Questions and Answers on Legal Texts Based on BERT-BiGRU

Na-Na Zhang\textsuperscript{1*}, Yinan Xing\textsuperscript{2}

\textsuperscript{1} Shanghai Jian Qiao University, China
\textsuperscript{2} Shanghai Ocean University, China
\*Email: nanazhang2004@163.com

Abstract. Using text question and answer technology to improve the efficiency of judicial personnel in the process of case handling can greatly reduce labour costs. This paper proposes a hybrid neural network model that combines pre-training model BERT and Bi-GRU. The model first uses the pre-training model to learn powerful semantic capabilities, then combines Bi-GRU to learn the semantic information between the text and the question, finally gets the answer corresponding to the legal text. The experimental results on the CJRC data set show that compared with the basic baseline model, the algorithm in this paper can effectively improve the accuracy and F1 value.

1. Introduction
With the rapid development of network technology and artificial intelligence, the use of deep learning and other technologies for intelligent processing in professional fields has become a research hotspot. Most of the early methods for text question and answer are to perform semantic matching between the question and the article to determine the similarity to predict the answer, but these methods often require a lot of feature engineering work. With the development of deep learning, it has greatly promoted the development of computer vision and natural language. In the field of computer vision, face recognition \cite{1}, and multi-label image classification \cite{2}, as well as the use of GAN to generate video \cite{3}, and some improved convolutional neural networks to achieve accuracy in image segmentation \cite{4}, these fields’ rate is greatly improved. In the field of natural language, some methods can use convolutional neural networks (CNN) to extract features for question and answer matching \cite{5}, or use Long Short-Term Memory (LSTM) to learn the context relevance of the article sequence to match \cite{6}. There are also some combination variants of neural networks, such as the neural network model combining CNN and LSTM \cite{7}, and the neural network model \cite{8} for the Bidirectional attention mechanism of the question and the article, by obtaining the mutual attention of the question and the article improving the accuracy of the answer. In addition, there are some methods that use Graph Natural Networks (GNN) to perform contextual reasoning to solve some text question answering that requires reasoning \cite{9}.

In recent years, the development of pre-training models has greatly promoted the natural Language development \cite{10}. The pre-training model obtains rich semantic information and syntactic features through unsupervised training on a large-scale corpus in advance, so that it can be transferred to other natural language tasks for targeted supervised learning. Common pre-training models are: the ELMo model, which uses bidirectional LSTM to learn bidirectional information of text \cite{11}; GPT model, which uses Transformer to capture long-distance information \cite{12}, but it is a single direction can only obtain semantic information from left to right \cite{13}; BERT (Bidirectional Encoder Representations from
Transformers) model [14], which uses bidirectional transformer to obtain contextual information and uses self-attention mechanism to train corpus in a parallel way, which has achieved good results on multiple natural language tasks.

Although the pre-training model has learned rich semantic features in a large-scale corpus in advance, there is still much room for improvement in the effect of the model when transferring learning in different fields. Therefore, this paper proposes a Bidirectional Gated Recurrent Unit model based on BERT for text question and answer in the legal field, which has achieved good results on the CJRC: A Reliable Human-Annnotated Benchmark Dataset for Chinese Judicial Reading Comprehension [15] data set.

2. Related Model Algorithm

2.1. BERT Model

In recent years, due to the continuous development of transfer learning, pre-training models have greatly promoted the development of natural language. Among them, the most influential is the BERT pre-training language model proposed by the Google artificial intelligence team [14], which refreshed the best results at the time in 11 natural language processing tasks. Its structure is shown in Figure 1.

![Figure 1. BERT model structure diagram.](image)

The BERT model uses a Bidirectional Transformer encoder, and the training method is divided into two steps: one is to cover 15% of the words by random MASK. Among them, the words marked with [MASK] have 80% probability to directly replace with [MASK] label, 10% probability to replace with any word, 10% probability to retain the original token, let the model predict the meaning of the word masked by MASK; select sentence pairs in the training text, including continuous sentence pairs and non-continuous sentence pairs, and let the model determine whether the sentence pairs have contextual semantic relations.

The Transformer in the BERT model mainly uses its encoding layer, and the specific unit structure is shown in Figure 2. After inputting the text, embedding is performed to vectored the text words, and then perform Positional Encoding. In order to pay attention to the semantic relationship between different words in each sentence, a self-attention layer is added for encoding. At the same time, considering the different attention points of words in different heads, multi-head attention mechanism to make the model get more information. The output of the self-attention layer based on the multi-head mechanism will pass through the Add &Norm layer, where the Add represents the residual connection, and the Norm represents the layer normalization. The output after the position information is encoded with the multi-head mechanism. The output of the attention layer is added, and then the layer normalization operation is performed, so as to retain as much information as possible and make the model easier to train. The output of the Add &Norm layer is passed to the feed forward neural network layer, and then output after the Add &Norm layer.
Transformer completely uses the Self-Attention mechanism as the basic structure of the model, abandoning the previous CNN and RNN networks to solve the long-term dependency in NLP, and has the advantage of parallel computing. The calculation formula is as follows:

$$\text{Attention}(Q, K, V) = \text{soft max}(\frac{QK^T}{\sqrt{d_k}})V$$

(1)

Among them, Q represents the Query vector, K represents the Key vector, and V represents the Key vector. They are the mapping matrix of the input vector of the encoder, and $d_k$ represents the dimension of the input vector.

![Transformer Encoder](image)

**Figure 2.** Transformer Encoder.

The input of BERT is a three-part position of word embedding, sentence embedding and position embedding to represent an input text sequence. As shown in Figure 3, Token Embeddings represents a word vector, and the first word is a CLS tag, which can be used for different tasks downstream of NLP; Segment Embeddings represents a sentence vector, used to distinguish two sentences; Position Embeddings represents what the BERT model learns to the position vector.

![BERT input vector representation](image)

**Figure 3.** BERT input vector representation.

The input of BERT is a three-part position of word embedding, sentence embedding and position embedding to represent an input text sequence. As shown in Figure 3, Token Embeddings represents a word vector, and the first word is a CLS tag, which can be used for different tasks downstream of NLP; Segment Embeddings represents a sentence vector, used to distinguish two sentences; Position Embeddings represents what the BERT model learns to the position vector.

2.2. Gated Recurrent Unit Model

Recurrent Neural Networks (RNN) are often used in sequence tasks to capture the information of long sequences. When the sequence length is too long, the gradient will disappear. It is difficult to learn the intermediate long-term dependence features. Long-short-term memory neural network (LSTM) [16] has made great improvements to traditional RNNs, introducing memory units and threshold mechanisms to capture long-distance information and solve the problem of gradient disappearance. Due to the excessive number of LSTM parameters, a variant of LSTM: Gated Recurrent Unit (GRU), proposed by Cho, et al. (2014) [17], combines the forget gate and the input gate into a single update gate. It also merged cell state and hidden state, and made some other changes. The result of the model is simpler than the standard LSTM model and the effect is not bad. Its structural unit is shown in Figure 4:
Figure 4. GRU cell structure diagram.

The output of the hidden layer of the GRU network is shown in formula (2) - formula (5):

\[ z_t = \sigma \left( W^z x_t + U^z h_{t-1} \right) \]  

\[ r_t = \sigma \left( W^r x_t + U^r h_{t-1} \right) \]  

\[ h_t^- = \tanh \left( W_{h'} x_t + r_t U_{h_{t-1}} \right) \]  

\[ h_t = (1 - z_t) * h_{t-1} + z_t * h_t^- \]  

Among them, formula 2 and formula 3 respectively represent reset gate \( r_t \) and update gate \( z_t \), and formula 4 represents candidate hidden layer \( h_t^- \). The \( h_t^- \) represents the new information at the current moment, and \( r_t \) is used to control how much previous memory is retained. The \( z_t \) controls how much information needs to be forgotten from the hidden layer \( h_{t-1} \) at the previous moment, and how much hidden layer information \( h_t^- \) needs to be added at the current moment, and finally get \( h_t \).

3. BERT-BiGRU Model

This paper combines the BERT model and the Bidirectional GRU neural network to design a question and answer model for the legal field. The specific structure of the model is shown in Figure 5. First, stitch the text questions and paragraphs into the BERT model, and then perform bi-directional feature extraction through the Bi-GRU neural network layer. Considering that as the number of network layers deepens, the neural network will have features in the forward propagation process weaken, so in order to ensure that the information is not lost, the hidden state of the BERT and the hidden layer of the Bi-GRU neural network layer are spliced together and input to the discrimination layer to predict the answer.

Figure 5. BERT-BiGRU network structure diagram.

3.1. BERT Embedding Layer
Since natural language cannot be directly used as input, it is necessary to first convert natural language into a computer-operable numerical form. The most commonly used feature representation method in natural language is to use One-Hot coding. Because it is a discrete code, there are obvious problems with this way of representation: (1) Different words are always orthogonal, and the similarity between different words cannot be measured. (2) It can only reflect whether each word appears, but it cannot be prominent the difference in importance between words. In order to overcome these shortcomings, distributed representations are introduced, which are called word representations or word embedding. Classic distributed word vectors, such as Word2vec [18], GloVe [19], these word vectors are difficult to learn more contextual information to represent the ambiguity of words. However, the BERT model is learned from a large-scale unlabeled corpus in an unsupervised manner, and fully considers the relationship characteristics of character-level, word-level, sentence-character level, and sentence-to-sentence to enhance the semantic representation of word vectors. Therefore, the BERT model can be used as an embedding layer. Compared with the traditional word vector embedding layer, the BERT can be directly used as a part of the neural network and connected to other neural networks behind it to improve the overall effect of the model. The introduction of the pre-training model BERT can make good use of the context relationship to greatly improve the lower limit of the model.

Combine the given question and the corresponding text to get \((Q, P)\), where \(Q\) represents the question and \(P\) represents the corresponding text. Use the BERT pre-training model for semantic learning to obtain the corresponding semantic representation \(h_{(q, p)}\). For the input question, it can form a question-and-answer pair \((Q, P)\) to get:

\[
h_{(q, p)} = BERT \left( Q, P, W_b \right)
\]

Among them, \(Q\) represents the question, \(P\) represents the corresponding text, and \(W_b\) represents the parameters corresponding to the BERT model.

### 3.2. BiGRU Information Extraction Layer

In order to increase the upper limit of the model, it is a better way to choose to connect other neural networks behind the pre-training model BERT. The GRU model performs well in the context information extraction process of the text. Its relatively simple structure and few parameters can reduce the training time of the model.

Bidirectional Gated Recurrent Unit (Bi-GRU) is a combination of forward GRU and backward GRU. It calculates the input sequence in order and reverse order to obtain two different hidden layer representations, and then passes the vector stitching method obtains the final hidden layer feature representation.

The specific structure of Bi-GRU is shown in Figure 6, where the two GRUs are the forward pass module and the backward pass module. The \(X_N\) represents the word vector input at the N position, and the \(h_F, h_B\) are the forward pass hidden at the N position layer output and N position backward pass hidden layer output, and then the final hidden layer feature representation \([h_B, h_F]\) is obtained through vector splicing.

![Figure 6. Bi-GRU network structure diagram.](image-url)
Input the output $h(q, p)$ of the BERT embedding layer into the bidirectional GRU to obtain the joint representation of the forward GRU and the backward GRU:

$$h_t = BiGRU\left(h_{(q, p)}, W_t\right)$$

Among them, $h(q, p)$ represents the semantic representation of BERT learning, and $W_t$ represents the parameters corresponding to the Bi-GRU model.

### 3.3. Feature Fusion and Classification Layer

In order to reduce the loss of information, finally the hidden state information $h_{(q, p)}$ of BERT and the information $h_t$ extracted by Bi-GRU are spliced to obtain a joint representation vector:

$$H = \left[h_{(q, p)}, h_t\right]$$

Input the obtained final vector representation $H$ into softmax for normalization operation, and calculate the probability that the answer belongs to each category:

$$p_{(y_i)} = \text{soft max}\left(H\right)$$

Among them, $p_{(y_i)}$ is the probability of predicting answer category $i$, $0 \leq p_{(y_i)} \leq 1$, $0 \leq i \leq 3$.

### 3.4. Model Training

The goal of model training is to minimize the loss function. The calculation formula of the loss function is as follows:

$$Loss = \sum_{(Q, P, Y) \in D} -\log\left(y = i \mid (Q, P)\right)$$

where $D$ is the question and answer pair training set, $i \in \{0, 1, 2, 3\}$, 0 means the answer is YES, 1 means the answer is NO, 2 means the answer is unanswerable, and 3 means the answer is Span of words. The gradient optimization algorithm Adam [20] is used to adjust the learning rate, use back propagation to iteratively update the model parameters, and minimize the loss function to train the model.

### 4. Experiment and Analysis

#### 4.1. Experimental Data Set and Evaluation Criteria

The data set used in this paper comes from CJRC (Chinese Judicial Reading Comprehension, CJRC) [15]. Because the data set is too large, only a part of the data set is selected for experiment in this experiment. The selected data set includes both civil and criminal a total of 2000 data sets, including 10,000 questions and 4 answer types are Span of words, yes, no, and unanswerable. Visualizing the number of answer types in the data set is shown in Figure 7. It can be seen that Span of words has the most answer types, which also means that most of the answers need to be extracted. The experiment divides the data into training set and validation set according to the ratio of 4:1. Therefore, the experiment did not use the BERT version of the Baseline given by CJRC, but used the results obtained from the Bert training segmentation data alone as the Baseline. The performance evaluation standard of the model adopts the accuracy rate and Macro-average F1.
4.2. Model Construction and Specific Parameter Settings
The text question answering algorithm model mentioned in this article is tested in the Ubuntu18.04 system environment and the configuration environment of Python 3.7 and Pytorch 1.5.1. The GPU of the hardware environment is GTX-1080Ti with 11G memory. The batch size of the training set and the test set is 8, the epoch is 6, the sequence_length is 256, and the learning rate is $2 \times 10^{-5}$. There are two commonly used versions of BERT's pre-training model. The model parameters are shown in Table 1. L represents the number of layers, H represents the number of hidden layers, A represents the number of self-attention heads, and Parameter represents the amount of parameters. This experiment uses the BERT-Base-Chinese model version for experiments. This model has 12 layers, the hidden layer is 768, 12 headers, and 110M parameters.

| Model     | L   | H    | A   | Parameter/M |
|-----------|-----|------|-----|-------------|
| BERT-Base | 12  | 768  | 12  | 110         |
| BERT-Large| 24  | 1024 | 16  | 340         |

4.3. Experimental Results and Analysis
In order to verify the effect of the BERT-BiGRU model used in this paper, in the same experimental environment, four models of BERT, BERT-CNN, BERT-MLP, and BERT-BiGRU-MLP were used as comparative experiments in the experiment process. The accuracy and F1 value evaluates the model recognition effect. The changes in accuracy and F1 value of the four models with the increase of the number of iterations are shown in Figure 8 and Figure 9, and the experimental results of the four models are shown in Table 2.

Figure 7. Distribution of answer types in the data set.

Figure 8. Comparison of accuracy of iteration times between different models.
Figure 9. Iterative changes in F1 values between different models.

Table 2. Model experiment results table.

| Model                  | Accuracy (%) | F1-value (%) |
|------------------------|--------------|--------------|
| BERT                   | 83.65%       | 56.96%       |
| BERT-BiGRU-MLP         | 84.26%       | 59.50%       |
| BERT-CNN               | 84.12%       | 59.49%       |
| BERT-MLP               | 84.31%       | 59.87%       |
| BERT-BiGRU             | 84.38%       | 59.95%       |

From the experimental results in Table 2, BERT-BiGRU has a good improvement in accuracy and recall rate compared with the BERT baseline model. This shows that using BiGRU with good contextual semantics on the basis of the BERT model can make the performance of the model becomes better. However, the effect of the BERT-BiGRU-MLP model after the addition of the Multilayer Perceptron has not been improved. The most likely reason is that as the number of network layers increases, the information carried by the neural network has attenuated, which also proves that our model splicing the information of the BERT layer with the information obtained by BiGRU will improve the performance of the model.

5. Conclusion
Aiming at the question and answer data set of legal text, this paper proposes a combined neural network model. Compared with the baseline model of BERT itself, the model effect of BERT-BiGRU has been improved in both accuracy and recall. However, the experiment is conducted on a small-scale data set, and the model can show good accuracy and F1 value. In the future, the scale of the data set can be increased to further improve the accuracy and F1 value of the model. In terms of models, in the future, we will consider introducing an attention mechanism into BiGRU behind BERT for training to further improve model performance.
Acknowledgments
This work was supported by the Shanghai Municipal Education Commission’s "Morning Plan" project (NO. AASH1702).

References
[1] Wen S P, Dong M H, Yang Y, Zhou P, Huang T W, and Chen Y R 2019 End-to-end detection-segmentation network for face labeling IEEE Transactions on Emerging Topics in Computational Intelligence, doi: 10.1109/TETCI.2019.2947319
[2] Wen S P, Liu W W, Yang Y, Zhou P, Yan Z, Guo Z Y, Chen Y R, and Huang T W 2020 Multi-label image classification via feature/label co-projection IEEE Transactions on Systems, Man and Cybernetics: Systems, doi: 10.1109/TSMC.2020.2967071
[3] Wen S P, Liu W W, Yang Y, Zeng Z G, and Huang T W 2019 Generating realistic videos from keyframes with concatenated GANs IEEE Transactions on Circuits and Systems for Video Technology, 29(8) 2337-48
[4] Wen S P, Wei H Q, Huang T W, and Zeng Z G 2018 Memristive fully convolutional networks: an accurate hardware image-segmentor in deep learning IEEE Transactions on Emerging Topics in Computational Intelligence 2(5) 324-34
[5] Dong L, Wei F, Zhou M, et al. 2015 Question Answering over Freebase with Multi-Column Convolutional Neural Networks Meeting of the Association for Computational Linguistics & the International Joint Conference on Natural Language Processing
[6] Zhou X, Hu B, Chen Q, et al. 2015 Answer Sequence Learning with Neural Networks for Answer Selection in Community Question Answering Computer Science doi 10.3115/v1/P15-2117
[7] Zhang Y, Chen Y, Liu Z. 2020 Hybrid Neural Network Model for Community Question and Answer Matching Small Microcomputer System, 41(9) 1833-8 (in Chinese)
[8] Seo M, Kembhavi A, Farhadi A, et al. 2016 Bidirectional attention flow for machine comprehension arXiv preprint arXiv:1611.01603
[9] Qiu L, Xiao Y, Qu Y, et al 2019 Dynamically fused graph network for multi-hop reasoning Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics pp 6140-6150.
[10] Qiu X, Sun T, Xu Y, et al 2020 Pre-trained models for natural language processing: A survey. arXiv preprint arXiv:2003.08271, 2020.
[11] Peters M E, Neumann M, Iyyer M, et al. 2018 Deep contextualized word representations. arXiv preprint arXiv:1802.06365
[12] Vaswani A, Shazeer N, Parmar N, et al. 2017 Attention is All you Need Neural Information Processing Systems 2017 5998-6008
[13] Radford A, Narasimhan K, Salimans T, et al 2018 Improving language understanding by generative pre-training Available at https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf
[14] Devlin J, Chang M W, Lee K, et al 2018 Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805
[15] Duan X, Wang B, Wang Z, et al. 2019 Cjrc: A reliable human-annotated benchmark dataset for chinese judicial reading comprehension China National Conference on Chinese Computational Linguistics. (Springer, Cham) pp 439-451
[16] Hochreiter S, Schmidhuber J 1997 Long Short-Term Memory Neural Computation 9(8) 1735-80.
[17] Cho K, Van Merriënboer B, Gulcehre C, et al 2014 Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078
[18] Mikolov T, Chen K, Corrado G S, et al. 2013 Efficient Estimation of Word Representations in Vector Space. International Conference on Learning Representations
[19] Pennington J , Socher R , Manning C 2014 Glove: Global Vectors for Word Representation Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)
[20] Kingma D, Ba J 2014 Adam: A method for stochastic optimization: Computer Science arXiv:1412.6980v8