JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 24 No. 12, pp. 1-7, December 2019
https://doi.org/10.9708/jksci.2019.24.12.001

A Text Sentiment Classification Method Based on LSTM-CNN

Guangxing Wang*, Seong-Yoon Shin**, Won Joo Lee***

*Professor, Dept. of Information Technology Center, Jiujiang University, Jiujiang, China
**Professor, School of Computer Inf. & Comm. Eng., Kunsan National University, Gunsan, Korea
***Professor, Dept. of Computer Science, Inha Technical College, InCheon, Korea

[Abstract]

With the in-depth development of machine learning, the deep learning method has made great progress, especially with the Convolution Neural Network(CNN). Compared with traditional text sentiment classification methods, deep learning based CNNs have made great progress in text classification and processing of complex multi-label and multi-classification experiments. However, there are also problems with the neural network for text sentiment classification. In this paper, we propose a fusion model based on Long-Short Term Memory networks(LSTM) and CNN deep learning methods, and applied to multi-category news datasets, and achieved good results. Experiments show that the fusion model based on deep learning has greatly improved the precision and accuracy of text sentiment classification. This method will become an important way to optimize the model and improve the performance of the model.

▸Key words: Machine Learning, CNN, LSTM, Text Sentiment Classification Methods, Deep Learning

[요 약]

머신 러닝의 심층 개발로 딥 러닝 방법은 특히 CNN(Convolution Neural Network)에서 큰 진전을 이루었다. 전통적인 텍스트 정서 분류 방법과 비교할 때 딥 러닝 기반 CNN은 복잡한 다중 레이블 및 다중 분류 실험의 텍스트 분류 및 처리에서 크게 발전하였다. 그러나 텍스트 정서 분류를 위한 신경망에도 문제가 있다. 이 논문에서는 LSTM (Long-Short Term Memory network) 및 CNN 딥 러닝 방법에 기반 한 융합 모델을 제안하고, 다중 카테고리 뉴스 데이터 세트에 적용하여 좋은 결과를 얻었다. 실험에 따르면 딥 러닝을 기반으로 한 융합 모델이 텍스트 정서 분류의 예측성과 정확성을 크게 개선하였다. 본 논문에서 제안한 방법은 모델을 최적화하고 그 모델의 성능을 개선하는 중요한 방법이 될 것이다.

▸주제어: Machine Learning, CNN, LSTM, 텍스트 정서 분류 방법, Deep Learning

• First Author: Guangxing Wang, Corresponding Author: Won Joo Lee, Seong-Yoon Shin
*Guangxing Wang (wanggxrs@163.com), Dept. of Information Technology Center, Jiujiang University
**Seong-Yoon Shin (s3397220@kunsan.ac.kr), School of Computer Inf. & Comm. Eng., Kunsan National University
***Won Joo Lee (wonjoo2@inhatc.ac.kr), Dept. of Computer Science, Inha Technical College
• Received: 2019. 11. 12, Revised: 2019. 12. 06, Accepted: 2019. 12. 06.

I. Introduction

Natural language processing(NLP) has always been the focus of attention in the field of science and technology. The study of natural language processing helps scientists to extract useful features from natural language and analyze the meaning and sentiment expressed by these semantic features. Further explore and predict people's behavioral information. For example, popular product recommendation systems, speech translation systems, intelligent robot question and answer systems, etc., are typical applications of natural language processing research[1, 2]. Natural language processing is mainly used in spell checking, network information extraction, text sentiment classification, machine translation, spoken dialogue, and complex question and answer systems[3]. The research of NLP promotes the development of computer bionics, combining biological neural network systems with computer technology, enabling computer model biology neural network systems to input, feature, analyze, and output the result. Research on computer information processing models of neural network systems based on deep learning has become popular in recent years, including deep neural network (DNN), convolution neural network (CNN), recurrent neural network (RNN), and variants of RNN, that is, Long-Short Term Memory networks (LSTM). There are the most widely used in computer vision, mainly for image recognition and classification, target tracking, and automatic driving. The application of deep learning neural networks to natural language processing (NLP) has also achieved good results[4, 5].

In this paper, we start with the basic neural network CNN model, briefly introduce the CNN model, LSTM model, and conduct text sentiment classification experiments based on THUC's THUCNews dataset. Through experiments, we found that the LSTM-CNN model based on the neural network fusion model can significantly improve the accuracy and efficiency of text sentiment classification. The model training takes less time and the convergence speed is faster. Compared with the traditional emotional text distribution method, the LSTM-CNN neural network model performed best in the experiment.

This paper is structured as follows. Section I provides introduction. In Section II, our proposed method is introduced. Section III provides the experimentation. Section IV lists the conclusion and future work.

II. Proposed Method

In the study of natural language processing, most of the problems are related to the analysis of grammar, sentences, and semantics. Therefore, the analysis of sentences is inseparable from the analysis of the connection of contextual content. CNN has the characteristics of feature extraction. LSTM has the characteristics of memory context in time series. Therefore, combining the advantages of LSTM and CNN for text sentiment classification in natural language processing will produce better results.

The model we proposed consists of an initial LSTM layer that receives the word embedding for each token in the experimental data after data preprocessing. The token it outputs not only stores the information of the initial token, but also stores any previous tokens. In other words, the LSTM layer is generating a new encoding for the original input. The output of the LSTM layer then inputs to the desired local feature convolution layer, and the output of the final convolution layer will be aggregated into a smaller dimension, ultimately outputting the classification label of the text sentiment. In the past experiments, the LSTM-CNN model was mostly used for the binary category problem of the text. In this experiment, we mainly solved the multi-classification problem of the text. The structure of the LSTM-CNN model is shown in Fig. 1.

The LSTM-CNN model mainly consists of word embedding layer, LSTM layer, convolution layer, maxpooling layer, fully connected layer, and output layer.

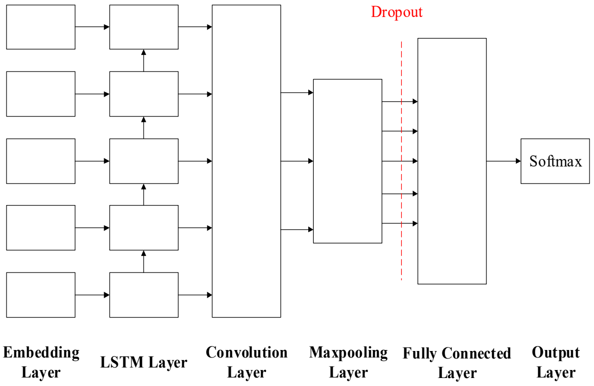

Fig. 1. LSTM-CNN model structure

Different from the CNN network, in the LSTM-CNN model, the text data is first converted into digital data through information extraction, and after pre-training, it is input into the word embedding layer by means of word vector mapping, that is the embedding layer, and then the data input is completed. The encoding of the statement is completed in the form of an index in the embedding layer to form an embedded matrix. The LSTM layer forms a text sequence of contextual content by inputting data in a time-based manner, which is, extracting useful text information to form a text sequence. The convolution layer extracts text features in a sequence of text. The maxpooling layer extracts the most important feature information of the text, and finally performs the category output of the text sentiment through the Softmax classification function. Dropout is set between the maxpooling layer and the fully connected layer to prevent over-fitting, and it is beneficial to the rapid convergence of the model and improves the accuracy of classification. The parameters of the LSTM-CNN model are shown in Table 1.

Table 1. LSTM-CNN Model Parameters Configuration Table

LSTM-CNN parameters configuration	
Parameter Name	Size
embedding_dim	100
vocab_size	10000
sequence_length	300
num_classes	10
hidden_dim	128
filters_size	[2,3,4]
num_filters	128
dropout_keep_prob	0.5
learning_rate	le-3
clip	5.0
batch_size	64
num_epochs	3
num_filters	128

III. Experiment

In this section, we introduce the datasets, experimental procedures and an analysis of the experimental results.

A. Dataset

The dataset used in the experiment is the THUCNews dataset[6]. THUCNews dataset is based on the historical data filtering and filtering of Sina News RSS subscription channel from 2005 to 2011. It contains 74 million news documents (2.19 GB), all in UTF-8 plain text format. After re-integration and division of 14 candidate classification categories. We selected ten categories as experimental data sets, a total of 65,000 pieces of data, of which 50,000 were used for model training, 10,000 were used for testing, and 5,000 were used for verification. The number of each category in the datasets are evenly distributed.

B. Experimental Environment

The experimental hardware environment is a desktop computer that can connect to the Internet, including Intel Core i7-4790, and the graphics processor Nvidia GeForce GTX 960 with 2GB memory equipped, 8GB RAM, 1TB Disk. Developed in python programming language and installed Keras deep

Table 2. Neural network model training and testing results. Where Improvement rate in test_acc is the percentage of the LSTM-CNN model test accuracy compared to the test accuracy of the corresponding model.

	Model Evaluation Index						
Model	Val_loss	Val_acc	Test_loss	Test_acc	Epoch	Training time	Improvement rate in test_acc
Logistic Regression		—	—	0.7772	—	0:01:40	25.68%
Random Forest		—	—	0.7624	—	0:01:45	28.12%
CNN	0.17	0.9595	0.11	0.9686	2300	**0:25:08**	0.85%
RNN	0.31	0.9272	0.18	0.9498	2300	5:55:21	2.84%
LSTM	0.29	0.9300	0.15	0.9568	7100	1:39:34	2.09%
LSTM_CNN	0.011	1.0000	0.012	0.9768	300	0:43:47	—

learning library with back-end as TensorFlow.

The experimental method is as follows. First of all, data preprocessing. Data preprocessing is performed by vectorizing the training text data set by means of de-stop words, establishing a dictionary, etc., and indexing each piece of data to form a form of a word vector matrix that can be input into the model. Then, the word2vec word vector model is used to pre-train the text data, and finally form a word vector mapping matrix, so that the data can be input into the word embedding layer of the model. Secondly, the model training. The pre-trained data is entered into the LSTM-CNN model for training. In the LSTM-CNN model, the data is indexed by the word embedding layer, and then the LSTM unit is used to generate the time text sequence, and then enters the convolution layer and the maximum pooling layer to extract the text features. Reduce the number of networks output and prevent overfitting by adding dropout parameter, and finally connect to the fully connected layer. Thirdly, output and testing. The LSTM-CNN model uses the Softmax function to classify, and then outputs and saves the trained model. Running the test using a trained model to get the loss rate and accuracy. Finally, the model evaluation. The same data set is used for training and testing in CNN, RNN, LSTM neural networks. The model is evaluated in terms of the loss rate and accuracy of the training, the loss rate and accuracy of the testing, the training time, and the number of sample batches to achieve the best training accuracy, and the final result is presented graphically and in tabular form.

C. Experimental Result

We use the THUCNews dataset to training and testing in CNN, RNN, LSTM and LSTM-CNN networks. The experimental result is shown in Table 2. From Table 2, it can be concluded that in this experiment the LSTM-CNN model performed best in terms of loss rate and accuracy, and the accuracy of the test was as high as 97.68%. The test accuracy is improved by 2.84% compared to the RNN model. The optimal training accuracy and the lowest loss rate were achieved when the training reached 300 batches, while the training batches of other models achieving the optimal accuracy and the lowest loss rate were more than two thousand. The prediction results are much better than traditional classifiers. Of course, the training time of the LSTM-CNN model is longer than that of the CNN model, but it is greatly shortened compared to the training time of other neural models. In the experiment, the confusion matrix of the four neural networks is shown in Fig. 2. From the comparison of the confusion matrix, the confusion matrix of the LSTM-CNN model shows the best prediction effect, because the color of the lattice above the diagonal of the matrix is deep, and the accuracy of the prediction is mostly close to 1.00. From the confusion matrix of LSTM-CNN, it can be seen that the predicted classification of finance, science and technology, fashion, and game is the most accurate, reaching 1.00, while the classification of home life is the lowest, only 92%. Figs. 3 and 4 are graphs of the accuracy and loss rate statistics in the tensor board during training. From Fig. 3, it is intuitively

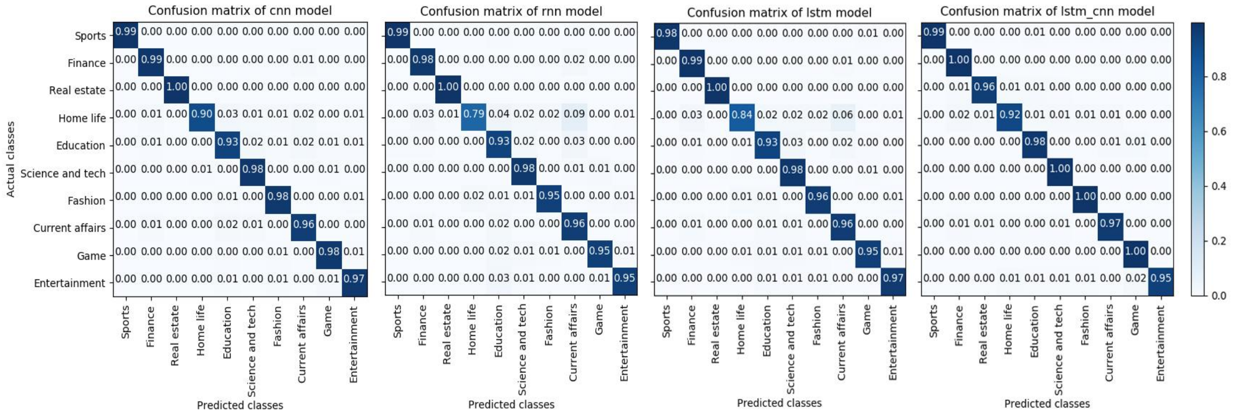

Fig. 2. Confusion matrix for the prediction results of the four neural network models. From left to right, the confusion matrix of CNN, RNN, LSTM, and LSTM-CNN models are followed. The left label is the classified real label, and the lower label is the predicted classification label. The darkest color of the grid above the diagonal in the confusion matrix indicates that the prediction is better.

shown that after about 300 batches of training, the accuracy of the LSTM-CNN model has reached a high level, indicating that the model training is efficient. Fig. 4 illustrates that the model gradient is fast and can converge quickly.

D. Analysis

The experimental data training has a batch size of 64, and each batch is trained in 100 batches, which are divided into 79 training batches. In the process of training, the best recording algorithm is used, that is, the performance of the current training result and the previous training result are checked and compared. If the performance of the training result is not improved after a certain number of training times, the training is ended early and the best training model is saved. Our analysis of the experimental results is as follows.

1) Compared with the traditional machine learning model, the deep learning-based neural network model has certain advantages in the classification effect of text sentiment classification.

2) Neural network model is generally better for text sentiment classification. The accuracy of LSTM-CNN based on model fusion is up to 97.68%, which is slightly better than the CNN model and better than LSTM model.

3) The best training accuracy was achieved when LSTM-CNN was trained about 300 times. Figs. 3 and 4 intuitively illustrate the fast convergence of the LSTM-CNN model, which is more efficient in obtaining accuracy.

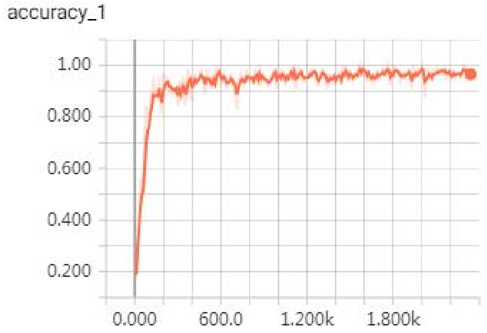

Fig. 3. Training accuracy curve of the LSTM-CNN model. The abscissa indicates the number of batches trained, and the ordinate indicates the accuracy of the prediction.

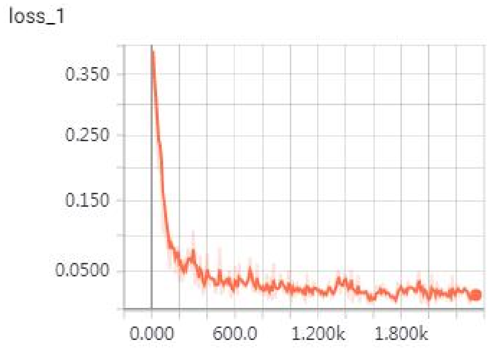

Fig. 4. Training loss rate graph of the LSTM-CNN model. The abscissa indicates the number of batches trained, and the ordinate indicates the predicted loss rate.

4) LSTM-CNN confusion matrix are shown in Fig. 4 intuitively illustrates that the classification of finance, science and technology, fashion, and game is the best, reaching 100%. But the lowest classes

of home life classification, only 92%. This indicates that the text contents of the home life news are short, and the features extracted by the model are limited, resulting in a decline in the classification effect.

IV. Conclusion and Future Work

This paper proposed a text sentiment classification method based on deep learning method model fusion. The method is compared with CNN, RNN, LSTM and two traditional text classification methods. A comparative experiment of text categorization was performed on the dataset THUCNews. The proposed method is to fuse the convolution neural network with the long-short term memory network to form the LSTM-CNN neural network model. The model exploits the performance of long-short term memory networks to process time-text sequences, which solves the problem of gradient disappearance and gradient explosion caused by RNN model, and draws on the advantages of CNN feature extraction in convolution neural networks. Compared with CNN, RNN and LSTM models, the performance of LSTM-CNN model text sentiment classification has been further improved. The prediction accuracy is up to 97.68%, and the training time is much shorter than RNN. Moreover, the LSTM-CNN model converges quickly and can obtain the best prediction accuracy faster, which greatly improves the efficiency of text classification. Compared with the traditional machine learning text sentiment classification method, the advantage of LSTM-CNN convolution neural network is more obvious, and the accuracy of the prediction has increased by 28.12%.

In this paper, we preprocessed the data and transformed text data into word vectors, which is also an important method to improve the prediction performance of the model. In addition, the model can be optimized from bidirectional LSTM, network depth, convolution kernel size, data expansion, gradient descent algorithm selection, etc., to further improve the accuracy of the model, which is what we will do in the future.

REFERENCES

[1] Chetan Arora, Mehrdad Sabetzadeh, Lionel Briand, Frank Zimmer, "Automated Checking of Conformance to Requirements Templates Using Natural Language Processing," IEEE Transactions on Software Engineering, Vol. 41, Issue 10, pp. 944-968, May, 2015. DOI: 10.1109/TSE.2015.2428709.

[2] Mohd Ibrahim, Rodina Ahmad, "Class Diagram Extraction from Textual Requirements Using Natural Language Processing (NLP) Techniques," 2010 Second International Conference on Computer Research and Development, pp. 200-204, 2010. DOI: 10.1109/ICCRD.2010.71.

[3] Sweta P. Lende, M. M. Raghuwanshi, "Question answering system on education acts using NLP techniques," 2016 World Conference on Futuristic Trends in Research and Innovation for Social Welfare(Startup Conclave), pp. 1-6, 2016. DOI: 10.1109/STARTUP.2016.7583963.

[4] C. Janarish Saju, A. S. Shaja, "A Survey on Efficient Extraction of Named Entities from New Domains Using Big Data Analytics," 2017 Second International Conference on Recent Trends and Challenges in Computational Models (ICRTCCM), pp. 170-175, 2017. DOI: 10.1109/ICRTCCM.2017.34.

[5] Tom Young, Devamanyu Hazarika, Soujanya Poria, Erik Cambria, "Recent Trends in Deep Learning Based Natural Language Processing," IEEE Computational Intelligence Magazine, Vol. 13, pp. 55-75, Nov. 2018. DOI: 10.1109/MCI.2018.2840738.

[6] THUCTC: An efficient Chinese text classification toolkit. [Online]. Available: http://thuctc.thunlp.org.

Authors

Guangxing Wang received his M.S. degree in Computer Application Technology from Huazhong University of Science and Technology, Wuhan, China in 2009. From 2016 to the present, he has been an associate professor in the Information Technology Center of Jiujiang University in China. His research interests include data science, information system, and artificial intelligence.

Seong-Yoon Shin received his M.S. and Ph.D degrees from the Dept. of Computer Information Engineering of Kunsan National University, Kunsan, Korea, in 1997 and 2003, respectively. From 2006 to the present, he has been a professor in the same department. His research interests include image processing, computer vision, and virtual reality.

Won Joo Lee received the B.S., M.S. and Ph.D. degrees in Computer Science and Engineering from Hanyang University, Korea, in 1989, 1991 and 2004, respectively. Dr. Lee joined the faculty of the Department of Computer Science at Inha Technical College, Incheon, Korea, in 2008, where he has served as the Director of the Department of Computer Science. He is currently a Professor in the Department of Computer Science, Inha Technical College. He has also served as the Vice-president of The Korean Society of Computer Information. He is interested in parallel computing, mobile computing, and data science, and cloud computing, artificial intelligence.