Improving Question Classification with Hybrid Networks

Yichao Cao[1,2], Miao Li[1], Tao Feng[1,2], Rujing Wang[1], Yue Wu[3]
1. Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei, 230031, China
2. University of Science and Technology of China, Hefei, 230026, China
3. Institute of Physical Science and Information Technology, Anhui University, Hefei 230601, China
{cycao,ft2016,wuyue16}@mail.ustc.edu.cn, {mli,rjwang}@iim.ac.cn

***Abstract*—Question classification is a basic work in natural language processing, which has an important influence on question answering. Due to question sentences are complicated in many specific domains contain a large number of exclusive vocabulary, question classification becomes more difficult in these fields. To address the specific challenge, in this paper, we propose a novel hierarchical hybrid deep network for question classification. Specifically, we first take advantages of word2vec and a synonym dictionary to learn the distributed representations of words. Then, we exploit bi-directional long short-term memory networks to obtain the latent semantic representations of question sentences. Finally, we utilize convolutional neural networks to extract question sentence features and obtain the classification results by a fully-connected network. Besides, at the beginning of the model, we leverage the self-attention layer to capture more useful features between words, such as potential relationships, etc. Experimental results show that our model outperforms common classifiers such as SVM and CNN. Our approach achieves up to 9.37% average accuracy improvements over baseline method across our agricultural dataset.**

Keywords-question answering; question classification; hybrid networks;

I. INTRODUCTION

QA is a challenging task in Natural Language Processing (NLP), which has drawn significant attention from the past few decades. As an important sub-module of QA, question classification can effectually narrow the space of candidate answers and impacts the quality of QA. To a certain extent, question classification can be regarded as a special case of text classification without sufficient lexical context [1].

However, question sentences with massive amounts of exclusive vocabulary are usually complicated in some particular domains. It is relatively difficult to extract features from question sentences using general classification algorithms, that results in poor classification performance. To address the above specific challenge, we build a novel hierarchical hybrid deep network to improve the accuracy of question classification.

In particular, we first apply denoising and THULAC's Chinese word segmentation [2] to question sentences to improve the quality of data. Then we use word2vec [3] combined with a synonym dictionary to train the word embeddings on the Question-Answer corpus, which can express the semantic information of words. Second, we utilize self-attention mechanism to capture useful features between words of a sentence, and then feed the attention representations to LSTM [4] to compute semantic representations of the question sentence. The sentence representations obtained by the above approaches contain more relationships and latent semantic information than just using word embeddings. Intuitively, it is a better way to handle the sentence features extraction issue in the particular domain. Finally, we exploit CNN [5] to extract the special features from the semantic representations of question sentences, and produce the final classification results through the fully-connected layer and the *softmax* layer. The experimental results demonstrate that the proposed hierarchical hybrid deep network for question classification can improve the average accuracy remarkably.

II. BACKGROUND

The goal of question classification is to classify the question to the anticipated type of the answer. In previous works, empirical rule-based and statistical-based approaches have been applied to the question classification tasks for decades. The rule-based approach generally exploits large-scale pre-defined rules to determine the type of question, which needs a lot of expertise [6,7]. The statistical-based approach is more scalable and versatile than the rule-based approach. [8] combined lexical, syntactic and semantic features to train three different classifiers: K-Nearest Neighbors classifier, naive Bayes classifier, and SVM to classify the question. Besides, many available means like HowNet [9] were introduced to improve the performance of the SVM-based method in the work. However, all of these use syntactic constituency parsing on the input when a trained classification model is applied. Meanwhile, the feature vector space is usually sparse [10].

Thanks to recent advances in deep learning, sentence classification has reached impressive performance. [11] defined a one-layer CNN architecture that uses pre-trained word vectors as inputs and had achieved state-of-the-art results across several datasets. [12] proposed a CNN architecture with multiple convolution layers, positing latent, dense and low-dimensional word vector as inputs. The most important point is that his model adopted a *k-max* pooling strategy, in which the maximum *k* values were extracted from the entire feature map and the relative order of these values was preserved. [13] combined high-order *n*-grams with CNN which contains multiple convolution layers and multiple pooling units associated with different regions. State-of-the-art performances on sentiment classification and topic classification were achieved using this approach. Unlike the above studies only use CNN, we build a hierarchical hybrid deep network consists of self-attention, bi-directional LSTM and CNN layers for question classification.

Self-attention, sometimes called intra-attention is an attention mechanism relating different positions of a single

978-1-7281-5014-7/19/$31.00 ©2019 IEEE

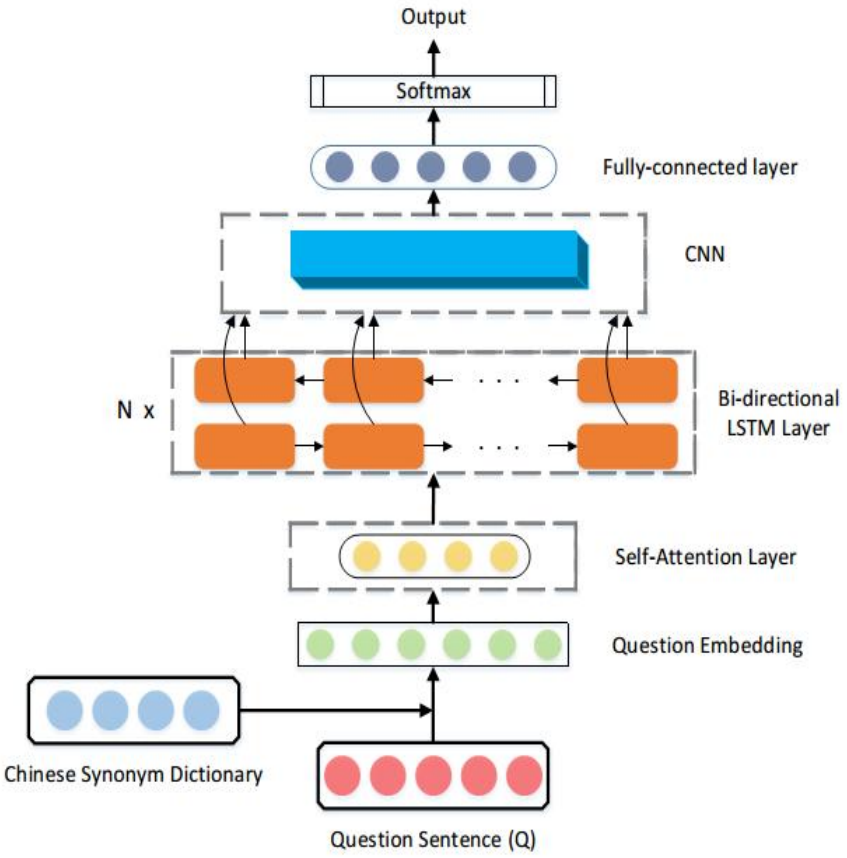

Figure 1. The overview of our model

sequence to achieve features between elements of the sequence. It has been used successfully in a variety of tasks such as learning sentence representations [14], reading comprehension [15], textual entailment [16] and abstractive summarization [17]. Bi-directional LSTM is able to encode the temporal information into representations. Besides, bidirectional architecture can access sequential and complete information about context before and after each time step in a sequence, and obtain the deep bidirectional context representations [18]. CNN is firstly proposed by LeCun for image processing [5], and has also been introduced to address sequential data. For instance, extracting robust and informative features from the sequential inputs [19-22].

It is worth noting that, research on question classification using deep learning methods is still in its infancy. Accordingly, we propose a novel network architecture to classify questions of QA. In the following sections, we describe the details of the proposed model, experiments and related discussions.

III. Model Architecture

A QA text is composed of question and answer, which is different from normal text. Consequently, we have to process QA text and transform questions from sentences into embeddings which are fed into the hybrid network model for question classification. The self-attention layer of the proposed model is capable of achieving latent relationships between word vectors of the question embeddings. Multiple bi-directional LSTM layers are applied to compute a representation of a question sentence using self-attention outputs. CNN layer plays the role of feature extractor to feed better sequential feature representations into the fully-connected layer, and then we utilize the *softmax* layer to obtain the final classification results.

The proposed model follows this overall architecture which uses self-attention, stacked bi-directional LSTM, CNN, fully-connected and *softmax* layers are shown in Figure 1.

A. Question Embeddings

Pre-trained word embeddings are considered to be a part of modern language models, offering significant improvements over embeddings learned from scratch [23]. Similarly to other sequence models, we use learned embeddings to convert the input tokens to vectors of dimension d_m. Specifically, we first use THULAC to perform Chinese word segmentation in question texts which have been denoised, and then we utilize word2vec to train the word embeddings on the QA corpus. Inspired by the work [24], we add Chinese synonym dictionary called *Tongyici cilin* [25] as additional information to the model which can improve the quality of the word embeddings. More specifically, we turn the word embedding of one pair of synonyms into the same one which randomly chooses from these two embeddings. Finally, each sentence in question texts can be converted into a question embedding composed of trained word representations.

B. Self-Attention Layer

In order to capture the internal features of the question sentences for achieving higher classification accuracy, the

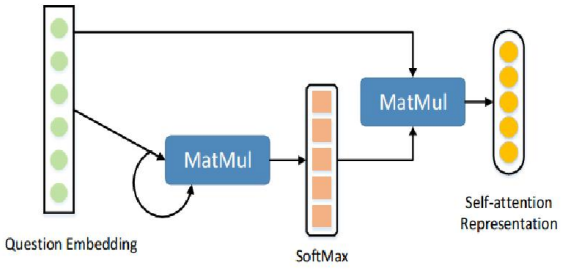

Figure 2. Question self-attention mechanism

self-attention mechanism is applied in our model. As described in Figure 2, assume that a question sentence $S_q=\{w_{q1}, w_{q2}, \ldots, w_{qn}\}$ of QA corpus contains n words. And then we can convert the i-th word w_{qi} into word embedding $\boldsymbol{C_i^q}=\{c_{i1}^q, c_{i2}^q, \ldots, c_{id_m}^q\}$, where c_{ij}^q denotes the j-th element of i-th word embedding $\boldsymbol{C_i^q} \in \mathbb{R}^{dm}$. Further, we can obtain a question embedding $\boldsymbol{S_q^e} \in \mathbb{R}^{n \times d_m}$ for question S_q. We compute the importance degree vector α of different words as:

$$T_q = \tanh\left(W_t\left(S_q^e(S_q^e)^T\right) + b_t\right) \quad (1)$$

$$\alpha = \mathrm{softmax}\left(T_q\right) \quad (2)$$

where $T_q \in \mathbb{R}^{n \times n}$, W_t is the weight matrices, b_t is the bias.

After obtaining the self-attention weight of each word in a question sentence, we can compute the attention representation $\boldsymbol{A_q}$ of $\boldsymbol{S_q^e}$ by the following formula,

$$A_q = \alpha S_q^e \quad (3)$$

where $A_q \in \mathbb{R}^{n \times d_m}$.

It is believed that the self-attention mechanism is able to capture latent information between words to get a higher quality context vector in the next layer.

C. Bi-directional LSTM Stack

LSTM can capture long-term dependencies in time series data and encode temporal information, which has been designed to address text classification [26, 27]. However, the major limitation is that the standard LSTM model is unidirectional, and this restricts the utilization of contexts. In contrast, bi-directional LSTMs can model the input sequence forward and backward, and then capture dependencies of past and future contexts. Figure 3 shows the details of one bi-directional LSTM layer, and this structure includes a forward LSTM and a backward LSTM. The forward LSTM reads the input $\boldsymbol{A_q} = [a_{q1}, a_{q2}, \ldots a_{qn}]$ from left to right, where a_{qi} denotes the i-th vector of the attention representation $\boldsymbol{A_q}$. At each time step t, the hidden state h_t is updated by the following formula.

$$h_t = f_{enc}\left(a_{qt}, h_{t-1}\right) \quad (4)$$

Similarly, the backward LSTM reads the input from right to left:

$$h'_t = f'_{enc}\left(a_{qt}, h'_{t-1}\right) \quad (5)$$

where f_{enc} and f'_{enc} are some nonlinear functions, h_t and h'_t are the hidden states of forward and backward LSTM at time t, respectively.

In this work, we employ l=2 bi-directional LSTM layers. For each of them, we apply $L2$ regularization to improve performance significantly. Then the sentence representation with more context semantic information would be obtained by the above structure, which is used as the input of the next CNN layer.

D. CNN Layer and Classification

In our approach, the CNN layer plays the role of feature extractor and the adopted CNN consists of two sub-layers: one convolutional layer and one pooling layer. The convolutional layer slides the filters over the whole inputs to generate feature mapping. In our model, the one-dimensional convolution operation is used along consecutive sentence representations, and we also employ a set of filters of varying widths to extract informative character patterns. Specifically, we exploit N_f=3 filters with filter size p=3, 4 and 5. And then the pooling layer is applied to extract the most vital features from each feature mapping. In order to obtain the most significant features, we apply the *max-pooling* strategy to the outputs from the convolutional layer. This procedure selects the most salient features to give a final feature vector **v**. In short, we can capture robust semantic features of a question through the CNN layer and prepare for the final classification.

After obtaining the feature vector for a question, we then feed it into the fully-connected dense layer to seek a higher-level representation. The computation in this layer is given by:

$$\mathrm{u} = \mathrm{g}\left(W_{fc}\mathbf{v} + b_{fc}\right) \quad (6)$$

where u is the output of dense layer, the function $g()$ is set to be *ReLU*, W_{fc} and b_{fc} denote the transformation matrix and the bias term, respectively.

Finally, we put the representation u into the *softmax* layer to achieve the conditional probability distribution:

$$\mathrm{y} = softmax(W_s\mathrm{u} + b_s) \quad (7)$$

where W_s and b_s are parameters of this layer. The label corresponding to the value with the highest probability in y stands for the final predicted category for a question sentence. Based on the above, we can try to predict and generate the real classification result considered by our model.

E. Training

Generally, the cross-entropy loss function is used to measure the similarity between predictions and actual values. We minimize the cross-entropy loss function to train the proposed model:

$$\mathrm{J}(\omega) = -\sum_{i}^{N}[\Delta(y_i, y_i')] + \frac{\lambda}{2}\|\omega\|_2^2 \quad (8)$$

where ω is training parameter, N is the number of training samples, y_i and y_i' are real label vector and output probability vector of the i-th sample respectively, Δ is a measure of discrepancy between these two vectors, the sum of token-level cross-entropy losses in our case. λ is a parameter of $L2$ regularization.

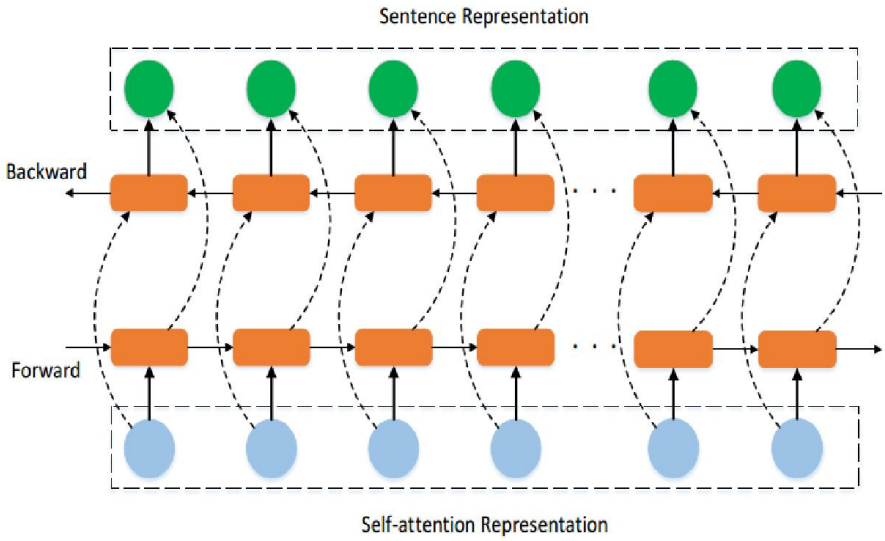

Figure 3. One bi-directional LSTM layer

Furthermore, we used the *Adam* optimizer [28] with $\beta_1 = 0.93$, $\beta_2 = 0.97$ and $\epsilon = 10^{-9}$. We also varied the learning rate over the course of training according to the formula:

$$lrate = slrate \times 0.5^{(global\_step/decay\_steps)} \quad (9)$$

where *slrate* is initial value for the learning rate, and *decay_steps* is the rate of learning rate decay. We also employ *dropout* to relieve the possible *over-fitting* problem during training.

IV. Experiments

A. Descriptions of Dataset

To have a comprehensive understanding about the capacity of our proposed approach, we adopt the agricultural domain dataset to carry out experiments. More specifically, we select five categories of data from the website "Nongye Wenwen"[1] (NW) as the train and test sets. Our agricultural dataset has a total of 10139 samples and five labels, including "Livestock breeding", "Agricultural machinery", "Fruiter planting", "Agricultural materials" and "High-tech related". Statistics of data are shown in Table I.

Besides, we perform denoising and the word segment using THULAC to ensure the quality of the dataset. And then we replace related synonyms of questions using *Tongyici cilin* and produce new question sentences by combining synonyms with original questions, respectively. Finally, we utilize the open source toolkit word2vec to train word embeddings as inputs of our model. Considering the balance of data, 15% of the samples are randomly selected from each category as the test set, and the remaining data are used as the training set.

B. Experimental Setup

We evaluated our approach by comparing with several baseline methods. Both SVM [29] and CNN are typical classification approaches, and we perform them on the dataset as two baselines. However, the above two methods can not address sequential data. Thus we combine LSTM and CNN to encode temporal information of data and then classify questions. Sometimes question sentences are complicated, we have to consider the previous and future contexts of each time step. Bi-directional LSTM can capture dependencies of past and future contexts, we put it on the CNN layer as another baseline method. Besides, the pooling layer can compress generated feature mapping to produce significant features, and affect the final predicted results. We change *max* pooling strategy to *k-max* pooling [12] in the CNN layer of the above approach, i.e. combination of bi-directional LSTM and CNN which uses *k-max* pooling.

For our proposed model, we exploit self-attention and bi-directional LSTM to compute the representations of question sentences. And then CNN, the fully-connected layer and the *softmax* layers are used to produce the final classification result. Similarly, we utilize *k-max* pooling strategy to obtain more significant features. Furthermore, the performance is evaluated by using *Accuracy* for all the methods.

The following methods will be compared in our experiments:

- SVM: Support Vector Machine based approach;
- CNN: Convolutional Neural Network based method;
- LMPC: This baseline method employs both a one-layer LSTM and CNN with max-pooling;
- BLMPC: This baseline approach puts question sentences into a two-layer Bi-LSTM and then CNN with *max-pooling* is used to classify;
- BLKMPC: This baseline method combines a two-layer Bi-LSTM and CNN with *k-max* pooling together for question classification task;
- Our Model: This is our proposed approach contains the self-attention layer, a two-layer Bi-LSTM and CNN with *k-max* pooling.

C. Results and Discussions

In this section, we show a comparison of our model with several benchmark methods and experimental results on the *NW* dataset are shown in Table II. In our work, experimental results of all models are evaluated using the classification accuracy.

[1] http://wenwen.yl01.com/list-1.html

TABLE I. "NONGYE WENWEN" DATA STATISTICS

Category	Label	Train	Test	Total
livestock breeding	0	1701	300	2001
agricultural machinery	1	1713	302	2015
fruiter planting	2	1723	304	2027
agricultural materials	3	1707	301	2008
high-tech related	4	1775	313	2088
Sum	-	8619	1520	10139

From the results, we observe that:

- Our model obtains a very competitive result compared to the baseline systems. More specifically, the classification accuracy of our proposed approach achieves an improvement of 9.37 percent compared to SVM and is 7.57 percent higher than CNN based method. Although the accuracy of our model is not much higher than other baseline methods except SVM and CNN, it has increased by at least 0.2%. The experiment results demonstrate that deep models are able to learn meaningful and discriminative representations from question sentences.
- From Table II, the accuracy of the LMPC method is higher 6.81 percent than CNN based method, indicating that LSTM can enable the model to capture long-term dependencies of sentences and encode them into the abstract representations. We also can find that the BLMPC method performs better than the LMPC method. It is obvious that bi-directional LSTM performs slightly better than LSTM, which owing to LSTM can only access the previous contexts but bi-directional LSTM can encode the question sentences in two directions to obtain better sentence representations.
- Moreover, comparing the classification accuracy of the BLKMPC method with the BLMPC method, we can observe that *k-max* pooling is more effective to question classification than *max* pooling strategy in the CNN layer. Noted that, our proposed hybrid deep network approach combining self-attention mechanism, bi-directional LSTM stack and CNN layers outperforms the above several outstanding baseline methods. Meanwhile, it also indicates that the attention mechanism can effectively capture the latent information between words dynamically and compute a meaningful representation of a question sentence.

It is interesting to observe that our proposed model significantly outperforms all baseline methods across this little training data. Intuitively, it is reasonable to believe that our hybrid deep network model can perform better on large-scale dataset.

V. CONCLUSION AND FUTURE WORK

In this paper, we presented a novel hierarchical hybrid deep network architecture, taking the place of general classification algorithms such as SVM most commonly used in question classification tasks. In our model, self-attention is firstly designed to capture useful features between words of question sentences, and then stacked bi-directional LSTM layers are applied to extend question embeddings depending on the obtained attention representations. Finally, the CNN layer can enable the model to extract question sentence features, and then classification results are achieved by the fully-connected layer and the *softmax* layer. The evaluation of the results on our agricultural dataset shows that the proposed hybrid deep network approach significantly outperforms all baseline methods.

TABLE II. CLASSIFICATION RESULT ON *NW* DATASET.

Methods	Accuracy(%)
SVM	81.63
CNN	83.43
LMPC	90.24
BLMPC	90.77
BLKMPC	90.80
Our Model	**91.00**

In future work, we plan to verify our method with more datasets from different domains. Besides, joint learning may be more effective and efficient in question classification tasks.

ACKNOWLEDGMENT

This work is supported in part by the National Natural Science Foundation of China under Grant No. 61572462 and the 13th Five-year Informatization Plan of Chinese Academy of Science, Grant No. XXH13505-03-203.

REFERENCES

[1] Li X, Roth D. Learning Question Classifiers [J]. Proc. COLING-2002, Taipei, Taiwan, 2002, 12(24):556—562.

[2] Maosong Sun, Xinxiong Chen, Kaixu Zhang, Zhipeng Guo, Zhiyuan Liu. THULAC: An Efficient Lexical Analyzer for Chinese. 2016.

[3] Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality [C]// International Conference on Neural Information Processing Systems. Curran Associates Inc. 2013:3111-3119.

[4] Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8):1735-1780.

[5] Cun Y L, Boser B, Denker J S, et al. Handwritten digit recognition with a back-propagation network[J]. Advances in Neural Information Processing Systems, 1990, 2(2):396--404.

[6] Magnini, B., Negri, M., Prevete, R., Tanev, H.: Mining Knowledge from repeated Co-occurrences: DIOGENE at TREC-2002 Proceedings of the Eleventh Text Retrieval Conference (TREC-2002), Gaithersburg, MD. (2002a).

[7] Hovy E, Gerber L, Hermjakob U, et al. Toward semantics-based answer pinpointing [C]// International Conference on Human Language Technology Research. Association for Computational Linguistics, 2001:1-7.

[8] Mishra M, Mishra V K, Sharma H R. Question classification using semantic, syntactic and lexical features [J]. International Journal of Web & Semantic Technology, 2013, 4(3): 39.

[9] Xu S, Cheng G, Kong F. Research on question classification for automatic question answering [C]//Asian Language Processing (IALP), 2016 International Conference on. IEEE, 2016: 218-221.

[10] Zhen L, Wang X, Yang S. Overview on question classification in question-answering system [J]. Journal of Anhui University of Technology (Natural Science), 2015, 32(1): 48-54.

[11] Kim Y. Convolutional neural networks for sentence classification [J]. arXiv preprint arXiv:1408.5882, 2014.

[12] Kalchbrenner N, Grefenstette E, Blunsom P. A convolutional neural network for modelling sentences [J]. arXiv preprint arXiv:1404.2188, 2014.

[13] Johnson R, Zhang T. Effective use of word order for text categorization with convolutional neural networks [J]. arXiv preprint arXiv:1412.1058, 2014.

[14] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [C]//Advances in Neural Information Processing Systems. 2017: 5998-6008.

[15] Cheng J, Dong L, Lapata M. Long Short-Term Memory-Networks for Machine Reading[J]. 2016.

[16] Paulus R, Xiong C, Socher R. A Deep Reinforced Model for Abstractive Summarization[J]. 2017.

[17] Parikh A P , Täckström, Oscar, Das D , et al. A Decomposable Attention Model for Natural Language Inference[J]. 2016.

[18] Schuster M, Paliwal K K. Bidirectional recurrent neural networks [J]. IEEE Transactions on Signal Processing, 1997, 45(11): 2673-2681.

[19] Gehring J, Auli M, Grangier D, et al. Convolutional Sequence to Sequence Learning[J]. 2017.

[20] He H, Gimpel K, Lin J. Multi-perspective sentence similarity modeling with convolutional neural networks[C]//Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. 2015: 1576-1586.

[21] Bai S, Kolter J Z, Koltun V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling[J]. arXiv preprint arXiv:1803.01271, 2018.

[22] Zhang X, Zhao J, LeCun Y. Character-level convolutional networks for text classification[C]//Advances in neural information processing systems. 2015: 649-657.

[23] Turian J, Ratinov L, Bengio Y. Word representations: a simple and general method for semi-supervised learning[C]//Proceedings of the 48th annual meeting of the association for computational linguistics. Association for Computational Linguistics, 2010: 384-394.

[24] Mei L, Zhou Q, Zang L, et al. Merge information in hownet and TongYiCi CiLin[J]. Journal of Chinese Information Processing, 2005, 19(1): 63-70.

[25] Jiaju M, Yiming Z, Yunqi G, et al. Tongyici cilin [J]. ShangHai Dictionary Publication, 1983.

[26] Zhou C, Sun C, Liu Z, et al. A C-LSTM neural network for text classification[J]. arXiv preprint arXiv:1511.08630, 2015.

[27] Lai S, Xu L, Liu K, et al. Recurrent convolutional neural networks for text classification[C]//Twenty-ninth AAAI conference on artificial intelligence. 2015.

[28] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.

[29] Chang C C, Lin C J. LIBSVM: A library for support vector machines[J]. ACM transactions on intelligent systems and technology (TIST), 2011, 2(3): 27.