A Neural Network-based Sentiment Analysis Scheme for Tang Poetry

Yongrui Tang
Beijing Key Laboratory
of Intelligent Telecom.
Software and Multimedia
Beijing Univ. of
Posts and Telecom.
Beijing 100876, China
Email: yongrui_tang@bupt.edu.cn

Xumei Wang
Wuhan Huaxia
University of Technology
Wuhan 430000,china
E-mail:364787159@qq.com

Peng Qi
Beijing Key Laboratory
of Intelligent Telecom.
Software and Multimedia
Beijing Univ. of
Posts and Telecom.
Beijing 100876, China
Email: 13591036298@163.com

Yan Sun
Beijing Key Laboratory
of Intelligent Telecom.
Software and Multimedia
Beijing Univ. of
Posts and Telecom.
Beijing 100876, China
Email: sunyan@bupt.edu.cn

***Abstract*—Poetry is a very popular literary form. Currently, its sentiment analysis is one of the hottest research trends. However, there are few relevant studies focusing on the sentiment analysis of ancient Chinese poetry, especially for Tang Poetry. In this paper, we propose a deep learning–based method to solve the above problem. Specifically, we combine Convolutional Neural Network and Gate Recurrent Unit to better extract the characteristics of Tang poetry. In addition, considering the special structural characteristics of Tang poetry, a multi-channel processing model is used to reshape the feature vector of sentences. Finally, in order to verify the rationality and superiority of the proposed methods, we construct a dataset by labeling more than 2500 representative Tang poems. The experimental results prove that our scheme has a higher accuracy rate, up to 64%, compared against three other competing methods.**

***Keywords*—neural network; sentiment analysis; Tang poetry**

I. INTRODUCTION

Recently, poetry has attracted significant attention in the field of machine learning and natural language processing (NLP) [1]. In particular, its sentiment analysis is one of the main research direction. Poetry formed in different languages and different eras show different characteristics. Therefore, the sentiment analysis of poetry does not make as much progress as other texts (e.g. reviews and news). To make matters worse, there are few studies focusing on the sentiment analysis of ancient Chinese poetry, especially for the representative Tang Poetry.

At present, there have been some studies on sentiment analysis of poetry. These studies are mainly concentrated in the field of non-Chinese poetry, such as Arabic poetry[2], English poetry[3] and Malay Forklores[4]. As for Chinese poetry, the use of deep learning to generate ancient poetry is a hot topic. Unfortunately, in the aspect of sentiment analysis, there are only a few studies on the analysis of sentimental tendencies in ancient poetry vocabulary. On the other hand, since Tang poetry pays more attention to the rhythm than non-Chinese poetry, the existing studies cannot be effectively applied to Tang poetry.

In this paper, we propose a novel sentiment analysis scheme based on Convolutional Neural Network (CNN) and Gate Recurrent Unit (GRU). Specifically, we first divide their emotions into two categories: positive and negative. Then the whole Tang poetry are trained to obtain their word embedding matrix (surface features). In addition, a two-channel processing model is designed to reshape these surface features. Next, we introduce the CNN network to further extract the abstract features. Finally, these abstract features are fed to GRU network to obtain emotional analysis results.

In order to verify the effectiveness and superiority of the proposed methods, we construct a dataset by labeling more than 2500 representative Tang poems as positive and negative. Furthermore, we compare it against decision tree (DT), long short-term memory network (LSTM) and Bidirectional Gate Recurrent Unit (Bi-GRU) network. The experimental result shows that the accuracy rate of our proposed model is the highest and reaches 64%,

II. RELATED WORK

A. Others Work

In the field of non-Chinese poetry research, there have been some achievements. In 2015, Justine T. Kao and Dan Jurafsky analyze the characteristics of English poems in the 19th century through comparison and research (Justine T. Kao and Dan Jurafsky) [5]. However, their study tends to be literary analysis and the method is not universal. In 2016, Shinji Kikuchi *et al*. propose convolutional neural networks for quality estimation of Haiku poems in Japanese (Shinji kikuchi *et al.*) [6]. Unfortunately, their approach does not focus the problem on sentiment analysis. In the same year, P.S. Sreeja *et al*. compare the corpus-based emotion recognition method of poetry with the vector space model method (VSM) for the sentiment analysis of English poetry. Their paper shows that the traditional VSM achieves better results (P.S. Sreeja and G.S. Mahalakshmi) [3]. We can see that machine learning methods begin to be used in the sentiment analysis of poetry. In 2018, Mastura Md Saad *et al*. compare the effect of support vector machine (SVM) and decision tree algorithm (DT) on

978-1-7281-3129-0/20/$31.00 ©2020 IEEE

the recognition of Malay Forklores (Mastura Md Saad *et al.*) [4]. It turns out that DT performs much better than SVM. In 2019, Munef Abdullah Ahmed *et al.* use Naive Bayes (NB), SVM, Linear Support Vector classification (SVC) methods to perform sentiment analysis on modern Arabic poetry. The recognition accuracy rate of their experiment result reaches 95% (Abdullah Ahmed *et al.*) [2]. Although deep learning methods have been used in many aspects of text sentiment analysis, they are still lacking in sentiment analysis of poetry.

In the study of Chinese ancient poetry, most of the deep learning methods are applied to poetry generation. In 2016, Zhe Wang use the encoder-decoder framework based on RNN to generate ancient Chinese poems (Zhe Wang *et al.*) [7]. Similarly, Xiaoyuan Yi also use this framework to generate ancient Chinese poems in 2017 with better results (Xiaoyuan Yi *et al.*) [8]. However, in terms of sentiment analysis of ancient Chinese poetry, only Yufang Hou propose in 2015 to classify the “YiXiang” (objects that partly reflect the poet ’s subjective emotions) word of the ancient Chinese poetry. They use the Weighted Personalized PageRank (WPPR) method to analyze common “YiXiang” words in Tang poetry (Yufang Hou *et al.*) [9]. However, their work is limited to sentiment analysis of words.

In view of the lack of research on the sentiment analysis of Tang poetry, this paper proposes a deep learning method to fill the gap.

B. Special Structural characteristic of Tang poetry

There are a lot of situations where the sentences are “parallel” (对仗) in Tang poetry. The specific content of “parallel” is firstly that the voiced tones (平仄) of the upper and lower sentences must be opposite. Secondly, the relative sentence should have the same sentence structure and the same syntactic structure, such as subject-predicate structure corresponds to the subject-predicate structure. Sometime, the structure of some opponents is not necessarily the same, but it is required to be relative. Thirdly, the part-of-speech of a word is required to be consistent, such as noun-to-noun, verb-to-verb, adjective-to-adjective, etc. Furthermore, the ”lexical meaning” of a word must also be the same. Like nouns, they must have the same meaning range, such as astronomy, geography, palace, clothing, utensils, animals, plants, human bodies, behaviors, actions, and other words within the same meaning range can be right. For example, the poem of the Tang Dynasty poet Du Fu, ”A pair of orioles sing amid the willows green, and a flight of white egrets are flying up toward the sky” (“两只黄鹂鸣翠柳，一行白鹭上青天”) is strictly “parallel” .

In image processing, the RGB color image is divided into three channels of Red, Green and Blue during convolution. The convolution kernel convolves the images of the three channels at the same time to extract the feature information of different channels at the same pixel. It can make the extracted features more obvious. Therefore, in view of the special structural characteristics of Tang poetry, we propose a multi-channel processing model for better extracting features.

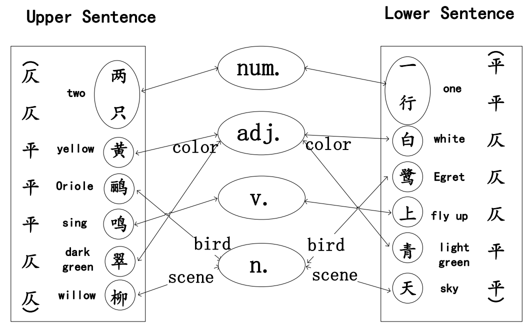

Fig. 1: special structural characteristics of Tang poetry

III. SYSTEM OVERVIEW

Our system is shown in Fig2.

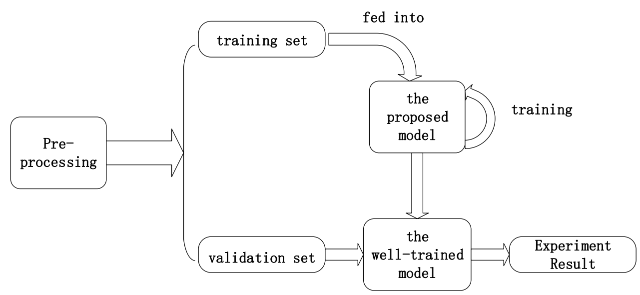

Fig. 2: system overview

In our system, the data set of Tang Poem is pre-processed first. Then the data set is used as training data of the word embedding model in order to obtain the word embedding matrix of the Tang Poem. Subsequently, the word vector matrix is sent to our model for training. After getting the trained model, we send the validation set to our model for prediction to get the prediction result and model evaluation parameters.

IV. THE PROPOSED MODEL

Our model consists of a word embedding layer, a reshape layer, a convolution pooling layer, and a GRU layer. The model is shown in Fig 3.

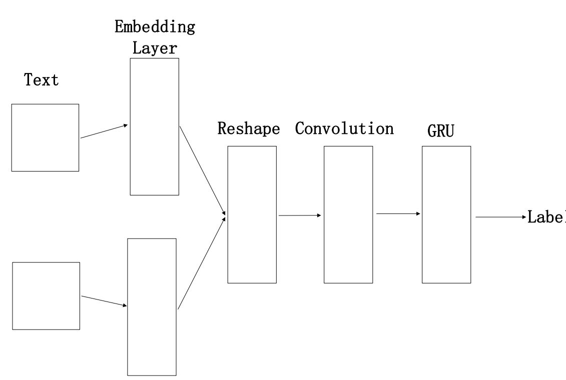

Fig. 3: The proposed model

In our model, the pre-processed verse is input to the word embedding layer, and the output of the word embedding layer is the word vector matrix of the verse. At this time, the vector matrix of the verse is sent to the reshape layer. The reshape layer recombines the vector matrices according to the “parallel” relationship between the upper and lower sentences of the Tang poems and outputs the reshaped verse vector matrix. Then, we pass these sorted matrices through the convolution layer and the pooling layer in order to extract the deep features of the verse. Finally, these features are input into the GRU network, and then the final classification result is obtained through the Softmax layer.

Algorithm 1 Model forward propagation

Input: the set of upper sentences(SUS) , the set of “parallel” sentences(SPS)
Output: the predict label set(PLS)
for each upper sentences(UST) in SUS and parallel sentences(PST) in SPS **do**
  generate $USTM$ from UST by $EmbeddingLayer$;
  generate $PSTM$ from SUS by $EmbeddingLayer$;
  combine $USTM$ and $PSTM$ as input sentences matrix(ISM) by $ReshapeLayer$;
end for
for each ISM **do**
  while Convolutional filter(CF) does not reach the end of ISM **do**
    calculate the feature matrix(FM) by convolution operation of CF and ISM;
    CF move one step;
  end while
  change shape of FM by sum operation of the two channel
  maxpool FM
  generate output feature sequence(OFS) by inputting FM into $GRULayer$
  generate decision vector(DV) by inputting OFS into $SoftmaxLayer$
  calculate the predict label(PL) from DV
  append PL to PLS
end for
return PLS;

A. Word Embedding Layer

Word Embedding is a general term for language models and representation learning techniques in NLP. It is to “embed” a word represented by one-hot into a low-dimensional space. In simple terms, it is the product of the embedding matrix E and the one-hot of the word. The mathematical relationship can be described using Equation(1):

$$E \bullet o_i = e_i \tag{1}$$

Word embedding can transform high-dimensional one-hot word vectors into low-dimensional word vectors. The similarity between words can be characterized by the distance of these vectors. The continuous bag-of-words model was first proposed by Tomas Mikolov in 2013 [10] [11]. It is a common method of word embedding. CBOW model has only three layers, namely the input layer, the hidden layer and the output layer.

When training the CBOW model, we put the ancient poems (“白日依山尽”) into the CBOW model. The input layer is one-hot vector of each word. After the shared hidden layer, the output layer is obtained. The output layer is the output probability of all words. The goal is to maximize the probability of the word: “依”. By comparing with the one-hot encoding of the “依”, the weights in the WI Matrix and WO Matrix are updated by the back-propagation algorithm. If the probability distribution has reached the set number of iterations, then the WI Matrix is the matrix we want. That is to say, one-hot representation of any word multiplied by this matrix will get its own word embedding.

After training the word embedding matrix WI Matrix, we multiply the one-hot vector matrix O and the word vector matrix E of the verse to obtain the word vector matrix of the verse:

$$E \bullet O = V \tag{2}$$

The matrix V is the low-dimensional word vector matrix of the verse. Because the CBOW model learns the context information of each word during the training process, the trained word embedding matrix contains the surface features of the verse.

B. Reshape Layer

According to the “parallel” characteristics of Tang poetry, the main work of this layer is to reshape the word vector matrix obtained from the word embedding layer.

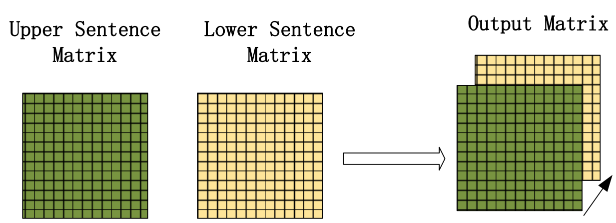

Fig. 4: Reshape Layer

As shown in Fig. 4, the upper and lower sentences of the poem pass through the word embedding layer respectively to obtain two different matrices. The two matrices are then stitched into multiple layers so that they can be convolved at the same time as they pass through the convolutional layers.

C. Convolutional Layer

CNN is a type of Feedforward Neural Networks with deep structure that includes convolution calculations. It is one of the representative algorithms of deep learning [12] [13]. Convolutional neural networks have the capability of representation learning.

In our model, the main function of the convolution layer is to extract the further features of the verse through the operation of convolution. The input matrix of a convolutional neural network has two layers. In each layer, the columns represent the word vectors of a word in the verse, and the rows represent the dimensions of the feature.

Therefore, the input word vector matrix is a $d \times n \times 2$ three-dimensional matrix, where d is the dimension of the pre-trained word vector and n is the number of words in the poem.

The convolution kernel will be convolved with the input matrix. As shown in Fig.5, the convolution matrix slides on the input matrix to complete the convolution operation. After the input matrix and the convolution kernel are convolved, the layer output a feature matrix of the same size as the input matrix. Finally, the three-dimensional matrix is compressed into a two-dimensional matrix by adding the features on the two channels.

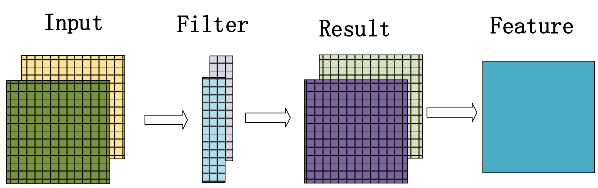

Fig. 5: Convolutional Layer

We use (C_{in}, H, W) to represent the input of the convolutional layer, and $(C_{out}, H_{out}, W_{out})$ to represent the output. C represents the number of channels. The values of C_{in} and C_{out} are both 2. H and W represent the height and width of the matrix. We analyze one of the channels. Let $out(N_i, C_{out})$ refer to the result of the output channel. then

$$\begin{aligned} &out(N_i, C_{out}) = bias(C_{out}) + \\ &\sum_{k=0}^{C_{in}-1} weight(C_{out}, k) * input(N_i, k) \end{aligned} \tag{3}$$

* represents a convolution operation of an input channel and a convolution kernel.

In order to make the extracted features more significant, a pooling operation is performed on the convolved matrix. After the matrix pooling, a new feature matrix (sequences) is obtained. These feature sequences are used as the input sequence of the Gate Recurrent Unit network.

D. GRU Layer

Recurrent neural networks (RNN) have short-term memory. If the sequence is long enough, it will be difficult for it to transfer information from earlier time steps to later steps. Therefore, RNN may miss important information from the beginning when processing text. During backpropagation, RNN encounter the problem of gradients (used to update the values of neural network weights) disappearing. The disappearance of the gradient problem means that when the gradient is reduced to very small when the gradient propagates over time, it will not continue learning. GRU, as a variant of LSTM, is also used to solve the gradient descent problem of RNN networks. The GRU network can solve the problem of the disappearance of the gradient of the traditional RNN network by forgetting some information by gate. At the same time, the account of training parameters and the training time of GRU are both smaller than those of LSTM.

There are two gates in GRU:

z: (update gate) taking sigmoid operation, used to indicate whether the previous information needs to be updated.

r: (reset gate), taking sigmoid operation, similar to LSTM's forget gate, which represents whether the previous information needs to be reset.

The network's forward propagation formula is:

$$\begin{aligned} r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\ z_t &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\ \widetilde{h}_t &= tanh(W_{\widetilde{h}} \cdot [r_t * h_{t-1}, x_t]) \\ h_t &= (1 - z_t) * h_{t-1} + z_t \cdot \widetilde{h}_t \end{aligned} \tag{4}$$

Among them, z_t is an update gate, which is a logic gate when updating activation. r_t is a reset gate, which decides whether to abandon the previous activation h_t when candidate activation. $\widetilde{h}_t$ is candidate activation. h_t is activation. W represents the parameter to be calculated.

Our GRU layer is a two-layer GRU structure.

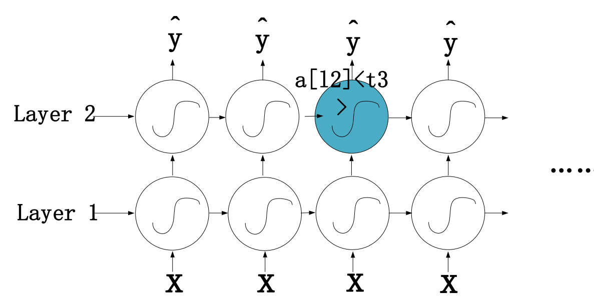

Fig. 6: 2-layer GRU Layer

As shown in Fig.6, the first layer sends the output sequence to the second layer. The output of the second layer in the GRU network is sent to the softmax function for final label determination.

V. EXPERIMENT

A. Dataset

All of our poem data is crawled from the web. Using Scrapy and Spider web crawling techniques, we have crawled 7993 poems from the Chinese ancient poetry website (www.gushiwen.org). The crawled content includes the poem title, author, and the full text of the poem. Then we select some poems from it for emotional labeling.

We divide the emotional tendency of the poems into Positive and Negative. When labeling the poems, we refer to the appreciation of ancient poems on the website(www.gushiwen.org). For example, we classify the boudoir(闺怨) poetry as negative, and the patriotic(爱国) poetry as positive. Finally, we labeled 2654 poems.

Some labeled poems are shown in TABLE 1.

TABLE I: Part of labeled poems

poems	label
万里路长在, 六年身始归. 所经多旧馆, 大半主人非.	-1
酒后高歌且放狂, 门前闲事莫思量. 犹嫌小户长先醒, 不得多时住醉乡.	1
莫言鲁国书生儒, 莫把杭州刺史欺. 醉客请君开眼望, 绿杨风下有红旗.	1
遥羡青云里, 祥鸾正引雏. 自怜沧海伴, 老蚌不生珠.	-1

B. Baseline

We selected Decision Tree algorithm from Keras library, Deep Learning method LSTM (Rafal Scherer, 2017), Bi-GRU (Rafal Scherer, 2017) as our baseline. Parameter settings for all baselines are as default. At the same time, for rigor, we also compare with the CNN-GRU model with the Reshape layer removed.

C. Parameter Settings

In our model, the output dimension of CBOW model is 128 and the number of iterations is 12. Because the input of the convolution layer has two channels, there are two convolution kernel, which take size 3×256. To ensure that the shape of the convolution is consistent with the input matrix, we use Same padding method, which means each value on the input matrix can be convolved. The convolution step is 1 and the pooling window size is 2. The GRU of each layer has a Dropout value of 0.2 and a Recurrent Dropout value of 0.1.

D. Result and Analysis

We selected two sizes of verification set (265 poems and 529 poems) for verification. The results are shown in fig7.

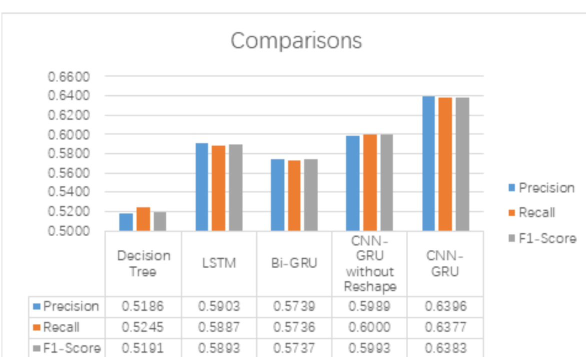

Fig. 7: Comparing of methods(265 poems)

The experimental results show that the precision, Recall and F1-Score values of the CNN-GRU model without Reshape layer are significantly improved compared to the Decision Tree method, about 8%. In addition, compared to deep learning methods: LSTM and Bi-GRU, there is also a 1% to 2% improvement. It turns out that the CNN-GRU without Reshape layer model performs better than baseline while we use the CBOW and CNN networks to extract the textual features of ancient poems in advance. At the same time, we see that the proposed model improves the parameters by about 4% compared to the CNN-GRU model without the Reshape layer, which indicates that the effect is significantly improved because of the Reshape layer. It also proves the correctness of considering the "parallel" characteristics of Tang poetry and the superiority of the model.

Next, we divided the training set and validation set according to a ratio of 8: 2. The results are as follows.

Fig. 8: Comparing of methods(529 poems)

We see that with larger validation sets, the accuracy of the DT, LSTM, and Bi-GRU methods has improved, and the accuracy of the CNN-GRU without Reshape layer model has also improved by about 2%. The accuracy of our model is still 63.92%, which is still about 2% higher than the CNN-GRU without Reshape layer method.

VI. Conclusions

In this paper, we propose a novel sentiment analysis scheme based on CNN network and GRU network. Particularly, considering the special structural characteristics of Tang poetry, we propose a 2-channel processing model to extract the features. With the proposed model, we conduct comparative experiments against other methods and perform the best. We make a significant contribution to the sentiment analysis of Tang Poetry.

In the future work, we will continue to expand and optimize our data set. Moreover, the sentiment analysis of poetry in other dynasties of China will be done. We will continuously improve our research

Acknowledgment

This work is partly supported by the National Natural Science Foundation of China under Grant 61877005, 61672109 and 61772085.

REFERENCES

[1] C. D. Manning, C. D. Manning, and H. Schütze, *Foundations of statistical natural language processing*. MIT press, 1999.

[2] M. A. Ahmed, R. A. Hasan, A. H. Ali, and M. A. Mohammed, "The classification of the modern arabic poetry using machine learning," *Telkomnika*, vol. 17, no. 5, pp. 2667–2674, 2019.

[3] P. Sreeja and G. Mahalakshmi, "Comparison of probabilistic corpus based method and vector space model for emotion recognition from poems," *Asian Journal of Information Technology*, vol. 15, no. 5, pp. 908–915, 2016.

[4] M. M. Saad, N. Jamil, and R. Hamzah, "Evaluation of support vector machine and decision tree for emotion recognition of malay folklores," *Bulletin of Electrical Engineering and Informatics*, vol. 7, no. 3, pp. 479–486, 2018.

[5] J. T. Kao and D. Jurafsky, "A computational analysis of poetic style," *LiLT (Linguistic Issues in Language Technology)*, vol. 12, no. 3, pp. 1–33, 2015.

[6] S. Kikuchi, K. Kato, J. Saito, S. Okura, K. Murase, T. Yamamoto, and A. Nakagawa, "Quality estimation for japanese haiku poems using neural network," in *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2016, pp. 1–8.

[7] Z. Wang, W. He, H. Wu, H. Wu, W. Li, H. Wang, and E. Chen, "Chinese poetry generation with planning based neural network," *arXiv preprint arXiv:1610.09889*, 2016.

[8] X. Yi, R. Li, and M. Sun, "Generating chinese classical poems with rnn encoder-decoder," in *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*. Springer, 2017, pp. 211–223.

[9] Y. Hou and A. Frank, "Analyzing sentiment in classical chinese poetry," in *Proceedings of the 9th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH)*, 2015, pp. 15–24.

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

[11] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[12] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[13] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.