corrado, G., Dean, J. (2013). Efficient estimation of word representations in vector space. *In:* Proceedings of the International Conference on Learning Representations Workshop(ICLR 2013), p. 1-12, Scottsdale, Arizona, USA: arXiv, June.2013.

[16] Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4) 193-202.

[17] Elman, J. L. (1990). Finding structure in time. *Cognitive science*, 14 (2) 179-211.

[18] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(8) 2493-2537.

[19] Kalchbrenner, N., Grefenstette, E., & Blunsom, P. (2014). A convolutional neural network for modelling sentences. In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL2014)*, pages 655-665, Baltimore, Maryland, USA: ACL, June. 2014.

[20] Zeng, D., Liu, K., Lai, S., Zhou, G., & Zhao, J. (2014). Relation Classification via Convolutional Deep Neural Network. *In: Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*, page 2335-2344. Dublin, Ireland: International Committee on Computational Linguistics , August.2014.

[21] Lewis, D. D. (1992). An evaluation of phrasal and clustered representations on a text categorization task. *In: Proceedings of the 15th annual international ACM conference on Research and development in information retrieval(SIGIR)*, pages 37-50, Copenhagen, Denmark: ACM. June.1992.

[22] Post, M., & Bergsma, S. (2013, August). Explicit and Implicit Syntactic Features for Text Classification. In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL2013)*, pages 866-872, Sofia,Bulgaria: ACL. Auguest.2013.

[23] Cover, T. M., & Thomas, J. A. (2012). *Elements of information theory*. John Wiley & Sons.

[24] Ng, A. Y. (2004). Feature selection, L 1 vs. L 2 regularization, and rotational invariance. In: *Proceedings of the twenty-first international conference on Machine learning(ICML 04),* pages 78. Banff, Alberta, Canada : ACM. July.2004.