doi:10.3772/j.issn.1006-6748.2015.01.010

Chinese word segmentation with local and global context representation learning[①]

Li Yan (李 岩)*, Zhang Yinghua **, Huang Xiaoping **, Yin Xucheng[②]*, Hao Hongwei **
(* School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, P. R. China)
(** Institute of Automation, Chinese Academy of Sciences, Beijing 100190, P. R. China)

Abstract

A local and global context representation learning model for Chinese characters is designed and a Chinese word segmentation method based on character representations is proposed in this paper. First, the proposed Chinese character learning model uses the semantics of local context and global context to learn the representation of Chinese characters. Then, Chinese word segmentation model is built by a neural network, while the segmentation model is trained with the character representations as its input features. Finally, experimental results show that Chinese character representations can effectively learn the semantic information. Characters with similar semantics cluster together in the visualize space. Moreover, the proposed Chinese word segmentation model also achieves a pretty good improvement on precision, recall and f-measure.

key words: local and global context, representation learning, Chinese character representation, Chinese word segmentation

0 Introduction

Until 2006, deep learning[1] was discussed much in machine learning, due to the poor training and generalization errors of the traditional neural network[2]. By now, deep learning has become a hot research point and made great progress in a wide variety of domains, such as hand-writing digits, human motion capture data, image recognition and speech recognition.

There is also rapid development in the natural language processing (NLP). Researchers have tried to use deep learning methods to solve problems in NLP for English language and made great contributions. In NLP, bag-of-words is one of the common traditional methods for representing texts and documents. It takes the statistic information of documents. However, it doesn't contain the context information and often makes a sparse structure which takes more representation space. Therefore, researchers[3-5] try to use vectors to represent the words with semantic. The vectoredrepresentations, called word embeddings, are trained for tasks in NLP. There are some versions of English word embeddings trained by several corpus and methods.

Unlike English, the smallest unit in Chinese language is Chinese character, and the words are combined with characters. However, all the above researches do not provide complete word embeddings or vectors in Chinese language, which is not able to support most of the Chinese natural language tasks. In this case, this work follows the related work in English NLP and builds a Chinese character representation model with local and global context representation learning. The experiment result shows that our character embeddings have strong representation capability on Chinese characters.

Chinese words are the smallest unit which can be used independently in Chinese natural language. Because of the continuous writing of Chinese characters, machines do not make sure where the boundaries are. Therefore, Chinese words segmentation is the primary work in Chinese NLP tasks. The proposed work is trying to figure out how to segment Chinese words based on the character representations. This paper proposes a Chinese word segmentation model with a neural network using character representations as input features. The experiment result shows that the proposed character embeddings play an important role in the Chinese word segmentation.

The rest of the paper is organized as follows. Related work is described in Section 1. Chinese character

[①] Supported by the National Natural Science Foundation of China (No. 61303179, U1135005, 61175020).

[②] To whom correspondence should be addressed. E-mail: xuchengyin@ustb.edu.cn

Received on Aug. 11, 2014

representation model and unsupervised training are presented in Section 2. Chinese word segmentation model is described in Section 3. Experiment results and some compared analysis on character representation and Chinese word segmentation are shown in Section 4. Finally, conclusions and discussions are drawn in Section 5.

1 Related work

As described above, Chinese word segmentation, which is the foundation task, plays an important role in NLP. However, the current Chinese word segmentation methods still have some key limitations, i. e., they may suffer from extracting of features and also the model of methods. In this section, the Chinese word segmentation methods are reviewed focusing on these two problems.

The traditional Chinese word segmentation methods are based on matching, which relies on a big vocabulary. If a Chinese character matches an entry in the vocabulary, the match is successful. Common matching methods contain forward maximum matching, backward maximum matching and minimum syncopation. However, with the new words emerging, the efficiency and accuracy of segmentation will be affected by new words out of vocabulary (OOV), which results in the problem of word boundary ambiguity.

Research turns to machine learning in Chinese word segmentation. A critical factor in Chinese word segmentation based on machine learning is the representation of characters. In traditional representation methods, words appeared in the training data are organized into a vocabulary, in which each word has an ID. The feature vector, the same length as the size of the vocabulary, has only one dimension activated, which is called one-hot representation. However, words under this representation method suffer from data sparsity, which increases the difficulty of model training. In testing phase, the model still suffers from the new words which are out of vocabulary.

Neural language models have shown to be powerful in NLP, which induces dense real-valued low-dimensional word embeddings using unsupervised approaches. Ref. [3] uses ranking-loss training objective and propose neural network architecture for natural language processing. Ref. [6] and Ref. [7] study language parsing and semantic analysis with neural language. Based on the work above, Ref. [4] combines local and global context to train English word embeddings. However, the way of Chinese word formation is different from English. In this paper, the neural language model is used to train Chinese character embeddings.

Chinese word segmentation task based on machine learning methods can be considered as assigning position labels to each character of given sentences. While traditional methods, like conditional random fields (CRFs)[8] and hidden markov model (HMM)[9], often use a set of specific features. It mainly lies in manual selection and extraction of features, which mostly depends on language sense. Refs[10,11] research on Chinese word segmentation with character representations. However, the training corpus for character representations of their segmentation models is not adequate.

2 Chinese character representation Learning

A description of the Chinese character representation model and a brief description of the proposed model' s training method are given in this section. The trained Chinese character representation will be the initial features for the Chinese word segmentation model.

2.1 Local and global context representation model

As is shown in Fig. 1, the purpose of the proposed model is to learn the semantical representations for characters from two levels: local context and global context. Local context is the character order sequence where the character occurs, while global context isthe document where the character order sequence occurs.

Given a short character sequence s and document d, the pair is defined as a positive sample(s,d). Otherwise, if the last character of the sequenceis is replaced with another character stochastically chosen from the vocabulary, it is defined as a negative sample (s^w, d), namely illogical language. The scores of positive sample and negative sample, named $n(s,d)$ and $n(s^w,d)$, are computed by our model. Therefore, the purpose is that the score of positive sample is greater than the score of any negative sample by a margin of 1. The loss function is expressed as

$$L = \sum_{s \in T} \sum_{w \in V} \max(0, 1 - n(s,d) + n(s^w,d)) \tag{1}$$

where T is the set of sequences and V is the vocabulary. The smaller the L is, the more reasonable the character representations are.

The architecture of the model consists of two neural networks respectively for the local context and global context. For the score of local context, the character sequence is represented by $s = [s_1, s_2, \cdots, s_m]$, where s_i is the vector representation of i_{th} character in the sequence. During training, all the embeddings in the vocabulary are learned and updated by a four-layer neural network with two hidden layers:

$$a = \tanh(W_1 \times s + b_1) \quad (2)$$

$$score_L(s) = W_3 \times \tanh(W_2 \times a + b_2) + b_3 \quad (3)$$

where s is the embedding sequence as input feature, $score_L$ is the output of the network, W_1, W_2 are the weight of the network, and b_1, b_2 are the bias of each layer.

Similar to the local score, the score of global context is also obtained by a three-layer neural network:

$$score_G(s,d) = W_{g2} \times \tanh(W_{g1} \times s_g + b_{g1}) + b_{g2} \quad (4)$$

$$s_g = [s_m; l] \quad (5)$$

where s_m is the last character of the sequence, l is the global context information of s_m which is from the document where the sequence occurs. Average weight function is used to compute l. The score of the entire model is the sum of two parts:

$$n(s,d) = score_L(s) + score_G(s,d) \quad (6)$$

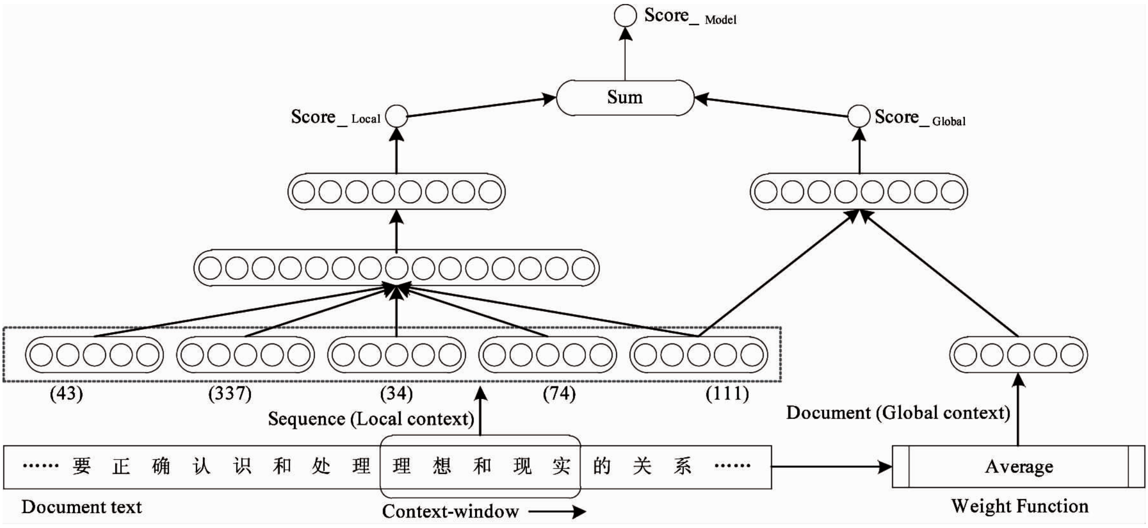

Fig. 1 Unsupervised architecture for Chinese character representation

2.2 Learning

The learning procedure tunes the model parameters by minimizing the specified loss function of the word segmentation model with L-BFGS. After training the corpus, it is found that the character embeddings move to good positions in the vector space.

3 Chinese word segmentation model

A Chinese word segmentation model is proposed in this section. First, the pre-processing of the corpus for training the model is profiled, followed by a description of Chinese word segmentation method, ending with a brief description of the model's learning.

3.1 Data proprecessing

The corpus for Chinese word training includes non-Chinese characters, such as numbers, English characters and punctuation marks. In order to ensure the authenticity of the results, the corpus is pre-processed before training. All numeric characters are replaced into a dedicated digital token 'NUMBER' and all English words, alphabet, punctuation marks are replaced by a special token, called 'UNKOWN', which also represents the new words in the future. This process may result in a certain loss in semantic information, but it can focus on the training of the Chinese characters.

3.2 Chinese word segmentation

A BMES label system is used for Chinese word segmentation. For a single-character word, the label is represented by 'S'. For a multi-character word, label 'B' is used to represent the first character of the word, label 'M' is used to represent the middle characters of the word, and label 'E' is used to represent the last character of the word.

In this work, the Chinese word segmentation model uses supervised learning structure. Discriminating the position label of each character in the sentence is a classification task. The current character and its context characters are selected as training features of the current character:

$$I = \{I_1, I_2, \cdots, I_w\} \in D^1 \times D^2 \times \cdots \times D^w \quad (7)$$

where D^k is the k^{th} character in the vocabulary, and w is the size of window where each $(w-1)/2$ word before and after the current character are selected. Characters in the window are replaced by corresponding character embeddings in the lookup layer and the embeddings of all context characters are concatenated as the input features, where L is the length of an embedding. Then, the input features are mapped into the hidden layer, with parameter, where h is the size of the hidden layer. The hidden layer is the input of softmax layer. There are four output nodes in the softmax layer which represents the probability of the current character position, respectively label 'B', 'M', 'E', 'S'. The maximum probability of the label is assigned to the current character. The architecture is shown in Fig. 2.

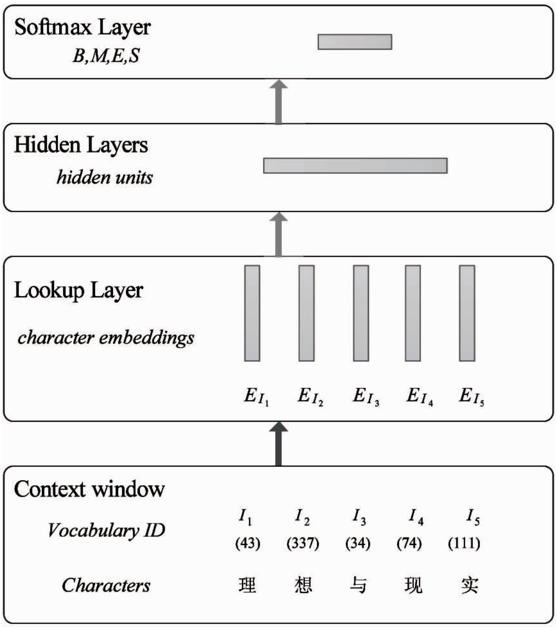

Fig. 2 Architecture for Chinese word segmentation

3.3 Learning

Given this model, activations for each node can be induced from the bottom up in the hidden layer by

$$h = \tanh(U \times x + b) \tag{8}$$

where the activation function is tanh, U is the weight between the input layer and the hidden layer, and b is the bias of the neural network.

The classifier model is trained by cross-entropy error:

$$p_i = \frac{\exp(V_i \times h + b_i)}{\sum_j \exp(V_j \times h + b_j)} \tag{9}$$

where V_j is the weight matrix for the j^{th} row of the classifier, b_j is the j^{th} bias of the classifier, and p_i is the i^{th} output unit.

Training is to optimize θ that minimizes the training corpus penalized log-likelihood:

$$E = -\sum_i y_i \log_{p_i}(x \mid \theta) \tag{10}$$

where θ contains the parameter U, V, and bias b, y is a 1-of-N encoding of the target class label and the parameter. Conjugate gradient descent (CGD) is used over the training data to minimize the objective.

4 Experiments

In this section, the character representations are shown in an intuitive way. Then, the results in Chinese word segmentation models trained by different sets of parameters are compared. Lastly, the model is compared with some current word segmentation tools.

4.1 Character representation learning

Baidu Encyclopedia is selected as the corpus to train the character representation models because of its wide range of Chinese word usages and its clean and regular organization of documents by topics. It contains 40GB original data from Baidu Encyclopedia which has 626238 websites including over 2.7 billion Chinese characters. The corpus covers most information described in Chinese language, such as, politics, philosophy, military, art, sports and science. In order to facilitate the training, numbers are converted into a '*NUMBER*' token. English words and punctuations are replaced by an '*UNKNOWN*' token. Then, a vocabulary with 18989 characters is organized from the corpus.

The model uses 50-dimensionality vectors to words of the vocabulary. For the local context neural network, 10-character windows of text are used as the input data. First and second hidden layers have 200 and 100 nodes respectively. For the global context neural network, 100 nodes are set in the hidden layer. The vectors are initialized stochastically before training. The whole training data is iterated for totally 10 times.

As is shown in Fig. 3(a), in order to present the vectors of the characters in a two-dimensional space, the dimensionalities of the vectors from 7200 most frequent characters are reduced from 50 to 2[12].

The characters are clustered into small groups by the *leader-follower* method. Eighty groups of characters are distributed in Fig. 3(b). Two example groups are shown in Fig. 4. It is found that characters with similar semantics gather together in one group after training.

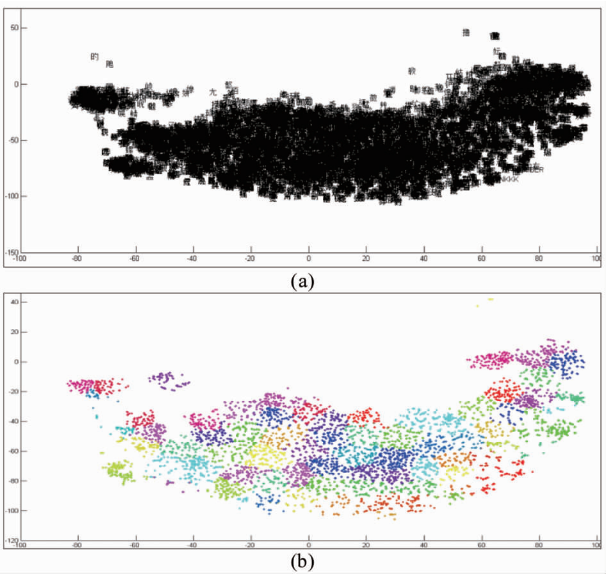

Fig. 3 The visualization of the distribution of character representation

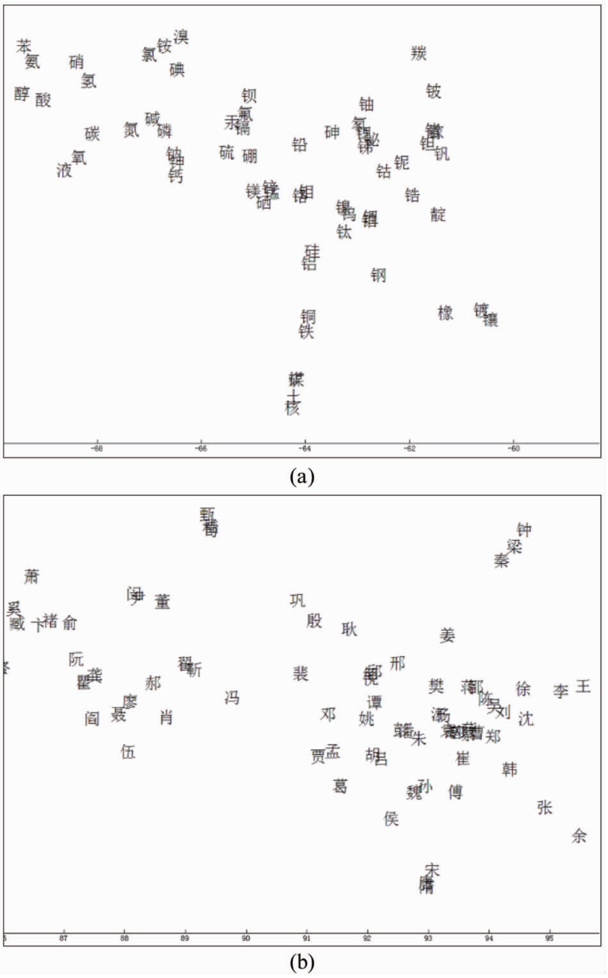

Fig. 4 Distributions of two example groups. Characters in (a) are about chemical substances and characters in (b) are about Chinese family names

4.2 Chinese word segmentation

The size of context window makes a major impact to the result of Chinese word segmentation. In this experiment, the results of models with 3, 5 and 7 characters as the context window are compared. The models use 18989 50-dimension character embeddings trained from Section 2 as input features.

This paper uses *SIGHAN* 2005 bakeoff dataset as the corpus for Chinese word segmentation. There are four different annotated corpuses in this dataset, from which two corpuses are in simplified Chinese, namely PKU and MSR dataset, respectively offered by Peking University and Microsoft Research. The corpuses have been divided into training dataset and testing dataset. The dataset named PKU is used in this work. The dataset contains training samples with 1,570,000 characters and testing samples with 140,000 characters. Our model is trained on the training set and evaluated on the testing set on a Linux Server with Intel(R) Xeon(R) 8-core 2.00GHz.

First, three different sizes of context windows are taken as the length of features, where 2, 4 and 6 context characters are respectively included beside current character. The three models contain one hidden layer with 200 neural nodes and uses CGD as the optimization to minimize the cost function.

With the increase of the size of context window, under the same iterations, the time of training is in ascent order, for 70 hours, 140 hours and 220 hours respectively. As is shown in Fig. 5, the performance of the model with 5-character context windows and 7-character context windows are better than the model with 3-character context windows. It shows that 5-character and 7-character context windows are more suitable for Chinese word segmentation.

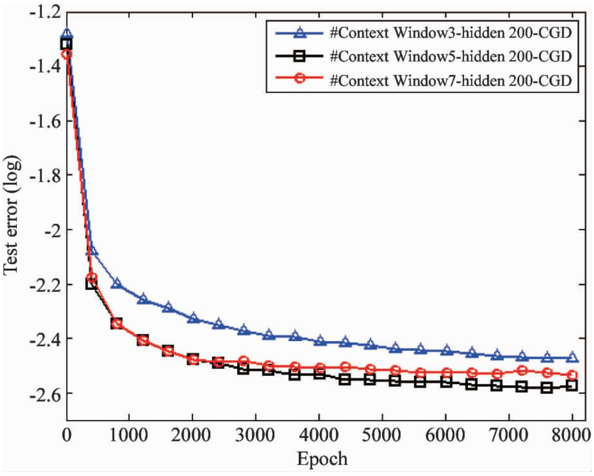

Fig. 5 Results of the models with different sizes of context windows

Then, 5 characters are used as the context window, along with 200 hidden layer nodes, and L-BFGS as the optimization instead of CGD. As is shown in Fig. 6, the error rate of the classifier with CGD drops faster than one with L-BFGS, the CGD optimization is more efficient than L-BFGS in this scale of parameter.

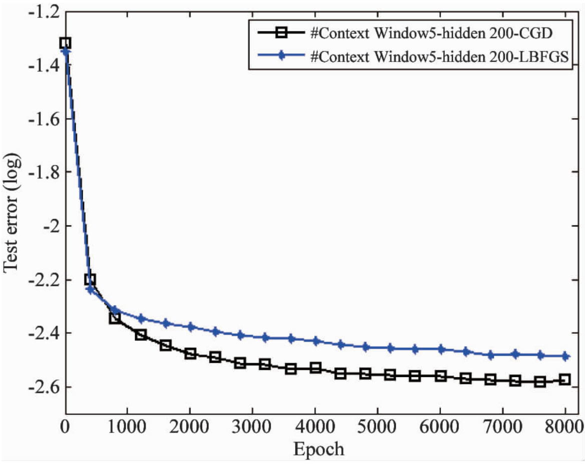

Fig. 6 Results of the models with CG and L-BFGS

Next, 7 characters are used as the context window and CGD as the optimization, along with 300 hidden-layer nodes, comparing with the model with 200 nodes. At last, the hidden layers of models with 5-character context windows and 7-character context windows are respectively replaced by two hidden layers, where there are 200 or 100 hidden-layer nodes and 250 or 200 hidden-layer nodes.

In Fig. 7, it is not obvious that increasing the number of the nodes of hidden layer can raise the efficient of training. Likewise, increasing the number of hidden layers also brings difficulty to train the classifier in Fig. 8. Due to increasing the number of parameters, it is difficult to converge the cost function, which will cause the error of the classifier fall into the local minimum or the classifier to be over fitting.

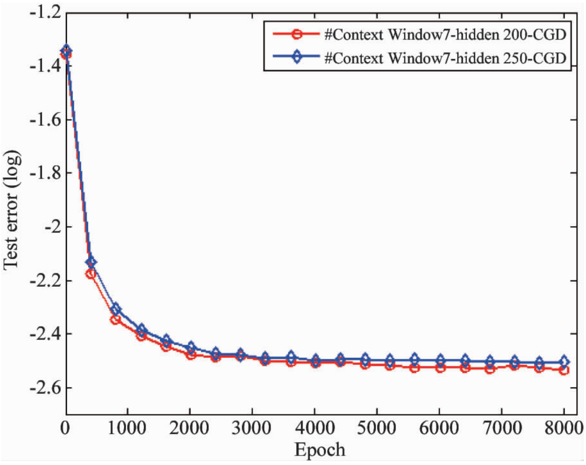

Fig. 7 Results of the models with different number of nodes of hidden layer

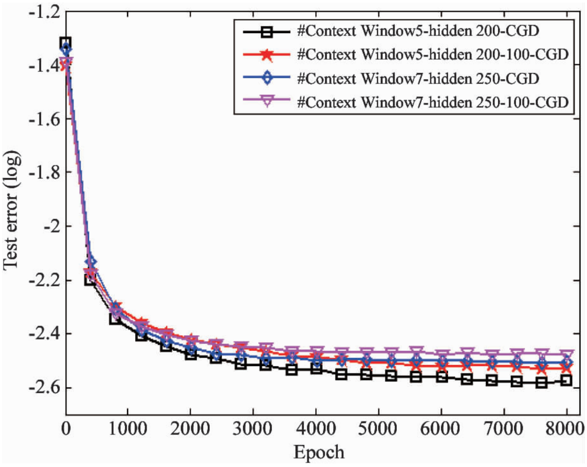

Fig. 8 Results of the models with different number of hidden layers

All the experiment results are listed in Table 1. Precision, recall and f-measure are used to measure the performance of the models. It is found that the Chinese word segmentation model with 5-character context windows and 200-node hidden layers trained with CGD gets the best performance among the experiments.

Table 1 Experiments of models with different sets of parameters

Context window	Hidden layer	Optimization	Precision	Recall	*f*
3	200	CGD	79.79	83.84	81.72
5	**200**	**CGD**	**90.59**	**89.94**	**90.26**
7	200	CGD	85.95	87.27	86.60
5	200	L-BFGS	89.24	88.51	88.87
7	250	CGD	86.90	85.65	86.27
5	200-100	CGD	89.45	89.44	89.44
7	250-100	CGD	87.04	84.34	86.57

Furthermore, comparsions of the performance are made between our model and word segmentation tools from Institute of Computing Technology, Chinese Academy of Sciences (SharpICTCLAS)[13], Harbin Institute of Technology (LTP _ cloud)[14] and PaodingAnalyzer[15]. As can be seen from Table 2, the recall, precision and *f*-measure of our model are better than the other current Chinese word segmentation tools with an obvious improvement on the same testing data.

Table 2 Performance(%) of models on PKU testing data

Tools	Precision	Recall	*f*
Our model	**90.59**	**89.94**	**90.26**
SharpICTCLAS	89.61	88.22	88.91
LTP-cloud	86.54	87.42	86.98
PaodingAnalyzer	72.01	70.67	71.33

5 Conclusion

An improved Chinese character representation model with local and global context information and average weight function is proposed in this paper. Using the representation model, the character embeddings are trained with a 2.7-billion-character corpus. Then, a neural network is used to train the Chinese word segmentation model with the character embeddings as the input features of the segmentation model. Finally, the result of the proposed Chinese word segmentation model is compared with some current Chinese word segmentation tools. Experimental results show that character embeddings trained by our representation learning model learns language information at semantic level, where input features may be better than ones initialized randomly. The Chinese word segmentation model with 5-character context windows and CGD optimization performs better than the other models with different sets of parameters, and our model is better than the compared word segmentation tools with an obvious improvement on precision, recall and f-measure.

There are still several main limitations of our technology for further research. First, due to time limitation, the result could be better if the iteration goes on. Second, how to improve the word segmentation model with more semantic information for a better result is the near future research. Third, how to accelerate the convergence of the cost function is also a challenge.

References

[1] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks. *Science*, 2006, 313(5786): 504-507

[2] Bengio Y. Learning deep architectures for AI. *Foundations and trends in Machine Learning*, 2009, 2(1): 1-5

[3] Collobert R, Weston J. A unified architecture for natural language processing: Deep neural networks with multitask learning. In: Proceedings of the 25th international conference on Machine learning, Helsinki, Finland, 2008. 160-167

[4] Huang E H, Socher R, Manning C D, et al. Improving word representations via global context and multiple word prototypes. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, Jeju, Korea, 2012. 873-882

[5] Turian J, Ratinov L, Bengio Y. Word representations: a simple and general method for semi-supervised learning. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, 2010. 384-394

[6] Socher R, Pennington J, Huang E H, et al. Semi-supervised recursive autoencoders for predicting sentiment distributions. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, Edinburgh, UK, 2011. 151-161

[7] Socher R, Lin C C, Ng A, et al. Parsing natural scenes and natural language with recursive neural networks. In: Proceedings of the 28th International Conference on Machine Learning, Bellevue, USA, 2011. 129-136

[8] Zhao H, Kit C. Scaling Conditional Random Field with Application to Chinese Word Segmentation. In: Proceedings of the 3rd International Conference on Natural Computation, Haikou, China, 2007, 5. 95-99

[9] La L, Guo Q, Yang D, et al. Improved viterbi algorithm-based HMM2 for Chinese words segmentation. In: Proceedings of the International Conference on Computer Science and Electronics Engineering, Hangzhou, China, 2012. 266-269

[10] Lai S W, Xu L H, Chen Y B, et al. Chinese word segment based on character representation learning. *Journal of Chinese information processing*, 2013, 27(5): 8-14

[11] Wu K, Gao Z, Peng C, et al. Text Window Denoising Autoencoder: Building Deep Architecture for Chinese Word Segmentation. In: Proceedings of the 2nd conference on Natural Language Processing and Chinese Computing, Chongqing, China, 2013. 1-12

[12] Maaten L, Hinton G E. Visualizing non-metric similarities in multiple maps. *Machine Learning*, 2012, 87(1): 33-55

[13] Zhang H P. ICTCLAS. http://ictclas.nlpir.org: CNZZ, 2014.

[14] Liu Y J. LTP_cloud. http://www.ltp-cloud.com: Research Center for Social Computing and Information Retrieval, 2014

[15] Wang Q Q. PaodingAnalyzer. https://code.google.com/p/paoding: Google Project Hosting, 2014

Li Yan, born in 1987. He is a Ph. D candidate. He received his B.S. degree from University of Science and Technology Beijing in 2009. His research interests include machine learning and natural language processing.