Fusing Multi-Modal Char-Level Embeddings for Chinese Sentence Sentiment Analysis

Dong Liu (✉ liudong@lynu.edu.cn)

Luoyang Normal University

Caihuan Zhang

Luoyang Normal University

Yongxin Zhang

Luoyang Normal University

Youzhong Ma

Luoyang Normal University

Research Article

Keywords:

Posted Date: August 11th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-786024/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

ARTICLE TYPE

Fusing Multi-Modal Char-Level Embeddings for Chinese Sentence Sentiment Analysis

Dong Liu^{*1} | Caihuan Zhang² | Yongxin Zhang¹ | Youzhong Ma¹

¹Dept. of Information Technology, Luoyang Normal University, Luoyang Henan China.

²Dept. of Mathematics, Luoyang Normal University, Luoyang Henan, China.

Correspondence

*Dong Liu. No. 6, JiQing Road, YiBing District, Luoyang, Henan, China. Email: liudong@lynu.edu.cn

Funding Information

This research was funded by the Science and Technology Key Project of Henan province under Grant (No. 182102210434, No. 202102210357); Training plan for young backbone teachers of Colleges and universities in Henan (No.2017GGJS134); and the National Natural Science Foundation of China (No.61602231)

ABSTRACT

Chinese characters are one of the logographic writing systems. There is some association between semantics and structures, shape, phonetic information of Chinese characters. In this work, multi-modal Chinese character-level embeddings are extracted, including visual features, pre-trained embeddings, shapes, and phonetic information. These embedding sequences of Chinese sentences are first fed into individual Bi-LSTM networks to capture context features, and then fused into one vector for sentiment analysis. Experimental results validate that multi-modal character-level can contribute to Chinese sentence sentiment classification. And its effect on the result is analyzed by modal features ablation test.

KEYWORDS

multi-modal features, char-level embedding, Chinese characters, sentiment analysis.

1 | INTRODUCTION

Chinese is a logographic language. Many characters are inspired by what people have seen in daily life and created based on visual features. For example, a few Chinese characters and their corresponding image still retain a visual connection in the following Figure 1. Furthermore, children begin to read by learning figure literacy in Chinese enlightenment education, suggesting that the logograms of Chinese characters encode rich information of their meanings. Intuitively, NLP tasks for Chinese should benefit from the use of the visual information embodied in images of characters.

FIGURE 1 Chinese characters and their images

There have been some efforts focusing on the visual features of characters. A common and effective method is CNN-based algorithms that are applied to the visual features of character to extract features for downstream NLP tasks. For example, Dai and Cai[1] obtain Chinese character representations from char logos and reported that the incorporation of glyph representations achieved better performance on a segmentation benchmark task, but did not provide extra useful information for language modeling. Liu et al.[2] and Zhang, and LeCun [3] observed performance boosts on text classification tasks using character images. Su and Lee [4] found glyph embeddings help two tasks: word analogy and word similarity.

As mentioned earlier, Chinese characters are logographic, the original purpose of which was easy-to-draw, and slowly evolved into being easy-to-write. Their spatial-structure, except visual features, provide semantic, phonetic, and syntactic hints. Chinese characters consist of up to five radicals, which can have various relative positions, such as left-right, up-down, inside-out, and so on. These radicals may be divided into morphemes and phonemes,

indicating meaning and pronunciation. For instance, 雪(snow) 雹(hail) 雾(fog) 雷(thunder) are up-down arrangement structure, and all have a sub-structure 雨(rain), which commonly denote meteorological phenomena. Another example is 抱bào、袍páo、胞bāo、饱bǎo, which all share 包bāo leading to similar pronunciations. Recently, there have been some effects taking into account these features. Tan et al.[5] use the Wubi scheme--- a Chinese character encoding method that mimics the order of typing the sequence of radicals for a character on the computer keyboard---to improve performances on Chinese-English machine translation. Peng et al. [6] propose radical-based hierarchical embedding incorporating semantic and sentiment information. Cao et al. [7] gets down to fine-granular and propose stroke n-grams for character modeling.

But, it should be noted that using only one of these features is often not enough because of the complexity and special evolution process of Chinese characters. For image features, many characters' shapes have changed beyond recognition and lost the most significant amount of pictographic information after a long evolutionary, simplified process. Besides, although 90% of Chinese characters are semantic-phonetic compound characters, 39% of phonemes can represent their pronunciations. Many morphemes also can not indicate the meanings, of which some are even opposite in meaning, such as 远(far) and 近(near). As a result, we suspect that combining multi-modal features, such as image, structure, pronunciation, context, and so on, should help understand the meaning and sentiment of Chinese text.

In this paper, we extract multi-modal char-level embeddings from character images, pretraining on large-scale corpus, structure, and pronunciation, and feed into three Bi-LSTM networks to learn context features respectively. The features are fused to form a vector in sentence-level for sentiment classifier. In sum, this paper makes the following three-fold contributions:

- We extract multimodal char-level embedding sequences from Chinese sentences and learn three context feature vectors to fuse for sentiment classification. We validate the utility of the fused feature vector by experiment.
- To the best of our knowledge, we firstly fuse multimodal embedding including text, image, structure, and pronunciation for sentiment feature learning, and explore different effects that various modal features in classification by ablation test.
- The multimodal char-level embedding sequence is applied directly to the sentiment classifier model. It is different from previous works that use character embedding to improve word embedding performance in NLP downstream tasks.

2. | RELATED WORKS

Currently, Mainstream research of the NLP model uses pre-trained word embedding. The most popular word embedding model is word2Vec[8] and GloVe[9], which is built on the distributional hypothesis[10], that is, words with similar contexts share similar meanings. The models regard words as atomic tokens, which potentially result in ignorance of useful internal structure information of words. Besides, vocabulary is rapidly expanding because of huge words number. To improve the performance and robustness of word embedding, sub-word information has been employed. These methods mainly focus on alphabetic writing systems, however, have little effect when they are applied to logographic writing systems.

Chinese is a typical example of a logographic writing system. Research on Chinese word embedding becomes gradually active. Sun et al.[11] use a synonym thesaurus to eliminate ambiguity of Chinese words and characters. Chen et al.[12] propose a CWE model for jointly learning Chinese words and characters embedding through integrating character structure information. Sun et al.[13] and Li et al.[14] extract sub-word features using a radical dictionary. Shi et al. [15] also utilize radical information to improve Chinese word embeddings. Yu et al. [16] propose a JWE model which learn jointly word embeddings based on an extended radical set.

Inspired by the advancement of deep learning in the image processing field, many authors explore Chinese word embeddings from a visual perspective. Su and Lee[4] use a CNN autoencoder to extract features directly from Chinese character bitmaps. Liu et al. [2] consider character-level compositionality and produce visual character embeddings through CNN. Meng et al. [17] use the ensemble of the historical and the contemporary scripts and utilize the Tianzige-CNN (田字格) structures tailored to logographic character modeling for Chinese by pretraining method like BERT[18]. Su et al. [19] propose a word embedding model, called VCWE, which learns the intra-character composition via a CNN, learn the inter-character composition via Bi-LSTM, and the contextual information based on the skip-gram. Our method is similar to the ideas mentioned above. The major difference is that we utilize directly multimodal character-level embeddings to NLP downstream tasks, instead of aiming to improve word embedding performance.

3. | MODEL AND METHOD

3.1 | Architecture

Structurally, our method is formed by five layers: characters sequence input layer, feature extracting layer, Bi-LSTM network layer, fusion layer, classifier layer. A Chinese sentence is divided into characters. Three modal features are extracted in the feature extracting layer, that is image feature (convAE), pre-trained

embedding (BERT), and structure and pronunciation feather (pictophonetic). Then, three type features are feed into the Bi-LSTM network respectively to learn the context information. Finally, these features are used by the classifier after being fused in the fusion layer. The whole framework is illustrated in Figure 2.

FIGURE 2 The model framework

The following sections will introduce three feature extraction methods in detail. Next, using Bi-LSTM for the context features and feature fusion process is described.

3.2 | Pretrained BERT embeddings

BERT(Bidirectional Encoder Representation from Transformers) , is a pre-trained bidirectional language encoder model based on Transformer that was published by Google in 2018. It achieves state-of-the-art performance on 11 GLUE benchmark (General Language Understanding Evaluation benchmark) tasks. BERT has made innovations in two aspects: one is feature extraction based on bidirectional transformer; the other is using a two-stage model of pre-training + fine-tuning. In this paper, we use Chinese BERT-Base

(https://storage.googleapis.com/bert_models/2018_11_03/chinese_L-12_H-768_A-12.zip. The model supports Chinese Simplified and Traditional, Basic parameters include 12-layer, 768-hidden, 12-heads, 110M parameters. Because it is just used as a feature extractor, we use an off-the-shelf Python package, bert-embedding(<https://github.com/imgarylai/bert-embedding>), to acquire token embedding.

3.3 | Image Embeddings based on deep CNN

As Liu et al. [2] pointed out, visually similar words share similar embedding. In this paper, a depth convolution self-encoder (convAE, Masci et al. 2011[20]) is used to extract Chinese character image features. The structure of the convAE model is shown in Figure 3. Firstly, for each Chinese character, the corresponding image (40, 40) is generated as input through noise reduction and binarization. There are five convolution layers in convAE, and the parameters of each

layer are consistent with those of Su et al. [4]. The difference is only the size of the image and the feature dimensions. To keep consistent with the embedding dimension in BERT, the dimensions of the feature vector of the image is 768.

FIGURE 3 The structure of convAE model

Each layer is followed by a ReLU nonlinear activation function layer, and finally, the bitmap is reconstructed. Adam optimizer are used and set the batch size to 128 , epoch to 30. We select 7438 Chinese characters and extract their image features.

The auto-encoder uses the MSE function as a loss function,

$$E(\theta) = \frac{1}{2n} \sum_{i=1}^n (x_i - y_i)^2$$

3.4 | Structural and phonetic embeddings

In terms of structural features, we use one of the commonly used indexing systems for Chinese characters, the four-corner index system, which classifies the four corners of complex Chinese characters into 10 types according to the shape of stroke, and marks them with 10 numbers 0 ~ 9. A Chinese character can be encoded by up to 5 numbers. It is a method to calculate the structural similarity of two Chinese characters by converting them into four-corner codes. See the following coding example:

From the Figure 4, 招提挥 have similar codes because they are similar in structure and stroke. The other three characters 角查, 过 are different in structure and stokes of their corners, so the codes are very different. However, the codes and Chinese characters are not one-to-one. Multiple Chinese characters may correspond to one number, such as the following four characters:

57062	56081	57054	27227	40106	34300
招	提	挥	角	查	过

FIGURE 4 The 4-corner code examples

The 4-corner codes of these four characters in Figure 5 are 80000, and it is difficult to distinguish them from the images of Chinese characters. However, we can see that their pronunciation (Pinyin, modern Chinese pronunciation system) is relatively great different. Therefore, we have further considered the Pinyin of Chinese characters.

FIGURE 5 The characters with the same code 80000

In terms of pronunciation features, there is a phenomenon that characters that sound alike will usually have a similar meaning. For instance, 蒙méng 灭miè 密mì 盲māng all are associated with dark. 广guǎng 朗lǎng 昌chāng 旺wàng 长zhǎng all have the meaning of high, bright, large and so on. Although this is not an inevitable connection, but because there exist a large number of semantic-phonetic compound characters in Chinese, the extracted features will be more comprehensive taking into account the pronunciation features of characters.

Another off-the-shelf tool package is used to extract the structural and phonetic features (https://github.com/howl-anderson/hanzi_char_featurizer). Figure 6 below is from the toolkit with some modifications. These two parts of features are represented as a 106-dimensional vector, in which the first 45 dimensions represent the pronunciation features, including initials, finals, and tones, while the rest 61 dimensions are the features of four corner coding.

FIGURE 6 The two parts of structural and phonetic features

3.5 | Sentiment classifier based on Bi-LSTM

As mentioned earlier, one innovation of BERT is the two-stage model of pre-training + fine-tuning. Since the fine-tuning stage of BERT still needs more resources and time consumption, we only use its pretraining model to obtain the embeddings of Chinese characters.

The features of Chinese characters in the context are obtained by a classification model based on bidirectional LSTM.

Long short term memory network (LSTM) is a special type of RNN. It was proposed by Hochreiter & Schmiduber in 1997 [21]. LSTM has been widely used in many NLP tasks and has achieved success. Due to its outstanding performance in solving the problem of long-term dependency, we chose a bidirectional LSTM. Figure 7 shows the structure of our model:

FIGURE 7 The classifier based on Bi-LSTM and feature fusion

Three Bi-LSTMs with the same structure receive three types of feature vectors corresponding to w_t as input: pretraining embedding, image embedding, and pictophonetic embedding. We use x_t to denote one of the t -th character embeddings, h_{t-1} is the previous hidden state. h_{t+1} is the next hidden state. The character w_t is encoded using corresponding Bi-LSTM:

$$\begin{aligned} h_t^F &= LSTM(h_{t-1}^F, x_t), \\ h_t^B &= LSTM(h_{t+1}^B, x_t), \\ h_t &= [h_t^F; h_t^B] \end{aligned}$$

The outputs of the three Bi-LSTM are fused in Merge_X layer. There are three infusion methods: concatenate, add, average; A Dropout layer is followed and the dropout rate is set to 0.5. The last is a Dense layer for prediction, and softmax function is used to calculate the classification probability:

$$P(y = j|X) = \frac{\exp(w_j^T X + b_j)}{\sum_{i=0}^1 \exp(w_i^T X + b_i)}$$

The Adam optimizer is selected in training. The loss function is binary cross-entropy between predicted and true categories:

$$J = -\frac{1}{B} \sum_{i=1}^B \sum_{j=0}^1 p'_{i,j} \log p_{i,j} + (1 - p_{i,j}) \log(1 - p'_{i,j})$$

where B is batch size, $p'_{i,j}, p_{i,j} \in \{0, 1\}$ denote the predicted and true categories of the i -th sample respectively.

4. | Ablation test

We apply the model to a Chinese e-commerce review dataset, which covers six domains and contains more than 20000 reviews. It is mainly about the service reviews of commodities and hotels. The shortest comment is only one word, such as "好(good)" and "赞(like)". The longest comment is 2968 characters, which is about complaints about tourism and hotel services. All comments were marked with positive or negative. After scrambling the order, the first 15000 were selected as the training set and the rest as the test set.

In the input layer, the maximum length of each comment sequence is limited to 50 Chinese characters. If there are less than 50, the last word is used to pad out. Because the BERT embedding is used, which dimension is 768, the same dimension is used for the image vector, and 106 dimension vector is used for the pictophonetic vector. There are 256 hidden states in LSTM, dropout is set to 0.5; 2 classification and "sigmoid" activation function are used in the full connection layer.

Accuracy is selected as the evaluation index, TP represents true positive, TN is true negative, FP is false positive and FN denote false negative:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\%$$

We conduct ablation experiments to verify the influence of the various mode feature vectors on the results. Table 1 shows the influences on classification accuracy and tests different fusion methods.

TABLE 1 The ablation test result

Features	None	Add	Concatenate	Average
BERT Embedding	91.80			
BERT Embedding+Image		92.89	93.21	92.66
BERT Embedding+Image +Pictophonetic		92.80	93.98	92.76

It can be seen from the table that the accuracy is improved after image features are fused with BERT embeddings (without fine-tuning), especially the accuracy is increased by 1.41 percentage points by using "concatenate" fusion mode. When the pictophonetic embedding is added further, the accuracy rate is still higher than that of using only BERT embedding in all three fusion modes, of which the accuracy rate decreased by 0.09 percentage points only in the "add" mode, however, in the other two modes, it is better than that of only BERT embedding and image embedding fusion. Furthermore, the "concatenate" mode still achieves the highest accuracy rate, which is 0.77 percentage points higher than that of the former two vector fusion cases. In comparison with the case of only using the Bert word vector, there is a 2.18 percent increase.

5. | Conclusion

Based on the analysis of the experimental results, three conclusions can be drawn as follows: Firstly, Chinese characters, as a kind of language symbols with phonetic, semantic, and shape, can be directly processed as the input sequence of NLP tasks without word segmentation. Second, sentiment classification of Chinese sentences using multimodal features is more effective than that of a single type feature. The last thing is that different fusion methods have a certain impact on the classification results, and the "concatenate" method gets better classification accuracy in our work.

Inspired by the experimental results, we will continue to study the Chinese language processing tasks based on multimodal embedding. In the future, we plan to extend the work in the following directions: one is to train and extract Chinese character embedding with multimodal features on large-scale corpus, like BERT embedding, then apply them to various NLP downstream tasks; the other is that we would like to explore new and more effective fusion methods since our experiment shows that the fusion mode can affect the accuracy rate.

ACKNOWLEDGMENTS

APPENDIX

REFERENCES

1. Dai, F.; Cai, Z. Glyph-aware Embedding of Chinese Characters. In Proceedings of the Proceedings of the First Workshop on Subword and Character Level Models in NLP; Association for Computational Linguistics: Copenhagen, Denmark, 2017; pp. 64–69.
2. Liu, F.; Lu, H.; Lo, C.; Neubig, G. Learning Character-level Compositionality with Visual Features. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers); Association for Computational Linguistics: Vancouver, Canada, 2017; pp. 2059–2068.
3. Zhang, X.; LeCun, Y. Which Encoding is the Best for Text Classification in Chinese, English, Japanese and Korean? ArXiv170802657 Cs 2017.
4. Su, T.-R.; Lee, H.-Y. Learning Chinese Word Representations From Glyphs Of Characters. In Proceedings of the arXiv:1708.04755 [cs]; Association for Computational Linguistics: Copenhagen, Denmark, 2017; pp. 264–273.
5. Nikolov, N.I.; Hu, Y.; Tan, M.X.; Hahnloser, R.H.R. Character-level Chinese-English Translation through ASCII Encoding. ArXiv180503330 Cs 2018.
6. Peng, H.; Cambria, E.; Zou, X. Radical-Based Hierarchical Embeddings for Chinese Sentiment Analysis at Sentence Level. Knowl.-Based Syst. 2018, 148, 167–176.

7. Cao, S.; Lu, W.; Zhou, J.; Li, X. cw2vec: Learning Chinese Word Embeddings with Stroke n-gram Information. In Proceedings of the AAAI; 2018.
8. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed representations of words and phrases and their compositionality. In Proceedings of the Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2; Curran Associates Inc.: Lake Tahoe, Nevada, 2013; pp. 3111–3119.
9. Pennington, J.; Socher, R.; Manning, C. Glove: Global Vectors for Word Representation. In Proceedings of the Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP); Association for Computational Linguistics: Doha, Qatar, 2014; pp. 1532–1543.
10. Harris, Z.S. Distributional Structure. *WORD* 1954, 10, 146–162, doi:10.1080/00437956.1954.11659520.
11. Sun, Y.; Lin, L.; Tang, D.; Yang, N.; Ji, Z.; Wang, X. Modeling Mention, Context and Entity with Neural Networks for Entity Disambiguation.; Buenos Aires, Argentina, 2015; pp. 1333–1339.
12. Chen, X.; Xu, L.; Liu, Z.; Sun, M.; Luan, H. Joint learning of character and word embeddings. In Proceedings of the Proceedings of the 24th International Conference on Artificial Intelligence; AAAI Press: Buenos Aires, Argentina, 2015; pp. 1236–1242.
13. Sun, Y.; Lin, L.; Yang, N.; Ji, Z.; Wang, X. Radical-Enhanced Chinese Character Embedding. In Neural Information Processing; Loo, C.K., Yap, K.S., Wong, K.W., Teoh, A., Huang, K., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, 2014; Vol. 8835, pp. 279–286 ISBN 978-3-319-12639-5.
14. Li, Y.; Li, W.; Sun, F.; Li, S. Component-Enhanced Chinese Character Embeddings. In Proceedings of the Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics: Lisbon, Portugal, 2015; pp. 829–834.
15. Shi, X.; Zhai, J.; Yang, X.; Xie, Z.; Liu, C. Radical Embedding: Delving Deeper to Chinese Radicals. In Proceedings of the Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers); Association for Computational Linguistics: Beijing, China, 2015; pp. 594–598.
16. Yu, J.; Jian, X.; Xin, H.; Song, Y. Joint Embeddings of Chinese Words, Characters, and Fine-grained Subcharacter Components. In Proceedings of the Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics: Copenhagen, Denmark, 2017; pp. 286–291.
17. Meng, Y.; Wu, W.; Wang, F.; Li, X.; Nie, P.; Yin, F.; Li, M.; Han, Q.; Sun, X.; Li, J. Glyce: Glyph-vectors for Chinese Character Representations. ArXiv190110125 Cs 2019.
18. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the NAACL-HLT; 2019.
19. Sun, C.; Qiu, X.; Huang, X. VCWE: Visual Character-Enhanced Word Embeddings. In Proceedings of the NAACL-HLT; 2019.
20. Masci, J.; Meier, U.; Cireşan, D.; Schmidhuber, J. Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction. In Proceedings of the Proceedings of the 21st International Conference on Artificial Neural Networks - Volume Part I; Springer-Verlag: Berlin, Heidelberg, 2011; pp. 52–59.
21. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* 1997, 9, 1735–1780.