An Effective Text Classification Model Based on Ensemble Strategy

Zhu Hong\textsuperscript{1,a}, Jin Wenzhen\textsuperscript{1,b} and Yang Guocai\textsuperscript{1,*}

\textsuperscript{1}College of computer and information science, Southwest University, China

E-mail: \textsuperscript{a}zhhtdz@163.com, \textsuperscript{b}jwenzhen@163.com, \textsuperscript{*}lyuanj@swu.edu.cn

Abstract. Automatic text classification is a classic topic for natural language processing. Text classification research mainly focuses on feature representation of text documents or designing an efficient machine learning model. Although various approaches have been proposed to address these problems, they are still far from being solved. In this paper, we proposed a novel method called LAC\textsubscript{DNN} to achieve the text classification based on diverse feature representation approaches and classifiers. More specifically, LAC\textsubscript{DNN} firstly introduces a novel feature representation approach called LATW to extract feature information of the documents, which integrates the feature information extracted by LSI model, TF-IDF weighted vector space model (TF-IDF\_VSM), TF-IDF weighted word2vec (TF-IDF\_word2vec) and average word2vec (Avg\_word2vec), respectively. Secondly, it trains different classifiers including support vector machine, k nearest neighbor, logistic regression and convolutional neural networks based on the feature encoded by LATW. Finally, LAC\textsubscript{DNN} integrates these classifiers into an ensemble predictor to leverage complimentary information of feature representation methods and classifiers, and predict the topic of text documents. LAC\textsubscript{DNN} achieves superior performance with accuracy of 97.44\% and 97.43\% on the text datasets of Fudan and Netease news, respectively. Extensive experiments show that LAC\textsubscript{DNN} is prominent and useful for text classification.

1. Introduction

Text classification technology is gaining more attention along with the explosive growth of the number of text documents. Text representation is one of the two main research problems in the text categorization area. The more common text representations are vector space model (VSM)[1], Latent semantic index (LSI) [2] and latent dirichlet allocation (LDA) topic model [3]. Calculate the weight of the word vectors with TF-IDF at the same time. With the rapid development of deep learning in recent years, Mikolov et al.[4] proposed the word2vec model which is trained by neural networks to calculate the distributed vector representation of text words in 2013. Hu et al.[5] employed CNN to extract the semantic combination information from local words in sentences in 2014. Kim[6] proposed a classic CNN text classification model. These methods based on deep learning reduce the dimension of text representation and achieve the good results.

The design of the classifier model is another research problem in the text classification area. Different classifier models generally have different prediction performance. At present, the popular classification methods in machine learning contain naive bayes (NB) [7], logistic regression (LR) [8], k nearest neighbor (kNN) [9], support vector machine (SVM) [10] and CNNs. Article[11] indicates that the performance based on CNNs classifier is better than traditional machine learning classifiers for
text classification. Since we advocate using a number of different classifiers, which can obtain more information than a single classifier alone, to predict the category of documents.

All in all, in this paper, we proposed an ensemble approach called LAC_DNN to predict text category. More specifically, LAC_DNN firstly introduces a novel text representation model LATW, which can better represent text information by integrating the feature representation of LSI, TF-IDF weighted vector space (TF-IDF_VSM), the TF-IDF weighted word2vec (TF-IDF_word2vec), and the average word2vec (Avg_word2vec). Then, LAC_DNN uses this represented feature as the inputs of LR, kNN, SVM and three different CNNs. After that, it takes the outputs of these six classifiers as inputs of a small neural network to produce the final ensemble prediction. LAC_DNN not only combines the advantages of different text representation models, but also combines the diversity of different classifiers. Thus produces a robust and competent ensemble predictor. We performed experiments on Fudan and Netease new text datasets, which show that LAC_DNN is prominent and useful for text classification. The framework of LAC_DNN is shown in Figure 1.

![Figure 1. The framework of LAC_DNN model.](image)

2. Materials and Methods

2.1. Datasets

To quantitatively evaluate the performance of LAC_DNN, we collected two text classification datasets from Fudan text classification corpus and Netease News Dataset. The former dataset contains 8,300 texts divided into eight categories, in which the category of art, history, computer, environment, agronomy, economics, politics, and sports contains 732, 454, 1158, 1093, 1015, 1594, 1020 and 1234 texts respectively. Similarly, the latter dataset contains 24,000 texts with six categories, including auto, culture, economy, medicine, military, and sports, in which each category contains 4000 texts. These two corpora cover the situation of class-balance, which can better evaluate the performance of the model. In addition, we employed the evaluate metrics of Accuracy, Micro_F_1 and Macro_F_1 to evaluate the performance of LAC_DNN in this paper. These two metrics are defined as follows:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$  \hspace{1cm} (1)
where TP is the number of positive examples that are correctly classified by the classifier, TN is the number of negative examples that are correctly classified by the classifier, FP is the number of positive examples that are mistakenly classified by the classifier, FN is the number of negative examples that are mistakenly classified by the classifier.

### 2.2. LATW Representation Model

Different feature representation models can extract different feature information of text documents. The feature information generated by these representation models are complementary to each other. We introduced a novel feature representation method called LATW representation model, which combines the advantages of four basic models of text classification. LATW can better represent the characteristics of text than a single representation model for text classification. Specifically, LATW firstly selects the most appropriate feature dimension for each single model to represent text information. Nextly, LATW concatenates the vectors represented by these four models. The corresponding equations are shown as follows:

\[
\text{LATW} = \text{LSI}(d_i) \oplus \text{TF-IDF}_\text{VSM}(d_i) \oplus \text{Avg\_word2vec}(d_i) \oplus \text{TF-IDF\_word2vec}(d_i)
\]

where \(d_i\) is the \(i\)-th text in the dataset \(D\); \(\oplus\) is the vector concatenating operator; \(\text{LSI}(d_i)\), \(\text{TF-IDF}_\text{VSM}(d_i)\), \(\text{Avg\_word2vec}(d_i)\), and \(\text{TF-IDF\_word2vec}(d_i)\) are the feature vector of training text \(d_i\) extracted by LSI model, TF-IDF weighted vector space model, average word2vec model and TF-IDF weighted word2vec model, respectively.

### 2.3. Ensemble model LAC\_DNN

Aggregating the outputs of multiple predictors can generally improve the performance of a single predictor. To achieve a good integration effect, the individual predictor must be as accurate and diverse as possible. In this paper, LAC\_DNN selects SVM, kNN, and LR as three base classifiers. In addition, since convolutional neural networks (CNN) have achieved great progress in the field of natural language processing, LAC\_DNN also selects it as base classifier. Meanwhile, CNN is very sensitive to parameters, such as different size of the filters, the depth of neural network, or the number of filters will result in different results. Therefore, LAC\_DNN designs three CNNs with a different number and size of filters. The detailed parameters configurations of CNNs are shown in Table 2. To fuse these classifiers, LAC\_DNN uses a small depth neural network (S\_DNN) to synergize their outputs and to predict the category of text. The framework of LAC\_DNN is shown Figure 1.

### 3. Experiment

#### 3.1 Experimental setup

The dimension of feature vectors extracted by representation model has a great influence on predicting performance. Therefore, to investigate select the optimal feature dimension and ensure the performance of the classification, we evaluate the performance of LDA, LSI, TF-IDF\_VSM, Avg\_word2vec and TF-IDF\_word2vec with feature dimension of 100, 200, 300, 400, 500, 1000, 1500 based on SVM, respectively. The predict results in terms of accuracy are presented in Figure 2.
Figure 2. The experimental results of LDA, LSI, TF-IDF_VSM, Avg_word2vec, and TF-IDF_word2vec with different feature dimensions on Fudan and Netease datasets.

From Figure 2, we can see the TF-IDF_word2vec and TF-IDF_VSM obtain the best performance, but the classifier using LDA achieve the worst performance on both datasets. Therefore, we do not employ LDA model to extract the feature information of text documents. Meanwhile, we can observe that the performance of all models steadily increase as the feature dimension increasing and keep relatively stable when the feature dimension ≥ 500 based on these two datasets. To balance the efficiency and effectiveness, we select 500-dimension feature for each single model, and thus we can obtain a 2000-dimension feature vector represented by LATW model. Finally, we summarized the dimension setting of all feature representation models in this paper in Table 1.

Table 1. The dimension of different representation models.

| Model             | Dimension Choice | Experiment Range |
|-------------------|------------------|------------------|
| LDA               | ———             | 100,200,300,400,500,1000,1500 |
| LSI               | 500              | 100,200,300,400,500,1000,1500 |
| TF-IDF_VSM        | 500              | 100,200,300,400,500,1000,1500 |
| Avg_word2vec      | 500              | 100,200,300,400,500,1000,1500 |
| TF-IDF_word2vec   | 500              | 100,200,300,400,500,1000,1500 |
| LATW              | 2000             | ———             |

In this paper, we employed LR, SVM, kNN and three CNNs with different configuration for ensemble. We separately summarized the recommended parameters setup of these three CNNs in Table 2.

Table 2. The parameters of three CNN classifiers.

| Model             | Embeding_dim | Filter_size | Num_filters | Dropout | Batch | epochs |
|-------------------|--------------|-------------|-------------|---------|-------|--------|
| CNN1(Fudan)       | 400          | 3,4,5       | 100         | 0.5     | 64    | 100    |
| CNN2(Fudan)       | 400          | 2,3,4       | 120         | 0.5     | 100   | 120    |
| CNN3(Fudan)       | 200          | 3,4,5       | 100         | 0.5     | 64    | 150    |
| CNN1(Netease)     | 400          | 3,4,5       | 100         | 0.5     | 100   | 150    |
| CNN2(Netease)     | 400          | 2,3,4       | 120         | 0.5     | 100   | 100    |
| CNN3(Netease)     | 200          | 3,4,5       | 100         | 0.5     | 100   | 120    |

3.2 Results of LAC_DNN
Considering the number of samples used in this work, 5-fold and 10-fold cross validation is respectively adopted for two datasets to reduce the impact of data dependency and avoid the risk of over-fitting. Table 3 and Figure 3 report the results of LAC_DNN on two datasets.
Figure 3. The accuracy of different models in Fudan and Netease datasets.

Table 3. The results of LAC_DNN and other classifiers.

| Method                  | Fudan Accuracy (%) | Netease Accuracy (%) |
|-------------------------|--------------------|----------------------|
| LATW+KNN                | 94.28              | 93.55                |
| LATW+LR                 | 95.79              | 95.54                |
| LATW+SVM                | 95.86              | 95.58                |
| char-CNN[12]            | -                  | 95.00                |
| LATW+CNN                | 97.14              | 96.22                |
| LATW+KNN+LR+SVM+DNN     | 96.84              | 95.71                |
| LAC_DNN                 | 97.44              | 97.43                |

- **LAC_DNN** achieves superior prediction performance with an average accuracy of 97.44%, 97.43% on Fudan and Netease news text datasets, respectively. The reason is that different classifiers may have different focus for the same feature, but LAC_DNN leverages the complimentary information of LR, SVM, kNN, and CNNs by integrating these classifiers into a small deep neural network, which result in the better prediction performance.

- The prediction performance of using single CNN achieves the best performance among comparing methods (except LAC_DNN), these results indicate that CNNs can better represent the characteristics of text.

- The prediction performance of using single traditional classifier also achieve good results with accuracy, respectively. However, compared with the performance of LAC_DNN, the accuracy of kNN, LR and SVM are 3.16%, 1.65%, 1.58% and 3.88%, 1.89%, 1.85% lower than that, respectively. The accuracy of char-CNN is lower 2.43% than LAC_DNN on Netease dataset. From these results, we can conclude that using single traditional classifier can not sufficient extract the feature information of text documents.

- The performance of combined model with LR, kNN, and SVM is better than the single traditional classifier but worse than LAC_DNN and CNN. The reason is that the combined model combine the advantage of different traditional classifiers. However, it is still not fully extracted the feature information of text documents.

Based on these experimental results, we can conclude that LAC_DNN can more effectively predict text category than other comparing methods and is prominent and useful for text classification.

3.3 Contribution of LATW Text Representation Model

To further investigate the contribution of the novel LATW text representation model, we separately trained SVM based on the feature representation extracted by LATW, LDA, LSI, TF-IDF_VSM, Avg_word2vec, TF-IDF_word2vec, [13] and [14] on two text datasets. The results are reported in
Tabel 4. From these results, we can conclude that the LATW model can more sufficiently capture the feature information of text documents for text classification.

Table 4. F1 values for different representation models(%)  

| Method          | Fudan Micro$_F_1$ | Fudan Macro$_F_1$ | Netease Micro$_F_1$ | Netease Macro$_F_1$ |
|-----------------|-------------------|-------------------|---------------------|---------------------|
| LDA             | 88.78             | 86.10             | 88.03               | 88.09               |
| LSI             | 95.95             | 95.31             | 94.39               | 94.40               |
| TF-IDF_VSM      | 95.66             | 94.97             | 94.25               | 94.26               |
| Avg_word2vec    | 95.67             | 94.84             | 91.70               | 91.73               |
| TF-IDF_word2vec | 94.91             | 93.87             | 93.39               | 93.40               |
| [13]            | -                 | 80.32             | -                   | -                   |
| [14]            | 87.09             | 84.61             | -                   | -                   |
| LATW            | **95.85**         | **96.40**         | **95.54**           | **95.55**           |

3.4. Results of different CNNs  
To further investigate the effectiveness of integrating different CNNs, we separately trained three CNNs with different configurations based on LATW representation feature and adopted 10-fold cross-validation on Fudan and Netease new datasets. The results are reported in Figure 4.

From the Figure 4, we can observe that CNNs with different size of the filters, the number of filters, the depth of neural network, and word embedding dimensions achieve different prediction performances, respectively. Based on these results, we can conclude that CNNs is very sensitive to network parameters and CNNs with different configurations can achieve different effectiveness. Thus, CNNs with different configurations can be used as base classifiers for ensemble.

Figure 4. LTAW model combines with different CNNs classification accuracy.

4. Conclusion  
In this paper, we proposed an effective ensemble strategy called LAC_DNN. LAC_DNN firstly extracts feature information of text document by LSI model, TF-IDF_VSM, TF-IDF_word2vec, and Avg_word2vec, respectively. Given these feature informations are complement to each other, LAC_DNN introduces a novel LATW representation method to integrate these complementary information. Nextly, LAC_DNN employs SVM, kNN, LR as basic classifiers. Meanwhile, it also uses three CNNs with different configurations as basic classifiers. To integrate these predictors, LAC_DNN uses a small deep neural network to predict category of text. Experiments on Fudan dataset and Netease New dataset show that LAC_DNN achieves some improvements compared with other traditional methods. These results indicate that our method is prominent and useful for text classification.
References

[1] S.-h. Liu, M. Dong, H.-j. Zhang et al., An approach of multi-hierarchy text classification based on vector space model, Journal of Chinese Information Processing, vol. 3, no. 16, pp. 8C14, 2002.

[2] W. Zhang, T. Yoshida, and X. Tang, TFIDF, LSI and multi-word in information retrieval and text categorization, in Systems, Man and Cybernetics, 2008. SMC 2008. IEEE International Conference on. IEEE, 2008, pp. 108C113.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan, Latent dirichlet allocation, Journal of machine Learning research, vol. 3, no. Jan, pp. 993C1022, 2003.

[4] T. Mikolov, K. Chen, G. Corrado, and J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv:1301.3781, 2013.

[5] B. Hu, Z. Lu, H. Li, and Q. Chen, Convolutional neural network architectures for matching natural language sentences, in Advances in neural information processing systems, 2014, pp. 2042C2050.

[6] Kim Y. Convolutional Neural Networks for Sentence Classification[J]. Eprint Arxiv, 2014.

[7] A. McCallum, K. Nigam et al., A comparison of event models for naive bayes text classification, in AAAI-98 workshop on learning for text categorization, vol. 752, no. 1. Citeseer, 1998, pp. 41C48.

[8] T. Pranckevicius and V. Marcinkevicius, Application of Logistic Regression with part-of-the-speech tagging for multi-class text classification, in Advances in Information, Electronic and Electrical Engineering (AIEEE), 2016 IEEE 4th Workshop on. IEEE, 2016, pp. 1C5.

[9] K. Shi, L. Li, H. Liu, J. He, N. Zhang, and W. Song, An improved knn text classification algorithm based on density, in Cloud Computing and Intelligence Systems (CCIS), 2011 IEEE International Conference on. IEEE, 2011, pp. 113C117.

[10] D. L. Guo, X. M. Liu, and Q. S. Zheng, Internet short-text classification method based on cnns, Computer & Modernization, 2017.

[11] C. N. Kamath, S. S. Bukhari, and A. Dengel, Comparative study between traditional machine learning and deep learning approaches for text classification, in DocEng. ACM Symposium on Document Engineering (DocEng-2018), August 28-31, Halifax, Nova Scotia, Canada. ACM, 2018.

[12] X. Zhang, J. Zhao, and Y. Lecun, Character-level convolutional networks for text classification, pp. 649C657, 2015.

[13] F. Ren and C. Li, Hybrid chinese text classification approach using general knowledge from baidu baike: Hybrid chinese text classification approach, Ieej Transactions on Electrical & Electronic Engineering, vol. 11, no. 4, pp. 488C498, 2016.

[14] Z. Wan, M. Wang, J. Zuo, and J. Wan, Feature selection combined with the global and local information(glfs), Journal of Shandong University, vol. 51, no. 5, pp. 87C93, 2016.