An Efficient Character-Level and Word-Level Feature Fusion Method for Chinese Text Classification

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

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

Email: \textsuperscript{a} jwenzhen@163.com, \textsuperscript{b} zhltdzh@163.com, \textsuperscript{*} lyuanj@swu.edu.cn

Abstract. In order to extract semantic feature information between texts more efficiently and reduce the effect of text representation on classification results, we propose a features fusion model C_BiGRU_ATT based on deep learning. The core task of our model is to extract the context information and local information of the text using Convolutional Neural Network(CNN) and Attention-based Bidirectional Gated Recurrent Unit(BiGRU) at character-level and word-level. Our experimental results show that the classification accuracies of C_BiGRU_ATT reach 95.55\% and 95.60\% on two Chinese datasets THUCNews and WangYi respectively. Meanwhile, compared with the single model based on character-level and word-level for CNN, the classification accuracies of C_BiGRU_ATT is increased by 1.6\%, 2.7\% on the THUCNews, and is increased by 0.6\%, 5.2\% on the WangYi. The results show that the proposed model C_BiGRU_ATT can extract text features more effectively.

1. Introduction

The researches on text classification mainly focus on text representation and classification model. The good text representation method plays a decisive role in the performance of text classification. The Bag of Word(BoW) \cite{1,2} or the Vector Space Model(VSM) \cite{3} are widely used in text representation. However, these two models have the problems of lack of contextual information and easy causing dimensional disaster. Hinton proposed Distributed Representation \cite{4} to solve the sparse problem of text representation. More recently, it has become more common to use a deep neural network to extract sentence or higher level representation in texts. CNN has been shown to achieve impressive results on the sentence classification \cite{5}. On the other hand, based on the problems faced by traditional RNN \cite{6}, scholars proposed Long Short-Term Memory(LSTM) \cite{7}. The C-LSTM model proposed by Zhou et al. \cite{8} to get the sentence representation. Chung et al. proposed the GRU model \cite{9} in 2014, and through multiple experiments, it was shown that the GRU had similar performance with LSTM on multiple tasks, although it had few parameters. Zhang et al. proposed a character-level convolutional network (ConvNets) and verified the accuracy of classification using the Chinese news corpus \cite{10}. Yang et al. designed the Hierarchical Attention Network (HAN), and added the attention mechanism \cite{11} after the output of the bidirectional Recurrent Neural Network(RNN). However, there are some problems with the use of a single deep learning model. For example, although CNN can extract local features it lacks the ability to learn sequence association information. Additionally, RNN is appropriate for serializing information but cannot extract features in parallel. Considering the above, inspired by the good performance of CNN and RNN in text representation at the same time, we propose a features fusion representation model C_BiGRU_ATT based on CNN and Attention based BiGRU at the character-level and word-level. Our model uses character-level to avoid auxiliary work.
such as Chinese word segmentation, and helps the model extract information from the original texts. Moreover, we concatenate the features extracted by Convolutional Neural Network (CNN) and Attention-based Bidirectional Gated Recurrent Unit (BiGRU) at the character-level and word-level to enrich text semantics information. Experimental results show that the classification accuracies of our model reach 95.55%, 95.60% on two Chinese datasets Netease and THUCNews respectively. The overall model is shown in Figure 1.

**Figure 1. C_BiGRU_ATT structure overview.**

### 2. Model

**2.1. Character-level Text Representation on CNN**

CNN mainly includes convolutional layer, pooled layer and fully connected layer. This paper generates a feature representation for each character vector through multiple filters of different sizes in CNNs. Let $x_i$ be the $k$-dimensional character vector of the $i$-th character in the text, and a text of length $n$ can be expressed as the Equation 1:

$$x_{in} = x_1 \oplus x_2 \ldots \oplus x_n$$  \hspace{1cm} (1)

Where $\oplus$ represents splicing and $x_{i+j}$ represents spliced multiple character $x_i, x_{i+1}, \ldots, x_{i+j}$.

- **Convolution Layer**: The convolution kernel as $w \in R^{hk}$, its filter size is $h$. The feature $c_i$ is obtained by convolution operation of the text $x_{i+h-1}$ in the window, as shown in Equation (2):

$$c_i = f(w \cdot x_{i+h-1} + b)$$  \hspace{1cm} (2)

Where $w$ is the weight factor of the convolution kernel, $b$ is the bias, and $f$ is the nonlinear activation function. This convolution operation is applied in every window of the text matrix \{ $x_{1k}, x_{2k+1}, x_{3k+2}, \ldots, x_{n-k+1}$ \}. The final feature map is shown in Equation (3):

$$c = [c_1, c_2, \ldots, c_{n-h+1}]$$  \hspace{1cm} (3)
• The pooling layer reduces features and network parameters by reducing the information output from the convolutional layer. Its definition is given in Equation (4):

$$\bar{c} = \max(c)$$ (4)

2.2. Word-level Text Representation on BiGRU

In our model, inputting word sequence \( \{x_1, x_2, \ldots, x_n\} \) to BiGRU network iteratively. The calculation method of the GRU node is indicated in Equation (5):

\[
\begin{align*}
  z_t &= \sigma(W^z x_t + U^z h_{t-1}) \\
  r_t &= \sigma(W^r x_t + U^r h_{t-1}) \\
  \tilde{h}_t &= \tanh(W^h x_t + U^h (h_{t-1} \odot r_t)) \\
  h_t &= (1 - z_t) \odot \tilde{h}_t + z_t \odot h_{t-1}
\end{align*}
\] (5)

Where \( z_t \) is a set of update gates, \( r_t \) is a set of reset gates and \( \odot \) is an element-wise multiplication. \( W^z, W^r, W^h \) and \( U^z, U^r, U^h \) are weight matrices to be learned, and \( \tilde{h}_t \) is the candidate activation. The input sequences fed into a BiGRU network, and each BiGRU unit calculation is as shown in Equation (6),(7) and (8):

\[
\begin{align*}
  \overrightarrow{h}_t &= \text{GRU}(x_t), t \in [1, T] \\
  \overleftarrow{h}_t &= \text{GRU}(x_t), t \in [1, T] \\
  h_t &= [\overrightarrow{h}_t, \overleftarrow{h}_t]
\end{align*}
\] (6),(7),(8)

Where \( \text{GRU} \) is the forward RNN network, \( \overrightarrow{h}_t \) is the hidden layer output of the forward RNN at time \( t \), \( \text{GRU} \) is the backward RNN network, \( \overleftarrow{h}_t \) indicates the hidden layer output of the backward RNN at time \( t \). \( h_t \) is a vector obtained by concatenating the forward \( \overrightarrow{h}_t \) and the backward \( \overleftarrow{h}_t \).

2.3. Attention Mechanism

In the text, the contribution of each word to the text category is different. Hence, we introduce attention mechanism to extract such words that are imported to the meaning of the text. Its calculation formula is as shown in Equation (9),(10) and (11).

\[
\begin{align*}
  u_t &= \tanh(W_a h_t + b_a) \\
  \alpha_t &= \frac{\exp(u_t^T u_a)}{\sum_t \exp(u_t^T u_a)} \\
  v &= \sum_t \alpha_t \cdot h_t
\end{align*}
\] (9),(10),(11)

Where \( W_a \) is the weight coefficient in the feedforward neural network, \( b_a \) is the bias, \( h_t \) is the output of BiGRU, \( \alpha_t \) is the value of the softmax function, \( u_t \) is the weight coefficient. \( v \) represents an important information representation of text information.
2.4. C_BiGRU_ATT

Based on the methods mentioned in Sections 3.1 and 3.2, we propose a method of combining character-level features and word-level features for Chinese text classification. The model C_BiGRU_ATT in this paper consists of three parts: the embedded layer, the presentation layer and the classification layer. Character-level embedding vectors can avoid auxiliary work such as Chinese word segmentation, and help the model to extract information from the original text, thereby improving the classification effect. At the same time, the model in this paper enriches the semantic information of the text and solves the problem of the degradation of the performance of subsequent classification tasks due to the incorrect word segmentation in Chinese texts. For a text, we concatenate character-level features $\tilde{c}$ extracted by char-level CNN through multiple filters and word-level features $v$ are extracted by Attention-based BiGRU. At last, our model outputs the text vector $D$ are defined by the Equation (12):

$$D = \tilde{c} + v$$ (12)

The output of the presentation layer is the composite feature vector of the original text. We did not linearly transform the concatenated feature vector $D$, but extracted the feature vector $D$ and input it into the classification layer for text classification.

3. Experiment

We evaluate our model C_BiGRU_ATT on two Chinese datasets, WangYi and THUCNews. The two datasets are not the same as the number of categories and the balance of the texts. The WangYi dataset contains six categories: automotive, culture, economics, medicine, military, and sports, 4000 texts in each category, totaling 24,000 texts. THUCNews has lottery, real estate, home, education, social, fashion, political, sports, constellation, games, entertainment, totaling 11 categories and 20,480 articles. The number of texts in each category of THUCNews is as follows: 3728, 1628, 586, 936, 1925, 412, 2385, 1605, 955, 2006, 4260. The general situation of the data set is shown in Table 1.

| Dataset       | Number of categories | Avg-length | Max-length | Train/validation/test set |
|---------------|----------------------|------------|------------|---------------------------|
| THUCNews      | 11                   | 1032       | 26852      | 8:1:1                     |
| WangYi        | 6                    | 994        | 5217       | 8:1:1                     |

3.1. Experimental Settings

The experimental environments of this paper are Ubuntu 16.04, RAM 24GB, GPU1080TI, and TensorFlow framework. In the experiment, We first use the Jieba tool to segment the original datasets, then build word-level dictionary. For character-level, we also build the corresponding dictionary. Table 2 lists the parameters and corresponding parameter values in C_BiGRU_ATT.

| Parameter                  | Number of categories | Avg-length |
|----------------------------|----------------------|------------|
| Number of characters       | 1600                 | 800-1800   |
| Number of words            | 800                  | 400-900    |
| Char embedding dimension   | 128                  | 64,128,256 |
| Word embedding dimension   | 128                  | 64,128,256 |
| CNN filter height          | 3-5                  | 3-5        |
| CNN number of filters      | 100                  | 64,100,128,256 |
| BiGRU hidden layer size    | 100                  | 64,100,128,256 |
| Learning rate              | 0.1                  | 0.1        |
In the course of the experiment, we choose the highest accuracy to save the model on the validation set. Accuracy, loss and training time are chosen as the evaluation criteria for the experiment. The comparison experiment design of C_BiGRU_ATT and other methods is as follows.

- **char-CNN:** In this paper, the character vectors are input into CNN.
- **word-CNN:** In the comparison experiment, we also input the word-level vectors into CNN.
- **C_BiGRU_ATT:** When verifying the validity of the C_BiGRU_ATT, the parameters of C_BiGRU_ATT are consistent with the parameters of the single model CNN.

### 4. Results and Analysis

To verify the classification performance of C_BiGRU_ATT proposed in this paper, we compare the effects of C_BiGRU_ATT with character-level and word-level in CNN. Figure 2 and Figure 3 show the accuracy and loss of single model and C_BiGRU_ATT on the validation set, respectively. The results show that C_BiGRU_ATT is equally accurate on both datasets. In addition, character-level performance is better than word-level of single CNN model. By comparison, the C_BiGRU_ATT is better than the single model CNN in terms of accuracy and loss. The character-level is more suitable for use on the CNN model, and the combination of CNN and Attention-based BiGRU in the character-level and word-level has a nice representation of the texts.

**Figure 2.** Accuracy of each model on THUCNews and WangYi validation sets.

**Figure 3.** Loss of each model on THUCNews and WangYi validation sets.

In the two Chinese datasets, we compared the accuracy and training time under different training times of the validation set. As shown in Table 3, Table 4. In general, because the training parameters of word-CNN are less than that of char-CNN, so that the training time of word-CNN is much less than char-CNN. C_BiGRU_ATT takes more time than single models. C_BiGRU_ATT emphasizes the key features of the text by adding the attention mechanism to the BiGRU and using the attention to assign the corresponding weights. Compared with the simple maximum pooling on CNN, the time cost increases.

**Table 3.** Training time of each model on THUCNews validation set(s).

| Training times | 1k | 2k | 3k | 4k | 5k | 6k | 7k | 8k | 9k | 10k | 11k |
|----------------|----|----|----|----|----|----|----|----|----|-----|-----|
| Char-CNN       | 8.03 | 8.01 | 5.02 | 5.16 | 5.01 | 5.01 | 6.19 | 5.07 | 8.21 | 4.97 |
| Word-CNN       | 1.89 | 1.86 | 1.88 | 1.87 | 1.89 | 1.89 | 1.88 | 1.89 | 1.90 | 1.91 | 1.87 |
| C_BiGRU_ATT    | 6.89 | 6.86 | 6.93 | 6.88 | 6.91 | 6.94 | 6.90 | 6.89 | 6.82 | 6.95 | 6.81 |

**Table 4.** Training time of each model on WangYi validation set(s).

| Training times | 1k | 2k | 3k | 4k | 5k | 6k | 7k | 8k | 9k | 10k | 11k |
|----------------|----|----|----|----|----|----|----|----|----|-----|-----|
| Char-CNN       | 4.44 | 4.38 | 10.22 | 4.39 | 4.37 | 4.41 | 4.40 | 4.42 | 4.37 | 4.48 | 4.38 |
| Word-CNN       | 1.90 | 1.95 | 1.89 | 1.88 | 1.90 | 1.86 | 1.88 | 1.92 | 1.89 | 1.90 | 1.91 |
| C_BiGRU_ATT    | 6.92 | 6.83 | 6.92 | 6.98 | 6.88 | 6.94 | 6.95 | 6.85 | 6.91 | 7.03 | 6.91 |

On the test set of the two datasets, we also made a comparative analysis of the accuracy. As shown in Table 5, the accuracies of C_BiGRU_ATT on the THUCNews data set are 1.6%, 2.7% higher than that of char-CNN and word-CNN, respectively. The accuracies on the WangYi dataset are 0.6%, 5.2%
higher than that of char-CNN and word-CNN, respectively. The results demonstrate that the proposed model C_BiGRU_ATT is better for feature extraction of the texts.

| Method       | THUCNews | WangYi |
|--------------|----------|--------|
| Char-CNN     | 93.95    | 95.00  |
| Word-CNN     | 92.85    | 90.40  |
| C_BiGRU_ATT  | 95.55    | 95.60  |

### 5. Conclusion

In this paper, we propose a feature fusion model C_BiGRU_ATT that combines character-level and word-level representations for Chinese texts. In the experiments, we demonstrate that our model achieves competitive results by comparing with the single model built on character-level and word-level on CNN. Our experimental results show that the classification accuracies of C_BiGRU_ATT model reach 95.55% and 95.60% on two Chinese datasets THUCNews and WangYi respectively. Meanwhile, it also shows that our model C_BiGRU_ATT is able to obtain better text representations through learning character-level and word-level features. However, the model C_BiGRU_ATT in this paper increases the number of parameters because it adds both character and word levels, which increases the calculation amount, so the training time is higher than the single model. Future work will be further improved during the training time of C_BiGRU_ATT.

### References

[1] Wallach, H.M.: Topic modeling: beyond bag-of-words. In: Proceedings of the 23rd international conference on Machine learning. pp. 977–984. ACM (2006)

[2] Uijlings, J.R., Smeulders, A.W., Scha, R.J.: Real-time bag of words, approximately. In: Proceedings of the ACM international Conference on Image and Video Retrieval. p. 6. ACM (2009)

[3] Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Communications of the ACM 18(11), 613–620 (1975)

[4] Hinton, G.E., et al.: Learning distributed representations of concepts. In: Proceedings of the eighth annual conference of the cognitive science society. vol. 1, p. 12. Amherst, MA (1986)

[5] Kim, Y., Jernite, Y., Sontag, D., Rush, A.M.: Character-aware neural language models. In: AAAI, pp. 2741–2749 (2016)

[6] Bengio, Y., Simard, P., Frasconi, P.: Learning long-term dependencies with gradient descent is difficult. IEEE transactions on neural networks 5(2), 157–166 (1994)

[7] Shen, Y., Jin, R.: Learning personal+ social latent factor model for social recommendation. In: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 1303–1311. ACM (2012)

[8] Zhou, C., Sun, C., Liu, Z., Lau, F.: A c-lstm neural network for text classification. arXiv preprint arXiv:1511.08630 (2015)

[9] Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014)

[10] Er, M.J., Zhang, Y., Wang, N., Pratama, M.: Attention pooling-based convolutional neural network for sentence modelling. Information Sciences 373, 388–403 (2016)

[11] Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E.: Hierarchical attention networks for document classification. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 1480–1489 (2016)