An Effective Approach for Chinese News Headline Classification Based on Multi-Representation Mixed Model with Attention and Ensemble Learning

Zhonglei Lu[1], Wenfen Liu[2(✉)], Yanfang Zhou[1], Xuexian Hu[1], Binyu Wang[1]

[1] State Key Laboratory of Mathematical Engineering and Advanced Computer, Zhengzhou, Henan, China

[2] School of Computer Science and Information Security, Guangxi Key Laboratory of Cryptogpraphy and Information Security, Guilin University of Electronic Technology, Guilin, Guangxi, China

{lzl_xd6j,zyf_xd6j,hxx_xd6j,wby_xd6j}@163.com
liuwenfen@guet.edu.cn

Abstract. In NLPCC 2017 shared task two, we propose an efficient approach for Chinese news headline classification based on multi-representation mixed model with attention and ensemble learning. Firstly, we model the headline semantic both on character and word level via Bi-directional Long Short-Term Memory (BiLSTM), with the concatenation of output states from hidden layer as the semantic representation. Meanwhile, we adopt attention mechanism to highlight the key characters or words related to the classification decision, and we get a preliminary test result. Then, for samples with lower confidence level in the preliminary test result, we utilizing ensemble learning to determine the final category of the whole test samples by sub-models voting. Testing on the NLPCC 2017 official test set, the overall F1 score of our model eventually reached 0.8176, which can be ranked No. 3.

Keywords: News Headline; Short Text; Classification; Multi-Representation; Ensemble Learning

1 Introduction

Chinese news headlines classification faces great challenges for short length, less information, weak information description, scattered themes and big noise, which cause much difficulty in characteristic extraction. With the strong capability of automatic feature extraction, deep learning has become the dominant means of short or very short text classification in recent years. Kim et al. [1] introduced a simple Convolutional Neural Network (CNN) with single convolution layer, which achieved state-of-the-art performance in several NLP tasks, such as sentiment analysis, question classification etc. Lai et al. [2] proposed Recurrent Convolutional Neural Network (RCNN) to model text classification task on Fudan set, which achieved better performance than CNN. Then character-level convolutional networks (ConvNets) [3] was proposed to classify Chinese news corpus, and obtained better result. Recently, Zhou et al. [4] presented Compositional Recurrent Neural Networks for Chinese short text classification, and got state-of-the-art results.

Under the current deep learning paradigm, there are still two problems in related work. On the one hand, errors in word segmentation easily lead to incorrect or incomplete semantic representation [4], and the Out-Of-Vocabulary (OOV) problem seriously affects the performance of classifiers[1]. On the other hand, there are less targeted means for the weak feature samples, resulting in poor performance.

To solve the above problems, we propose a multi-representation mixed model with attention and a targeted ensemble learning strategy. For problem (1), we integrate character-level feature into word-level feature to obtain headlines representation. The missing semantic information by the error of word segmentation will be constructed; meanwhile, the wrong semantic relevance will be reduced. Considering the strong correlation between certain keywords and classification results, this paper introduces attention mechanism on the basis of multi-representation mixed model. For problem (2), we present two strategies for different testing samples, with multi-representation mixed model attached attention for all testing samples first. For samples with lower prediction confidence, we combine votes from multiple complementary sub-models. Experiments on the NLPCC 2017 Task two datasets show that, the proposed method puts up a good performance, effectively alleviating the influences brought by errors in word segmentation and managing weak feature samples.

The paper is organized as follows. Section 2 describes our multi-representation mixed model with attention. Section 3 introduces the targeted ensemble learning strategy. Experimental results and discussion are reported in Section 4. Finally, we draw some conclusions and give the future works.

2 Multi-representation Mixed Model Based on Attention Mechanism

Word segmentation is the first step in Chinese natural language processing, and errors caused by word segmentation can be transmitted to the whole deep neural networks. In order to reduce the impact of word segmentation and improve the overall performance of Chinese news headline classification system, we propose a mixed model of character-level and word-level features based on BiLSTM. By integrating character-level feature into word-level feature, the missing semantic information by the error of word segmentation will be constructed; meanwhile the wrong semantic relevance will be reduced. At the same time, analysis shows that factors determining category of headlines only relate to certain key words or characters, rather than the all. Therefore, this paper introduces attention mechanism to allocate weights to each word or character in the headline, highlighting the key ones. In summary, we propose a word-level and character-level representation mixed model based on attention mechanism which

[1] According to the SIGHAN (http://www.sighan.org/) Bakeoff data evaluation results, the loss of word segmentation caused by OOV is at least 5 times greater than word sense ambiguation.

consists of look-up layer, mixed encoding layer, attention layer and softmax classifier. The structure is shown in Figure 1.

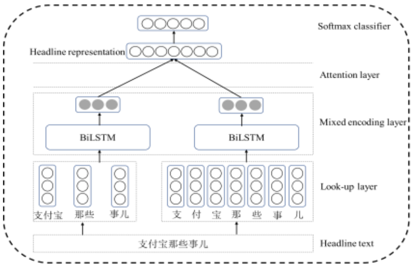

Fig. 1. The structure of multi-representation mixed model based on attention mechanism

2.1 Look-up Layer

Look-up table, a huge word embedding matrix, is the first layer of our model. Each column, which is *N*-dimensional, corresponds to a word. Given a dictionary D, that is extracted from the training corpus, we can construct a $N\times|D|$-dimensional matrix as the look-up table M. E_{w_i}, the column vector of M in the index of i, is the word embedding for word w_i. As a result, this component maps an input word sequences $\{w_1, w_2, \dots, w_n\}$ into a series of word embeddings $\{E_{w_1}, E_{w_2}, \dots, E_{w_n}\}$. This layer contains two look-up tables for characters and words. If pre-trained, we can obtain M from large unlabeled corpus by word2vec[2]. Otherwise, we randomly initialize it.

2.2 Mixed Encoding Layer

This part is based on Recurrent Neural Network (RNN) [5], which is an extension of Feedforward Neural Network (FNN) in time series. It is widely used in machine translation [6], automatic text summarization [7] etc. Unfortunately, the traditional RNN is still hard to apply in practice due to the vanishing and exploding gradient problems [8] during the back propagation training stage. Long Short-Term Memory (LSTM) [9] solves this problem by a more complex internal structure which allows it to remember information for either long or short terms. The structure is shown in Figure 2.

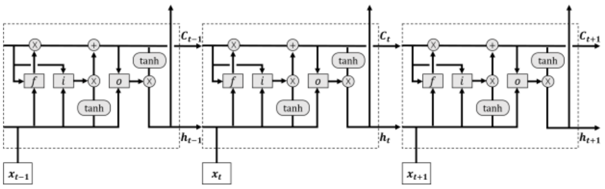

Fig. 2. The structure of LSTM

[2] http://code.google.com/p/word2vec/

Given the input sequence $(x_1, x_2, \ldots, x_T)$, we can get the hidden layer states $(h_1, h_2, \ldots, h_T)$ and the memory states $(C_1, C_2, \ldots, C_T)$ as follows.

$$i_t = \sigma(W_i x_t + W_i h_{t-1} + b_i) \tag{1}$$

$$f_t = \sigma\left(W_f x_t + W_f h_{t-1} + b_f\right) \tag{2}$$

$$o_t = \sigma(W_o x_t + W_o h_{t-1} + b_o) \tag{3}$$

$$\tilde{C}_t = \tanh(W_c x_t + W_c h_{t-1} + b_c) \tag{4}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{5}$$

$$h_t = o_t \odot \tanh(C_t) \tag{6}$$

Where i, f and o are input, forget and output gate, respectively. x, h and C represent input layer, hidden layer and memory cell, respectively. W and b are weight matrix and bias, namely network's parameters. σ is a sigmoid function, and $\odot$ is element-wise multiplication.

LSTM only encodes the above information, while the following information is equally important to the characterization of the whole semantics. In order to better represent headlines, we propose to use BiLSTM, which reads the headline in both directions with 2 separate hidden layers: the forward and the backward. Thus it can be trained with all the information from history or future for richer semantic representation.

$$\overrightarrow{h}_t = LSTM\left(x_t, \overrightarrow{h}_{t-1}\right) \tag{7}$$

$$\overleftarrow{h}_t = LSTM\left(x_t, \overleftarrow{h}_{t+1}\right) \tag{8}$$

We summarize the information from the forward and the backward hidden states by concatenating them, i.e.

$$h_t = \left[\overrightarrow{h}_t, \overleftarrow{h}_t\right] \tag{9}$$

By this way, the hidden state h_t contains the information of headlines not only in the original order but also in the reverse one. This improves the model's performance in memory. We encode the headlines on character-level and word-level, obtaining semantic vector h_c, h_w, respectively. Thus, the preliminary mixed representation h of headline can be expressed as follows.

$$h = [h_c, h_w] \tag{10}$$

2.3 Attention Layer

Attention mechanism was first used in Neural Machine Translation (NMT). As generating a target language word, not all words contribute equally to the representation of the sentence meaning. Thus, Bahadanau et al. [10] introduced attention mechanism to extract such words which are important to the meaning of the sentence and aggregate the representation of those informative words to form a sentence vector dynamically.

The same is true for Chinese news headline classification. We adopt the attention mechanism to focus on some key words that are strongly correlated to the decision of classification. We can see the attention mechanism in Figure 3.

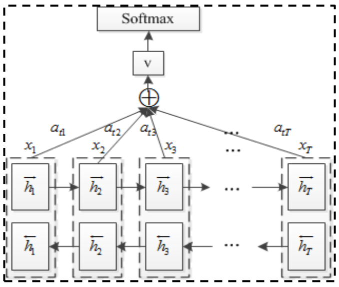

Fig. 3. The structure of attention mechanism

$$u_t = tanh(W_w h_t + b_w) \tag{11}$$

$$a_t = \frac{exp(u_t u_w)}{\sum_{s=1}^{T} exp(u_s u_w)} \tag{12}$$

$$v_w = \sum_{t=1}^{T} a_t h_t \tag{13}$$

We get u_t as a nonlinear representation of h_t for each word through one-layer Multi-layer Perception (MLP). Then we measure the importance of the word as similarity of u_t with a word level context vector u_w, and get a normalized importance weight a_t through a softmax layer. Finally, we compute the word-level headline vector v_w that summarizes all the information of words in a headline. We can obtain the character-level headline vector v_c in a similar way. Consequently, we update headline representation in Eq.(10) to h in Eq.(14).

$$h = [v_c, v_w] \tag{14}$$

2.4 Softmax Classifier

Based on the above work, we can obtain the headline representation vector for classification. To be specific, given class number C, the headline representation vector h is mapped to a real-valued vector $\{e_1, e_2, \dots, e_C\}$ by a linear layer. Then we add a softmax layer to map each real value to conditional probability, which is computed in Eq. (15).

$$P_i = \frac{exp(e_i)}{\sum_{j=1}^{C} exp(e_j)} \tag{15}$$

where $\sum_{i=1}^{C} P_i = 1$.

We use the cross entropy as the loss function, calculating it through back propagation and update parameters with Stochastic Gradient Descent (SGD). The whole model is finally trained end-to-end [11] with supervised classification task.

3 Combination Model

Base on the multi-representation mixed model with attention, we analyze the confidence level distribution of the correct and error predictive samples in development set. The results are shown in Table 1.

Table 1. The confidence level distribution of samples in development set

Total number of error samples	6847			
Confidence of error samples	>0.8	>0.85	>0.9	>0.95
	1051	821	589	218
Total number of correct samples	29153			
Confidence of correct samples	<0.95	<0.9	<0.85	<0.80
	10053	7383	4739	2612
Notes：The accuracy of our multi-representation mixed model with attention is about 0.8098, and there are 36,000 samples in development set.				

Statistical results show that, only 15.34% error samples have predictive confidence above 0.80, while only 8.96% correct samples below 0.80. In other words, the rate of error prediction in samples with confidence above 0.80 is very low, and in samples with confidence below 0.80 is relatively high. Furthermore, the proportion of samples with confidence below 0.80 is 23.36, which seems a big scale. So, we propose to construct combination model to vote for the samples with low confidence.

3.1 Model Selection

Besides our multi-representation mixed model with attention, we select N-BoW and CNN [1] as sub-models according to the principle of "difference meets complementation" in feature extraction. Table 2 lists the differences of the N-BoW, CNN and RNN models in text modeling.

3.2 Strategy in Use

The N-BoW and CNN models are trained using the same training data. After obtaining three trained sub-models, we first predict on the whole testing set (TestData) using the single multi-representation mixed model with attention(we just call it CA-BiLSTM). Samples with lower confidence are screened as TestData-2. Then we test our three sub-models on TestData-2, getting three results Result-1, Result-2, Result-3. Finally, we use a simple voting mechanism to determine the category. Specially, for a sample $x \in$ TestData-2, we have three predictive results $R_1(x)$, $R_2(x)$, $R_3(x) \in \{c_1, c_2, \dots, c_{18}\}$, c_i denotes the category. We take the more one as the category of x. When $R_1(x)$, $R_2(x)$ and $R_3(x)$ are different from each other, the result from our multi-representation mixed model with attention is prevailing. Figure 4 shows the strategy in use.

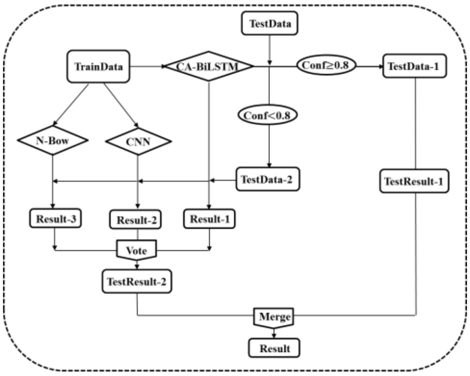

Fig. 4. The structure of our combination model

Table 2. Comparison of models in text modeling

Model	Advantages	Disadvantages
N-BoW	It trains very fast and is very suitable for online tasks which are strict on the response time. Its effect is acceptable.	It seriously relies on word segmentation. It is easy to introduce noise; meanwhile its robustness is weak. Lost sequence order, and no use of context information. More network parameters.
CNN	It can model local feature information, and pooling operation can achieve greater span information modeling, extracting the most significant features. It has high efficiency in long text modeling. It trains fast due to parallel computing.	Due to the size of windows and other issues, it has higher requirements on parameters tune. It cannot capture the long range dependency of words in a text, and can only process the context in a local window (i.e. the window size of convolution filter), losing some semantic information.
RNN	It models the whole sequence order. It can use history and future information to fully model context information and discover long-distance semantic dependencies. It has less network parameters.	It trains slowly.

4 Experiments

4.1 Datasets

The shared task data[3] is collected from several Chinese news websites, such as toutiao, sina etc. There are 18 categories in total. All the headlines are segmented by jieba[4]. Most of title sentence character number is less than 40, with a mean of 21.05. Title sentence word length is even shorter, most of which is less than 20 with a mean of 12.07.

4.2 Experiment Settings

In our experiments, each hidden layer has 400 units. The input is a 200-dimensional vector denotes word embedding or character embedding. We use the longest character-level (40) and word-level (20) length of headlines in the data to unfold all the BiLSTM networks. According to Greff et al. [12] on the experience of parameters setting research, the learning rate is initialized to 0.001, and decay rate [13] per 500 training steps is 0.9. Instead of using a fixed number of epochs, we apply an early stop technique [14] in which the system stops training whenever the F1 score of the development set does not increase after 5 epochs. The specific model parameters are set as shown in Table 3.

Table 3. Parameter configurations of our model

Parameters	Configurations
Maximum length of text (Character)	40
Maximum length of text (Word)	20
Vocabulary size(Character)	6,000
Vocabulary size(Word)	100,000
Embedding dimension(Character)	200
Embedding dimension(Word)	200
Hidden layer size	400
Max epoch	50
Mini-batch size	32
Probability of dropout	0.7
Early stop epoch	5
Learning rate	0.001
Decay rate	0.9

[3] https://github.com/FudanNLP/nlpcc2017_news_headline_categorization

[4] https://github.com/fxsjy/jieba

4.3 Results

According to official evaluation requirements, we use the macro-averaged precision, recall and F1 to evaluate the performance. They are defined as:

$$Micro\_avg = \frac{1}{N}\sum_{i=1}^{m} w_i \rho_i \tag{16}$$

Where m denotes the number of class, in the case of this dataset is 18. ρ_i denotes the accuracy, recall or F1 score of i-th category, w_i represents how many test examples reside in i-th category, N is total number of examples in the test set.

For ease of description, symbols in each model are specified as follows: "C" means mixed, "A" means attention mechanism, "W" means word-level, "Ch" means character-level, "Ens" means combination model. Thus, CA-BiLSTM is our attention-based multi-representation mixed model.

We first investigate the effect of pre-training, then test the performance of classifiers on single-granularity. After that, the effectiveness of attention mechanism is verified. Finally, we evaluate our combination model.

Table 4. The results of each model

Pre-trained or not	Model	Results		
		P	R	F1
No	N-BoW	0.7483	0.7448	0.7465
	CNN	0.7076	0.7021	0.7048
	CA-BiLSTM	0.7438	0.7430	0.7434
Yes	N-BoW	0.7911	0.7835	0.7873
	CNN	0.7692	0.7631	0.7661
	CA-BiLSTM	**0.8098**	**0.8093**	**0.8095**
	WA-BiLSTM	0.7704	0.7702	0.7703
	ChA-BiLSTM	0.7654	0.7650	0.7652
	C-BiLSTM	0.7885	0.7883	0.7884
	Ens-Only	0.8113	0.8108	0.8110
	CA-BiLSTM+Ens	**0.8180**	**0.8172**	**0.8176**

Results show that, each model has a high degree of reliance on pre-training, and the pre-trained look-up table should be large enough to cover all the words. If not, it may result in a certain loss of accuracy. No matter pre-trained or not, our CA-BiLSTM model is better than N-BoW and CNN. Word-level representation of headline is more related to the category decision. When we drop the attention mechanism of CA-BiLSTM, the results will decline. In addition, it can be seen that combination model can predict more accurate than single model, and targeted combination model (CA-BiLSTM+Ens) performs better than the combination model (Ens-Only) directly testing on the whole test set. F1 score of our targeted combination model eventually reaches 0.8176, which can rank No.3 among the participating teams.

4.4 Discussion

In summary, our CA-BiLSTM model performs better than others under the single-model paradigm. When we adopt combination model, the targeted processing mechanism can fully exploit the complementarity between the sub-models and get the best experimental results. The details of best experimental results are shown in Table 5.

Table 5. The details of best experimental results

Category	P	R	F1
history	0.8146	0.8240	0.8192
military	0.8330	0.8480	0.8404
baby	0.8452	0.8815	0.8630
world	0.7081	0.6915	0.6997
tech	0.7979	0.8330	0.8151
game	0.8860	0.8820	0.8840
society	0.5687	0.6270	0.5964
sports	0.9083	0.8950	0.9016
travel	0.7385	0.7980	0.7671
car	0.9061	0.8780	0.8918
food	0.8284	0.8670	0.8473
entertainment	0.7514	0.7615	0.7564
finance	0.8214	0.8120	0.8167
fashion	0.7998	0.8210	0.8103
discovery	0.9201	0.8460	0.8815
story	0.8502	0.7350	0.7884
regimen	0.8754	0.7590	0.8131
essay	0.8175	0.8175	0.8175
OVERALL	0.8180	0.8172	0.8176

As can be seen from Table 5, the accuracy of "society", "world" and "travel" are low. Analysis shows that the "world" and "travel" are highly correlated and prone to significant feature confusion such as country names or district names, resulting in weak identification and prediction. The lack of significant features brings about the lowest indicators to "society" among 18 categories. For "discovery", "car", "sports" and "game", classifier performs very well owing to their obvious and distinguished characteristics.

Further, we visualize the distribution of attention information, as shown in Figure 5. Only a small part of the headline is strongly related to its category, so our attention mechanism is necessary.

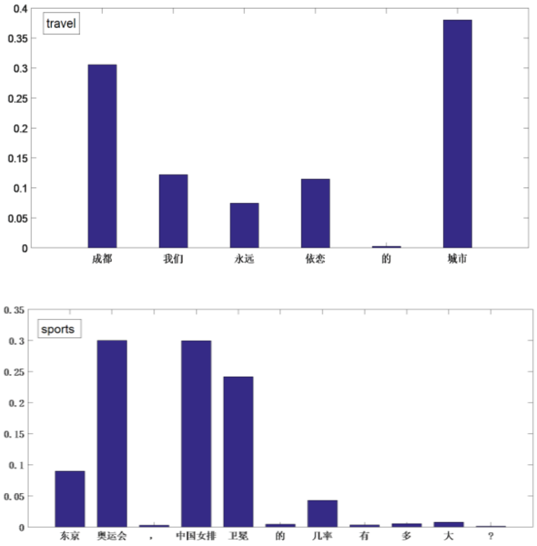

Fig. 5. Visualization of attention

What's more, we get a relatively poor result when removing the stopwords. It means that, there is no need to remove stopwords when modeling very short text because the stopwords contain certain syntax and semantic information. If we remove them, it may destroy the original syntactic structure, even damage the semantics representation.

5 Conclusion

This paper presents an effective approach for Chinese news headline classification based on multi-representation mixed model with attention and ensemble learning. We model and integrate the character-level feature into word-level feature of headlines via BiLSTM, alleviating the influence of word segmentation and strengthening the semantic representation. Meanwhile we adopt the attention mechanism to highlight the key characters or words related to the classification decision, with a preliminary test result. Finally, for samples with lower confidence in preliminary test result, we introduce ensemble learning to determine the final category of the whole test samples by sub-models voting. When testing on the NLPCC 2017 official test dataset, we obtain a competitive result. Next, we will integrate part of speech and named entities into our model, because we believe they stress certain potential impact on Chinese news headline classification.

References

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

2. Lai S, Xu L, Liu K, et al. Recurrent Convolutional Neural Networks for Text Classification[C]//AAAI. 2015, 333: 2267-2273.
3. Zhang X, Zhao J, LeCun Y. Character-level convolutional networks for text classification[C]//Advances in neural information processing systems. 2015: 649-657.
4. Zhou Y, Xu B, Xu J, et al. Compositional Recurrent Neural Networks for Chinese Short Text Classification[C]//Web Intelligence (WI), 2016 IEEE/WIC/ACM International Conference on. IEEE, 2016: 137-144.
5. Mikolov T, Karafiát M, Burget L, et al. Recurrent neural network based language model[C]. Interspeech. 2010, 2: 3.
6. Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078, 2014.
7. Nallapati R, Zhou B, Gulcehre C, et al. Abstractive text summarization using sequence-to-sequence rnns and beyond[J]. arXiv preprint arXiv:1602.06023, 2016.
8. Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult[J]. IEEE transactions on neural networks, 1994, 5(2): 157-166.
9. Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.
10. Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate[J]. arXiv preprint arXiv:1409.0473, 2014.
11. Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks[C]//Advances in neural information processing systems. 2014: 3104-3112.
12. Greff K, Srivastava R K, Koutník J, et al. LSTM: A search space odyssey[J]. IEEE transactions on neural networks and learning systems, 2016.
13. Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: A simple way to prevent neural networks from overfitting[J]. The Journal of Machine Learning Research, 2014, 15(1): 1929-1958.
14. Raskutti G, Wainwright M J, Yu B. Early stopping and non-parametric regression: an optimal data-dependent stopping rule[J]. Journal of Machine Learning Research, 2014, 15(1): 335-366.