information

Article

Attention-Based Transformer-BiGRU for Question Classification

Dongfang Han, Turdi Tohti * and Askar Hamdulla

College of Information Science and Engineering, Xinjiang University, Urumqi 830017, China;
easth@stu.xju.edu.cn (D.H.); askar@xju.edu.cn (A.H.)
* Correspondence: turdy@xju.edu.cn; Tel.: +86-139-9999-4696

Abstract: A question answering (QA) system is a research direction in the field of artificial intelligence and natural language processing (NLP) that has attracted much attention and has broad development prospects. As one of the main components in the QA system, the accuracy of question classification plays a key role in the entire QA task. Therefore, not only the traditional machine learning methods but also today's deep learning methods are widely used and deeply studied in question classification tasks. This paper mainly introduces our work on two aspects of Chinese question classification. The first is to use an answer-driven method to build a richer Chinese question classification dataset for the small-scale problems of the existing experimental dataset, which has a certain reference value for the expansion of the dataset, especially for the construction of those low-resource language datasets. The second is to propose a deep learning model of problem classification with a Transformer + Bi-GRU + Attention structure. Transformer has strong learning and coding ability, but it adopts the scheme of fixed coding length, which divides the long text into multiple segments, and each segment is coded separately; there is no interaction that occurs between segments. Here, we achieve the information interaction between segments through Bi-GRU so as to improve the coding effect of long sentences. Our purpose of adding the Attention mechanism is to highlight the key semantics in questions that contain answers. The experimental results show that the model proposed in this paper has significantly improved the accuracy of question classification.

Keywords: QA system; question classification; deep learning; Transformer; Bi-GRU; Attention

check for updates

Citation: Han, D.; Tohti, T.; Hamdulla, A. Attention-Based Transformer-BiGRU for Question Classification. *Information* **2022**, *13*, 214. https://doi.org/10.3390/info13050214

Academic Editor: Kostas Vergidis

Received: 11 March 2022
Accepted: 15 April 2022
Published: 20 April 2022

1. Introduction

According to the prediction of Data Age 2025 white paper released by IDC, in 2025, the amount of global data will reach an unprecedented 163ZB [1]. All walks of life are constantly generating data every day: Mobike generates 25 million orders per day, 50 million messages per day from Twitter, Youtube uploads more than 400 h of video per minute, Taobao generates 20 tb data every day, Facebook generates 300 tb data every day, and Google processes 24 pb data every day. In this age of information explosion, people are often dissatisfied with search engines simply returning to a related page, especially in specific areas, such as law, health care, etc. As traditional search engines return more web pages, it is more difficult to find the key information they need. However, a QA system can better identify users' intentions and meet their needs for obtaining information quickly and accurately, which has become one of the current research hotspots.

The question and answering system is an information retrieval system that accepts questions from users in natural language (e.g., what is the longest river in the world?) and finds accurate, concise answers to those questions (e.g., the Nile) from a large amount of heterogeneous data. There is a fundamental difference from traditional search engines. The goal of a question and answering system is to accurately answer the questions that users ask in natural language. Compared with traditional search engines that search based on keywords and return a collection of relevant documents, question answering systems focus more on accuracy. In addition, regarding question classification as a special form of text classification, including in sentiment analysis, label classification, news text classification, and other text classification subordinate tasks above, ideas and methods can refer to and learn from each other. Therefore, the research and implementation of a question classification system is of great importance to improve the performance of a question and answer system and regarding how text classification can obtain valuable information and improve information efficiency.

Reviewing and summarizing the history and current state of research in question and answering systems will help to promote the development of a question and answering system as well as question classification technology.

In the early 1960s, researchers tried to build an intelligent system that could answer people's questions to meet the development of artificial intelligence (AI). This period is known as the AI period, which is mainly devoted to AI systems and expert systems, represented by systems such as BASEBALL [2] and LUNAR [3], which are mainly domain-limited question and answer systems that deal with structured data.

In the 1970s and 1980s, due to the rise of computational linguistics, a large amount of research focused on how to use computational linguistics technology to reduce the cost and difficulty of constructing QA. This period is known as the computational linguistics period, which mainly focuses on defining the field and processing structural data, and the representative system is Unix Consultant [4].

In the 1990s, QA entered a new period of open domain and text-based systems. Along with the rapid development of the Internet, a large number of electronic documents were generated, which provided objective conditions for QA to enter the open domain, text-based period. Especially since the establishment of the QA track of TREC (text retrieval conference) in 1999, the development of question and answer systems has been greatly promoted. Subsequently, frequently asked questions (FAQ) data appeared on the Web; especially since the end of 2005, a large number of community-based question answering (CQA) data (e.g., Yahoo! Answer) appeared on the Web. With a large amount of question–answer pair data available, the QA entered the open domain, question–answer pair-based period.

In recent years, with the development of deep learning technology, research on question answering systems based on deep learning methods has emerged. For QA in the open field, predecessors have done a great deal of research. In 2014, the GoogleBrain [5] team and the Yoshua Bengio [6] team published respective articles, and the two articles coincided with the idea of solving machine translation, which is the seq2seq model. In 2015, Kyunghyun, Bengio, and Bahdanau proposed an attention mechanism on the basis of seq2seq [7], which improved the accuracy of translation. There are already some mature QAs in industry and academia. For example, Microsoft's "Xiaobing", Apple's "Siri", and so on.

QA is a computer system that interacts with the user in natural language, which can automatically process questions asked by users and give users concise and correct answers. A general QA system consists of three parts: question analysis, information retrieval, and answer selection [8]. The structure of a typical QA system is shown in Figure 1.

It can be seen from Figure 1 that question classification is the initial part of the QA task, which has an important influence on the subsequent answer extraction and the overall performance of the QA system. In short, question classification plays an important role in QA, mainly manifested in two aspects:

(1) Assign the corresponding label to the question according to the expected answer type, thereby narrowing the range of candidate answers. For example, in the question "Who was the first Chinese to enter space?", the answer that users really want to know is "Yang Liwei" instead of searching for too many materials containing content related to "first" or "space". After question classification, it can be learned that this is a question asking for a person's name. Therefore, the candidate statement outside the person's name will be screened out in the answer extraction stage, and it is only necessary to focus on some answers related to the person's name without having to pay too much attention to

the candidate answer statement unrelated to the person's name. This helps to improve the accuracy of answer selection and reduce the amount of calculation.

(2) For different question types, QA will develop different strategies in subsequent operations. For example, the question "What kind of food is there in Sichuan?" The answer to the question is about the food category. The focus of the extraction should also be placed on the selection of food-related strategies.

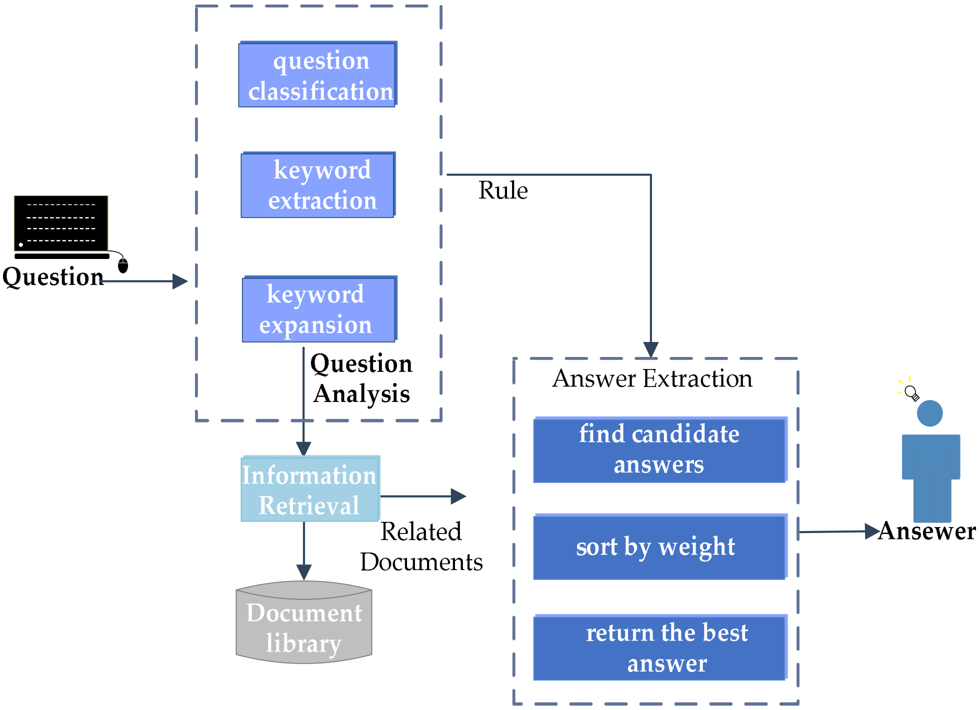

Figure 1. A typical QA system structure.

Researchers have reached a consensus that the accuracy of question classification results plays a key role in answer selection, and even the efficiency and performance of the entire QA system. The experimental results of Moldovan et al. [9]. indicate that the wrong answer is caused by the inaccurate question classification.

From the perspective of the task category, question classification also belongs to text classification. Therefore, the main methods of question classification usually refer to and quote some ideas of text classification [10], but they are different in some details. For example, common words (stop words) such as "what" and "is" are usually filtered in text classification, but these words are often very important in question classification. In addition, questions are all natural language questions randomly posed by users rather than traditional normative texts. Therefore, compared with conventional text classification, a question classification task mainly faces two major challenges.

The first challenge is that the user's question is too short, has a small vocabulary, and contains little information. For example, the Chinese question "007?" should be classified as an entity, and it is difficult to determine whether "007" is a number or a movie in this question.

The second challenge is that the questions are too long, such as the English question "why do people get goosebumps when they have something emotional happen to them, like when they hear a beautiful piece or see something beautiful, or get aroused by someone they love ?" The question should be classified as DESC (Description category), but, because the question is long and has a lot of entity words, it is easy to be misclassified in other categories, which is one of the difficulties.

Early question classification mainly used rule-based matching methods [11], which required the manual formulation of a large number of rules and establishment of a rule base.

In recent years, question classification methods mainly develop lexical, syntactic, semantic, and other feature extraction strategies for question sentences [12] and then classify question sentences with the help of machine learning (e.g., K-nearest neighbor, SVM, Bayesian, etc.) methods. The accuracy of its classification results is determined by the merit of feature extraction, and, the richer the extracted features, the higher the accuracy of classification. The rule-based and feature extraction methods have the following three shortcomings.

(1) The manually developed feature extraction strategy is somewhat subjective and cannot comprehensively understand the interrogative sentences.

(2) In order to achieve better classification results, the feature extraction strategy needs to be constantly adjusted and optimized to better represent the question sentence in syntactic and semantic aspects, which is not very flexible.

(3) When the syntactic complexity of the question is high or the category granularity of the question is small, it is more difficult to develop feature rules and the classification effect is not good.

Compared with the above method, deep learning technology is undoubtedly a new research hotspot in the branch of machine learning, which has become a powerful force to promote the rapid development of machine learning. The vigorous development of deep learning technology provides technical support for the research of question and sentence classification, which will play a powerful role in promoting the development of QA in the future and will become a trend.

Based on the above summary, the main research of this article is summarized as follows:

- A brief introduction to traditional machine learning question classification models based on statistical methods is presented.
- There is no unified public dataset for the Chinese question and answer corpus compared to the English question and answer corpus, and the small amount of corpora for Chinese question classification and insufficient resources is one of the main reasons that restrict the accuracy of question classification. To solve the problem, thousands of question and answer sentences were captured from Chinese community question and answer platforms, including Baidu Know and Sogou Q&A, and the categories were manually marked and the missing and noisy samples were processed, finally, after manual verification. In addition, the difficulty of short questions with little information can be better solved by combining the answer information of the question with the classification of the question. Through experimental comparison, it is found that the question sentence containing the information-rich features of the answer has a better classification effect than the single question sentence.
- For longer question sentences that contain answer information, a hybrid neural network (TBGA) model that combines Transformer and Bi-GRU and includes the Attention mechanism is used. The input of Transformer's encoder is the sum of word vector and position vector, which can obtain the relationship between words and capture the internal characteristics of question sentences; Bi-GRU can consider the context on the basis of a time series and has good dependence on long sequences. The effect of longer question information can also be captured well. The introduction of Attention can highlight the key feature information based on the features extracted from the above network, thus avoiding complicated words and lengthy interrogative sentences from affecting the classification results. In addition, the answer content of question sentences is introduced to enhance the information of question sentences, which ultimately improves the efficiency and accuracy of question sentence classification. The experimental results show that the question classification method used in this question helps to improve the accuracy of question classification.
- Experimentation with multiple deep models on TREC, a classical dataset in the publicly available open domain, and comparison of fine-grained category accuracies are performed to identify classification models that are superior for application in different category domains.

2. Materials and Methods

The rest of this article is organized as follows. The second chapter introduces related work, including Chinese and English question classification system standards, as well as the machine learning classification methods mainly used in the past few years and the current research status of popular deep learning technologies in this field. The third part introduces the method we proposed and the dataset used in the experiment. The fourth part shows the experiment and results. Finally, the paper is concluded in the fifth part.

2.1. Question Classification

Question classification is a very important part of the QA. It needs to classify the question into a certain category according to its answer type. The subsequent retrieval and extraction will adopt different measures according to the question category.

Firstly, question classification must determine the classification standard. The standard of classification is the basis and premise of question classification. As shown in Table 1 [13], English mainly adopts UIUC's question classification standards. In this classification system, question sentences are divided into 6 coarse categories and 50 fine categories, and the question category is determined by the type of answer.

Table 1. UIUC classification standards.

Coarse	Fine
ABBR	abbreviation, expansion
DESC	definition, description, manner, reason
ENTY	animal, body, color, creation, currency, disease, event, food, instrument, language, letter,other, plant, product, religion, sport, substance, symbol, technique, term, vehicle, word
HUM	description, group, individual
LOC	city, country, mountain, other, state
NUM	code, count, date, distance, money, order, other, percent, period, speed, temperature, size,weight

China has not yet established a unified Chinese question classification standard. As shown in Table 2, Harbin Institute of Technology's Wen Xu et al. [14] are based on the existing question classification standards in English, and, in view of the complexity of Chinese, they formulated a set of Chinese question classification standards that has 7 coarse categories and 60 fine categories, each large class containing several small categories.

Table 2. Chinese classification standards.

Coarse	Fine
DESC	abbreviation, meaning, method, reason, definition, describes, description other
HUM	specific person, organization, character description, character list, human other
LOC	planet, city, continent, country, province, river, lake, mountain, range, ocean, island, location, listed, address, location other
NUM	number, quantity, price, percentage, distance, weight, temperature, age, area, frequency, speed, range, order, number list, number other
TIME	year, month, day, time, time range, time list, time other
OBJ	animals, plants, foods, colors, colors, currency, language, text, material, mechanical form, religious entertainment entity, entity, entity other
Unknown	unknown

2.2. Traditional Question Classification Method

After determining the question category, then there are the methods and models used in the classification. In the past decades, the use of machine learning methods has been the mainstream method of question classification research, and the quality of question feature extraction is the decisive factor of classification performance; question classification methods based on machine learning are mainly divided into two categories: one is a classification method based on empirical rules and the other is a machine learning classification method based on statistics [15]. The former method is more common in the early stage, and it is mainly based on preset empirical rules and templates to distinguish the types of question sentences [16]. Statistics-based machine learning methods are relatively common in recent years, and they have the advantages of strong versatility, ease of transplantation, and expansion. This method first needs to extract some feature vectors, the category of the question is represented by these feature vectors, and then the real test questions that have been accurately labeled are learned through statistics and analysis so as to automatically build a classifier and finally use the classifier to mark the category to which the question belongs. The core of the statistics-based machine learning classification method is to extract the feature vector of the question sentence. The classification accuracy of the classifier is usually affected by the quality of the feature vector.

Zhang [17] uses word bag and multi-word chunks as the main features and adopts the Bayesian model to classify English training sets of different sizes; Silva et al. [13] use only a single word bag as a classification feature and combine Support Vector Machine (SVM) to classify the UIUC English question set; Lee et al. [17] use bag of words and word blocks (including all consecutive word sequences in the question), which is the main feature, and the K-Nearest Neighbor (KNN) is used to classify the UIUC10 English question set; Sundblad [18] uses bag of words as a classification feature to classify in the TREC10 English question set. The classification accuracy of the large category is 67.2%, and the classification accuracy of the small category is 60.0%; Li et al. [19] proposed a classification method for Transformation-based Error-driven Language rules (TBL) and adopted the English synonym set from WordNet, the concept of hypernyms from nouns, and Minipar's dependency relationship and other basic language knowledge as the characteristics of the question classification method and achieved 91.4% classification accuracy on the adopted English public question set; THINT M et al. [20] proposed to use the sentence's hypernym, the head word is used as a feature, and the Maximum Entropy model (ME) is used to classify the UIUC English question set.

2.3. Deep Learning Technology

In recent years, deep learning technology is undoubtedly a new research hotspot in the branch of machine learning. The question classification method based on deep learning has strong adaptive learning capabilities and relatively high fault tolerance. Some larger noises and complex deformations also have higher resistance. In the field of natural processing, more and more scholars and researchers use deep learning methods to solve problems.

Commonly used deep learning classification methods are CNN, RNN, and Attention. Many deep learning classification models are improved on the basis of these methods. For example, DPCNN is an improvement on CNN, while BILSTM and BIGRU are improvements on RNN, and BERT pre-training model, the core component Transformer, a multi-head attention mechanism, is an improvement on Attention.

3. Question Classification Method Based on Deep Neural Network

The application of traditional machine learning classification algorithms in the field of question classification is mainly based on the question classification dataset, manually extracting the characteristics of the question, or combining certain features to represent the question, so it has a strong subjectivity. Moreover, it has a relatively diverse language expression, which means that it has a relatively high cost of manually formulating an accurate feature extraction method.

Deep learning has undoubtedly become a powerful force driving the rapid development of machine learning nowadays, and deep learning methods have been playing a huge role in different fields, such as image processing, speech recognition, and natural language processing. Deep learning methods do not require many corresponding rules in advance, and there is no complicated feature engineering to obtain feature representations, especially the emergence of word vector techniques, such as word2vec, glove, and textfast [21], which use words as network parameters to supervise the training of randomly initialized vectors through the network, thus making deeper extensions of deep neural networks possible. The semantic and other information of the text is also more informative.

The development of deep learning techniques is driving research related to interrogative sentence classification. For example, Kim et al. [22] classified English questions by turning question sentences into word vectors and using Convolutional Neural Network (CNN); Kominos et al. [23] studied the effect of word embedding on deep neural networks, and the results show that context-based word embedding can achieve better classification results in question classification tasks. Xu Jian et al. [24] used LSTM to construct a two-channel LSTM question classification model on both Chinese and English corpora, which can be more extensively obtained through text translation, thereby improving the accuracy of question classification; Shi et al. [25] introduced the Attention mechanism to extract the characteristics of the question and made full use of the answer information of the question to enhance the expression of the question, more effectively extract the effective information in the question, and grasp the semantic key points. The experimental results show that the question classification method that introduces the attention mechanism improves the YahooAns question set and CQA question set by 4% and 7%, respectively, compared with the model not introduced in the article.

As introduced at the end of the previous section, some neural networks have their own defects and shortcomings and are not able to fully learn the feature information and semantic information of question sentences. Therefore, this paper tries to explore a deep learning framework that is more effective and suitable for question classification by combining multiple deep learning methods. In addition, the Chinese open question corpus is relatively small, and experiments and studies are usually conducted based on small samples of interrogative sentences, such as the dataset size of 1500 used by Yu Zhengtao et al. [26], the dataset size of 4280 used by Tian Weidong et al. [27], and Li Ru et al. [28]. All of the above were extended on the question set of Harbin Institute of Technology.

BERT is a pre-training model that has been very hot in the NLP field in recent years. BERT achieved record-breaking results in many NLP tasks, as shown in the GLUE benchmark [29]. Many other Transformer architectures followed BERT, such as RoBERTa [30], DistillBERT [31], OpenAI Transformer, and XLNet [32], achieving incremental results.

The perspective of some deep learning models is relatively single, and, as introduced at the end of the previous chapter, some neural networks have their own shortcomings and cannot fully learn the characteristic information and semantic information of the question. Therefore, we try to use a variety of deep learning methods to explore the question classification and try to find a more effective and more suitable deep learning framework for question classification.

Since this paper uses a small sample size dataset, the current more popular pre-training models, such as BERT and XLnert, are not applicable, and these pre-training models have high requirements for the lower line of the data volume and do not work well in small sample data. Experiments in Aysu [33] showed that, with a small number of data samples, BERT is not as effective as some neural networks that are simple compared to pre-trained models, such as LSTM models, and LSTM accuracy is better than BERT, and BERT takes more time and is prone to overfitting on small datasets for specific tasks. For this reason, this paper uses Transformer, the core component of BERT with fewer parameters, to perform feature extraction. On the one hand, Transformer, as the core component of BERT, is a powerful feature extractor that has been widely proven in image, audio, and text research fields. The Transformer Encoder is able to efficiently extract features from the input interrogative text and capture semantic and other information using the multi-head attention mechanism.

Finally, add an attention mechanism behind the two-layer network. On the one hand, it can aggregate the feature information of the upper two layers while reducing the output dimension. On the other hand, Attention can further highlight key information, enhance semantic feature capture, and improve the accuracy of question classification. The overall structure of the model is shown in Figure 2:

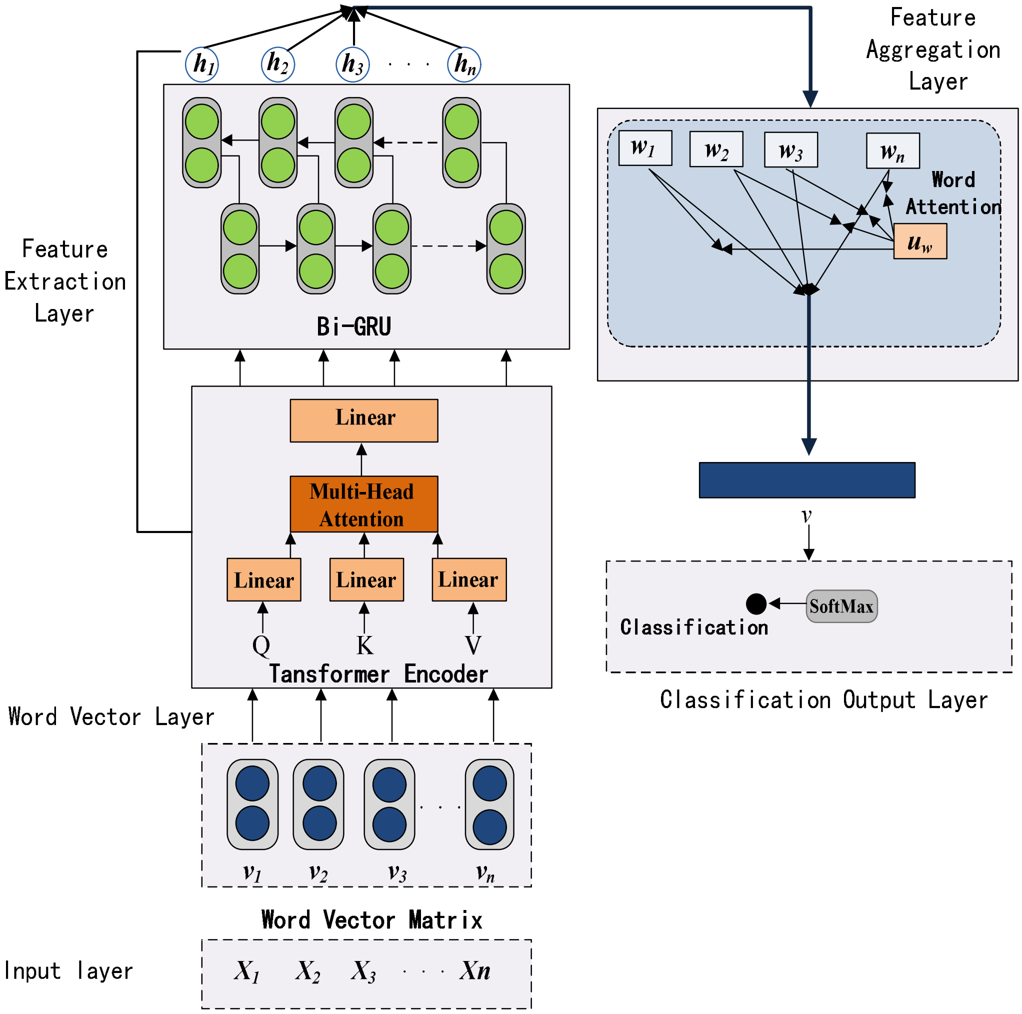

Figure 2. The structure of the TBGA model.

According to the above analysis, the combination of Transformer Encoder and Bi-GRU is used to better enhance the representation of interrogative features and to perform deep feature extraction. On the one hand, Bi-GRU, as an improved version of LSTM, merges and reduces the original three gating units to two units, reducing the parameters while improving efficiency and not degrading performance, and, on the other hand, Bi-GRU can well increase the long-distance dependence length of the Transformer, preventing overfitting on small sample datasets and increasing stability. At the same time, the obvious benefit of the bidirectional GRU is that it can simultaneously extract the features of the hidden layer units from both the front and back directions of the question, which is more effective for obtaining contextual information. The attention mechanism is then added to the back of the two-layer network, which can, on the one hand, aggregate the feature information of the upper two layers while reducing the output dimension, and, on the other hand, the attention can further highlight the key information and enrich the information by combining the answer information of the question and sentence, thus enhancing the semantic feature capture, strengthening the learning of word order semantics and deep features, and thus improving the accuracy of question classification.

As shown in Figure 2, in this paper, all the questions are first represented by word vectors, then input to the embedding layer, and then successively entered into the Transformer and Bi-GRU double-layer networks, giving full play to the respective advantages of Transformer and Bi-GRU and complementing each other, keeping the features of the question and extracting the features, then focusing on and identifying the features of important words and sentences through attention, and aggregating the features of the upper two layers; finally, the final result of the classification is obtained through the Softmax classifier.

As shown in Figure 2, all the question sentences are first represented by word vectors and then input to the embedding layer, which is successively entered into the Transformer and Bi-GRU two-layer networks, giving full play to the advantages of each Transformer and Bi-GRU, as well as complementing each other, keeping the question sentence features and extracting the features; then, the important words and utterances are focused and identified by attention, and the features of the upper two layers are aggregated; finally, the final results of classification are obtained by the Softmax classifier.

We set the question as Q_i:

$$Q_i = \{x_1, x_2, \ldots, x_{n-1}, x_n\} \tag{1}$$

where the i-th word in the question sentence is represented by x_i. If it is a question and answering sentence containing answer information, the word vector of the answer is spliced behind the question word vector. For example, the question Q “Which team has won the most championships in La Liga?” corresponds to the answer A “Manchester United Club.”, then Q and A together represent the question.

What this article uses is the method of initializing word vector randomization, which is constantly updated during the training process. After this step, the question sentence will be input into the next layer of the network in the form of a word vector, as shown in the following formula:

$$S_{1:n} = \{S_1, S_2, \ldots, S_{n-1}, S_n\} \tag{2}$$

3.1. Word Vector Representation Layer

First, jieba is used to segment the Chinese question and stop words are removed. English does not require a word segmentation step, and then the word vector of each word in the question is obtained through training by word2vec [34]. Word2vec was first proposed by Tomas Mikolov in 2013 on the basis of the NNLM model [35]. At the same time, Google also open-sourced an efficient tool for generating word vectors in the same year. The Chinese training corpus of word vectors used in the article is Sogou News [36] and Tencent News [37], and the English training corpus is Google News.

3.2. Transformer Layer

This layer mainly performs feature extraction on the input word vector, using the Transformer [38] encoder belonging to seq2seq; the word embedding of the question sentence and the corresponding position word embedding are added as input. The Transformer model can capture a certain distance dependency while computing in parallel through the multi-head self-attention mechanism, thereby effectively learning the semantic information of the input text. After introducing a series of operations, such as position coding, residual connection, normalization processing, and feedforward layer network connection, the input question sentence is compressed into a fixed-length semantic vector.

The structure of Transformer encoder is shown in the Figure 3. First, positional encoding is introduced, which exists to interpret the order of words in the input sequence and to determine the positional information of the words. In order to enable parallel operations, the attention mechanism drops the order that is important in the temporal sequence, and, if a sequence is disrupted, then the semantics also changes, which is solved by introducing positional encoding. The dimensionality of positional encoding is the same as that of embedding.

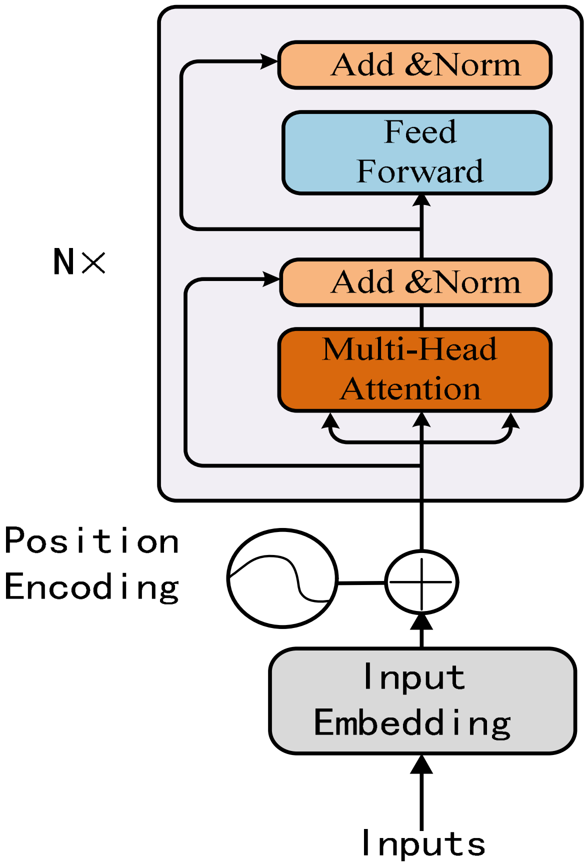

Figure 3. The structure of Transformer Encoder.

The position code calculation formula is as follows:

$$PE(pos, 2i) = \sin(pos/10000^{2i/d_{model}}) \tag{3}$$

$$PE(pos, 2i+1) = \cos\left(pos/10000^{2i/d_{model}}\right) \tag{4}$$

where *pos* is the position of the current word in the sentence, *i* represents each index in the vector value, and dmodel represents the dimension of the word vector. The odd position corresponds to the cosine code, and the even position corresponds to the sine code. PE_{pos+k} at any position can be represented by a linear function of PE_{pos}. The word vector and the position vector are added to obtain the fused word vector, which further enriches the word vector representation of the question sentence.

In the Transformer encoder, each encoder includes a multi-head attention sublayer and a feedforward network sublayer. All sublayers and output dimensions in the model are 200 dimensions.

The core of the encoder is the Multi-head Attention sublayer, which can calculate the degree of association between each word and other words in the question in parallel. In the calculation, each word is independent of the output of the previous word. It can be calculated in parallel. The general form of self-attention layer calculation can be expressed as:

$$\text{Att}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{5}$$

The matrices of the three self-attention layers of Q, K, and V represent Query, Key, and Value, respectively. Through three different linear transformation layers, the input vector is calculated. These three matrices are the calculation results on the input vector, that is, self-attention; d_k is the dimensional size of the word embedding layer, which plays the role of adjusting the inner product size after Q and K transposition to prevent the vector distribution of too large inner product after Softmax is not uniform. Q and K adjust the size of the inner product after transposition through d_k, thus avoiding the problem of too large vector inner product and uneven distribution after Softmax.

$$\text{Multihead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O \tag{6}$$

$$\text{where head}_i = \text{Att}\left(QW_i{}^Q, KW_i{}^K, VW_i{}^V\right) \tag{7}$$

$$sublayer(x) = [Att_1, Att_2, \ldots, Att_n] \tag{8}$$

where $W_i{}^Q$, $W_i{}^K$, $W_i{}^V$, $W^O \in R_{model} \times d_{model}$, respectively, represent the matrix of linear transformation of Q, K, and V. The word vector of d_{model} is mapped into an inverted d_k-dimensional space, and the values of Q, K, and V are equal to the word vector matrix after fusion. i = 1, 2, … , h, h represent the number of heads; each attention head can capture a subspace information in the text sequence and perform h self-attention mechanism calculations and then stitch together through linear transformation. Matrix W^O gets the final self-attention value of multi-head.

The feedforward sublayer consists of two linear transformations with the ReLU activation function in the middle, with the following equation.

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{9}$$

where max is the ReLU activation function, W_1 and W_2 are linear transformations. Considering that the attention mechanism may not fit the complex process sufficiently, the learning ability of the model is enhanced by adding these two layers of linear transformations.

The general form of output can be expressed as:

$$output = LN(x + sublayer(x)) \tag{10}$$

LN is the residual full connection and the normalization layer specification; *sublayer(x)* is a function implemented by the sublayer itself, which are added after the multi-headed attention sublayer and the feed-forward sublayer, respectively. The normalization can improve the convergence speed of the algorithm, and the residual connection can prevent the phenomenon that the current network layer is poorly learned.

The purpose of normalization is to unify the data into a fixed interval in order to avoid the problem of gradient disappearance or gradient explosion when the input data fall into the saturation zone of the activation function later so that activation functions, such as ReLU, can work better. Batch normalization calculates the mean and variance of each layer for each small batch; i.e., the data are normalized to a mean of 0 and a standard deviation of 1 according to the batch dimension. Layer normalization, on the other hand, computes the mean and variance of each sample in each layer independently; i.e., it normalizes the vector data vertically each time. Layer normalization is chosen here because, for models such as transformer that deal with text sequence information, batch normalization becomes very complicated, while layer normalization is possible for a single sample without calculating the global mean-variance, thus improving the convergence speed of the model.

The multi-headed attention sublayer and feedforward sublayer of the encoder are fed as output to the next layer of the neural network through the residual concatenation and normalization (Add & Layer norm, LN) operations described above.

3.3. BiGRU Layer

Gated Recurrent Unit (GRU) [6] is an improved and simplified neural network for Long Short Term Memory (LSTM), which effectively solves the problems of gradient disappearance and gradient explosion in traditional RNNs. For many sentence-level processing tasks, it is very important to consider the context. However, traditional LSTM often only considers timing information and ignores the following information. BiGRU expands the unidirectional network through the second layer of the network; in the input and output, in the process of mapping between sequences, the relevant information about the past and the future of the question is fully utilized so that the information before and after the sentence is captured, and the past and future information is fully considered. Its significant advantages are that the accuracy of question classification is higher, the dependence on word vectors is small, the long-distance dependence is long, the complexity is low, and the response time is relatively fast.

In the Figure 4, x and h are input data and GRU unit output, respectively. c is the reset gate, g is the update gate, and c and g jointly control the calculation and update from the previous hidden state h_{t-1} to the new hidden state ht. Compared with the three gating units of LSTM: input gate, forget gate, and output gate, GRU combines the input gate and the forget gate into an update gate, and the output gate serves as a reset gate to reduce the parameters, and linear self-update does not need to be established In the additional memory state, it is directly linearly accumulated based on the hidden state and controlled by the gate structure, which is more flexible and efficient. The calculation formulas for the update gate and reset gate are as follows:

$$g_t = \phi\left(w_g[h_{t-1}, x_t] + b_g\right) \tag{11}$$

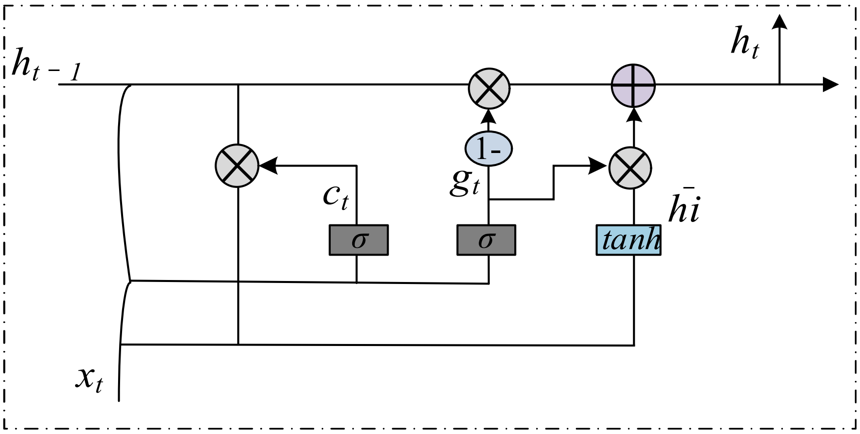

Figure 4. The structure of GRU.

In Formula (11), x_t is the data input by the upper layer, φ is the sigmoid function, w_g is the weights of the update gate the, b_g is bias terms. Last input information h_{t-1} and the current input data g_t are controlled by the update gate at the same time, and the output is a value from 0 to 1; 0 is discarded, 1 is reserved, g_t decide whether to transfer the previous state to the next time state.

$$c_t = \phi(w_d[h_{t-1}, x_t] + b_c) \tag{12}$$

In Formula (12), the reset gate determines the importance of the last time state h_{t-1} to the result h_t; w_d is the weights parameter; b_c is bias terms;

$$\overline{h}_t = \tanh(w_h[c_t h_{t-1}, x_t] + b_h) \tag{13}$$

In Formula (13), the update gate generates new memory information?h_t. Where w_h is the weights of the update reset gate, respectively, and b_h is the bias terms.

The output at the current moment is h_t, that is:

$$h_t = (1 - g_t)h_{t-1} + g_t\overline{h}_t \tag{14}$$

The final result h_i output by this layer is a fusion result of the front and back outputs, as shown in the following formula.

$$h_i = (\overrightarrow{hi} \oplus \overleftarrow{hi}) \tag{15}$$

3.4. Feature Aggregation Layer

In question classification, the contribution of each clause and each word in the question to the classification is different. Some words or clauses are particularly important to the question classification, while the contribution of some words and clauses to the classification is insignificant. In order to capture the effective information in the question, grasp the semantic key points, in this paper, an attention mechanism [39] is added behind BiGRU, and, in order to succeed, we can highlight key semantic feature information, extract effective information, and fully evaluate the contribution of each word to the classification of the entire question so as to retain the most critical information and filter out redundant information and improve the efficiency and performance of question classification. This layer network takes the output of the upper layer network model as the input of the layer model and obtains the vector expression of each sentence in the BiGRU network with the Attention mechanism.

The basic form of Attention can be expressed as:

$$S = tanh(M) \tag{16}$$

$$\alpha = \mathrm{softmax}(w^n S) \tag{17}$$

$$r = M\alpha^n \tag{18}$$

$$q = tanh(r) \tag{19}$$

In Figure 5, where M represents the matrix composed of the word vectors output by the upper BiGRU network, $M \in R^{dn}$, d represents the dimension of the word vector, n is the length of the sentence, w is a training parameter vector, w^n is a transpose, and the dimension w, α, r correspond to d, n, and d, respectively, and q is the final representation of the question sentence used for classification.

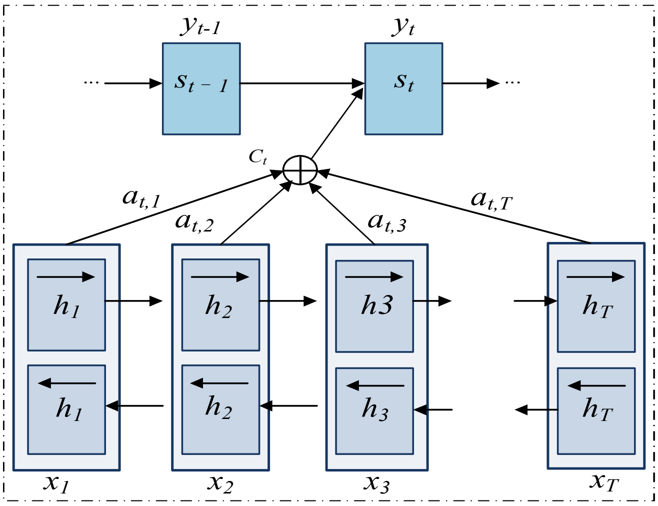

Figure 5. The structure of attention mechanism.

3.5. Softmax Layer

Finally, the category is divided in this layer, and a set of discrete categories Y is used to predict the label of the question Q through the Softmax classifier. The classifier takes the final hidden state q in the upper layer as the input of this layer. Calculated as follows:

$$\hat{p}(y\Big|Q) = \mathrm{softmax}(W^{(Q)}q + b^{(Q)}) \tag{20}$$

$$\hat{y} = argmax\hat{p}(y|Q) \tag{21}$$

The loss function is as follows:

$$J(\theta) = -\frac{1}{m}[\sum_{i=1}^{m}\sum_{j=0}^{1} 1\left\{y^{(i)} = j\right\} \log p(y^{(i)} = j|x^{(i)};\theta)] \tag{22}$$

where j ∈ Rm is onehot encoding, x ∈ Rm represents the estimated probability of each category, and m is the number of target categories, representing a regular term L2, which is used to constrain the weight vector.

3.6. Algorithm Steps

The question and answer data is input into the model and features are extracted for classification, and the final step of outputting the category is as shown in Algorithm 1.

Algorithm 1 Classification Using Deep Neural Network

```
Input: Traning dataset of Quetion $Q$ = ($x_1, x_2, \ldots, x_n$)
    Convert to word vector $V$ = ($v_1, v_2, \ldots, v_n$)
    1. Enter Transormer layer T
            Initial model parameter p
            {Learning rate E
            Momentum $\alpha$
            Batch size m
            $Q$ = $K$ = $V$
            ..}
            Multi-head self-attention calculation $S$ = ($s_1, s_2, \ldots, s_n$)
    Output: $S$ = ($s_1$,$s_2$,..,$s_n$)
    2. Enter BiGRU layer B
            Initial model parameter p
            {..}
            B to Hidden layer computing $H$ = ($h_1, h_2, \ldots, h_n$)
    Output:$H$ = ($h_1, h_2, \ldots, h_n$)
    3. Enter Attention layer A
            Initial model parameter $p$
            {..}
            Attention computing $A$ = ($a_1, a_2, \ldots, a_n$)
    4. Updated model parameter $\theta'$
    repeat
    Input to Softmax
Output $Q$ belongs to category
```

4. Results and Analysis

4.1. Dataset

This experiment mainly uses three datasets. The English dataset uses the TREC dataset and YahooAns dataset, where the former does not contain answer information and the latter contains answer information. However, the Chinese question sentence classification dataset does not have a unified question sentence collection. Most scholars and researchers have expanded the Harbin Institute of Technology Question Collection. Therefore, the question and answer datasets containing answer information from Chinese community question and answer websites, such as "Baidu Know" and "Sogou QA", are collected.

TREC: TREC English question collection [40] belongs to the UIUC question classification standard, and they are all fact questions. There are two versions of the question set: "TREC-6" and "TREC-50". There are six large categories (ABBR, DESC, ENTY, HUM, LOC, NUM) and 50 small categories. The training set is more classic and universal, which can effectively prove the performance of the method.

YAHOO: The YAHOO dataset [41] is a batch of question and answering sentences collected from the English community question and answer platform YAHOO QA. Each question sentence has a corresponding answer, which has been manually verified. There are four categories, namely: information, advice, opinion, and polling.

OQA (Open domain question and answer set): This dataset is a batch of data grabbed from the Chinese community question and answer platforms Baidu Know and Sogou QA. Each question has a corresponding best answer, and some questions in this article are added. The questions were processed by the sentence expansion method. There are a total of seven categories, namely description, character, location, number, time, entity, and unknown, all of which have passed manual labeling and verification. Because the data are noisy, data cleaning was carried out, including the processing of missing and invalid data, and the checking of data consistency.

The structures of the three datasets are shown in Table 3.

Table 3. Dataset structure.

Data Type	Categories	Dataset	Training Set	Test Set	Answer Set
TREC	6	5985	500	500	——
YAHOO	4	1185	885	150	1185
OQA	6	3250	300	300	3250

4.2. The Validity Threats

The datasets created by the experiments in this paper are all manually annotated, and the commonly used classification standards in Chinese are used. Judgment errors occur due to personal, subjective reasons. In order to reduce this threat, we selected the most commonly used open-source datasets, TREC and YAHOO, in the question classification dataset for comparative experiments, hoping that using the repeatedly verified datasets would more effectively avoid the impact of subjective annotation. In addition, because the paper is aimed at the deep learning question classification model of python language, the templates set are for some specific cases of this method, so the results may not be generalized to other python programs or other languages.

4.3. Experimental Setup

The Chinese dataset uses the Jieba toolkit to segment the experimental data, and the word vectors are obtained from the training text of the CBOW model of Word2Vec. The word vectors used on the English datasets TREC and YahooAns are Google News Corpus pre-trained by word2vec containing 100 billion vocabulary, and the dimension is set to 200 dimensions. The word vectors used on the Chinese dataset OQA are the Sogou news and the The Sogou news and Tencent news corpus. The Sogou news corpus size is 711 megabytes, the word vector dimension is 200 dimensions, the Tencent news corpus size is 800 megabytes, and the training word vector dimension is also 200 dimensions.

On all three datasets, the parameters used in the algorithm model are the same. In order to compare with the previous work completed by Kim et al., the experiments in this article use some basic parameters. To prevent over-parameterization and over-fitting during the training process, and to avoid the occasional bad local minimum phenomenon, set the Dropout parameter to 0.5, the value of l2consraint(s) is 3, the learning rate is 1×10^{-3}, the L2 regular term is 1×10^{-2}, the number of multi-head attention heads is 2, the batch size is 32, and the maximum sequence length is 200. In addition, through the optimizer, the learning rate exponentially decays dynamically; each epoch: learning rate = gamma ×learning rate so as to ensure the efficiency while training to achieve the best results. In addition, if the validation set loss exceeds 1000 batches and does not decrease, the training ends.

4.4. Evaluation Indicators

The evaluation indicators of the experiment are Accuracy (Acc), Precision (Prec), Recall (Rec), and F1 value. The specific calculation formula is as follows:

$$\mathrm{Acc} = \frac{\mathrm{AccNum}}{\mathrm{Num}_{\mathrm{total}}} \tag{23}$$

where AccNum represents the number of question samples correctly classified, and $\mathrm{Num}_{\mathrm{total}}$ represents the total number of questions in the test set. The ratio of the number of samples in which the category predicted by the classifier is consistent with the actual category and the number of questions in the test set is used as an important measure.

$$\mathrm{Precision}(a) = \frac{\mathrm{PrecNum}(a)}{\mathrm{Num}_{\mathrm{total}(a')}} \tag{24}$$

where Recall (a) represents the recall rate of category a, RecNum (a) represents the number of samples in the test set that are predicted to be category a and are actually category a, and Numtotal (a) the true number of the a dataset in the test set. The recall rate of category a is the ratio of the correct number of datasets in category a to the true number of category a datasets, and it is designed to detect the recall rate of the category.

$$F1 = 2 \times \frac{\mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{25}$$

4.5. Hardware and Software Environment

The software and hardware environment used in the experiments in this paper are shown in Table 4.

Table 4. Hardware and software environment.

No.	Type	Description
1	Operation System	Ubuntu 18.04.3 LTS
2	Experiment Environment	Pytorch 1.3
3	CPU	Intel Core i7-8700K
4	RAM	DDR432GB
5	CUDA	10.0
6	GPU:	NVIDA GeForce GTX 1080Ti
7	Disk	2TB NVMe SSD

4.6. Experimental Results

As shown in Figure 6, first of all, we use the Text-CNN model on the YAHOO dataset to compare 600 single question sentences with questions containing answer information. As shown in the figure, the results show that adding an answer and not adding an answer is two to three percentage points higher. It shows that the classification effect is better after adding the answer information to enrich the question feature information. It also shows that the Chinese question and answer dataset constructed in this article has a certain meaning. On the one hand, it can improve the classification accuracy and, on the other hand, it can provide convenience for subsequent answer extraction tasks.

Figures 7–9 are the comparison results of experiments performed by different models on the validation sets of the three data sets.

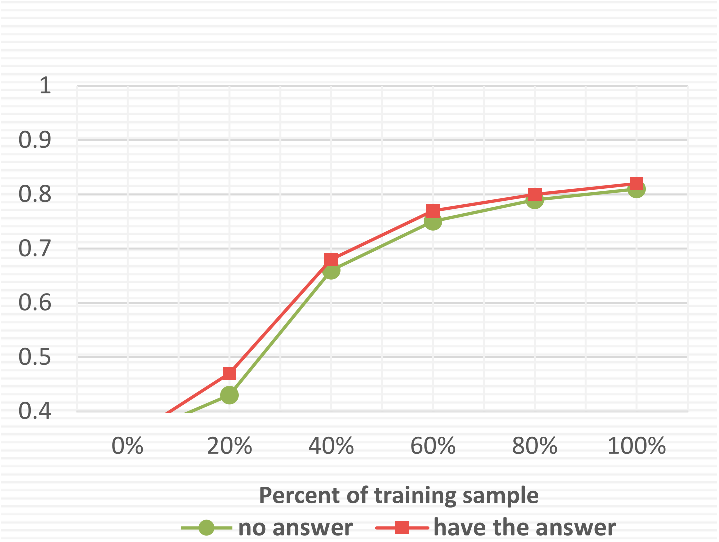

Figure 6. Comparison of the YAHOO dataset.

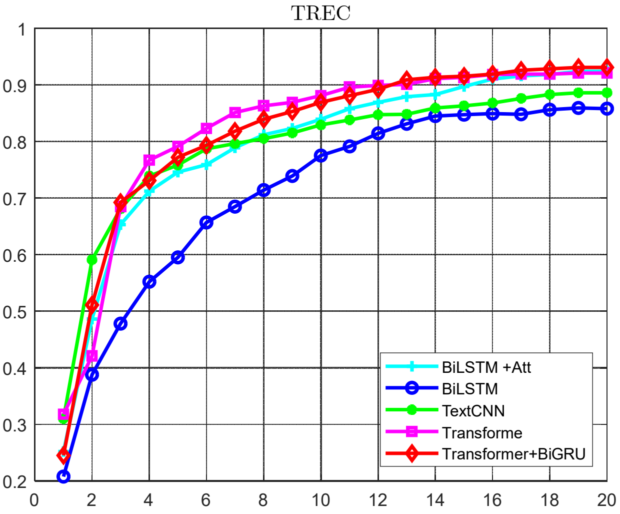

Figure 7. Comparison of accuracy rates on the TREC verification set.

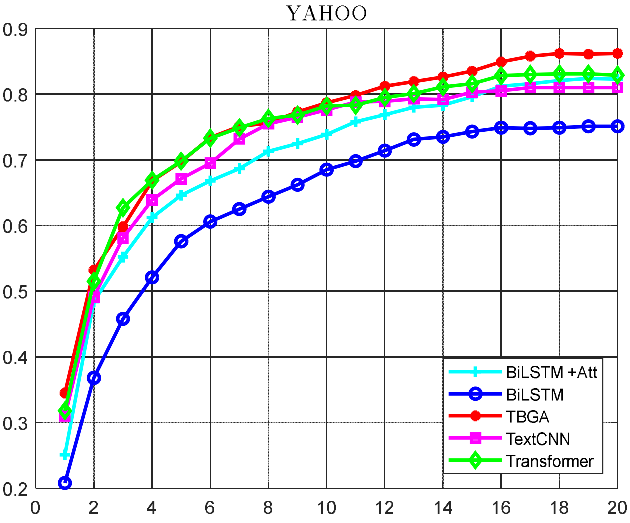

Figure 8. Comparison of accuracy rates on the YAHOO verification set.

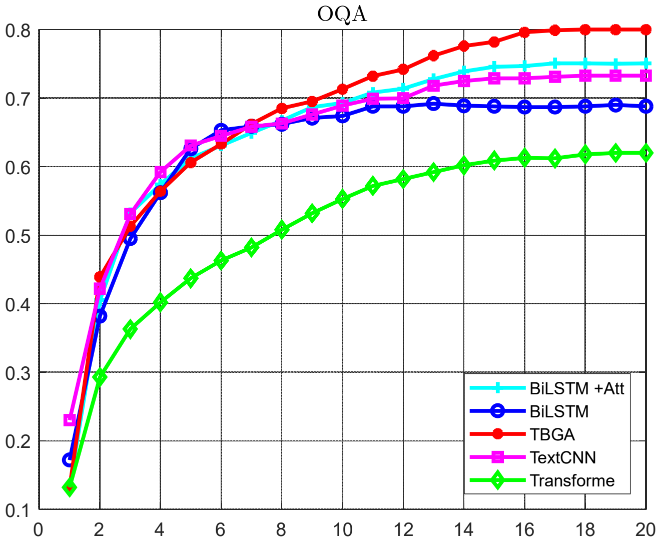

Figure 9. Comparison of accuracy rates on the OQA verification set.

As shown in Table 5, it can be seen from the results that LSTM has the worst effect on the TREC dataset, and it is basically the same as the Transformer after the attention mechanism is added. In general, the classification effect of TREC is better than that of YahooAns and OQA, mainly because the TREC dataset is composed of question sentences and has no answer information. Compared with the latter, it is relatively simple. The classification effect of OQA is the lowest due to community questions and answers, such as Baidu Know and Sogou QA. The questions on the platform are more complex, with many abbreviated and colloquial uses of vocabulary and sentences, and they also contain complex answer information.

Table 5. Experimental results.

Model	TREC				YAHOO				OQA			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
CNNs	89.02	89.43	89.03	89.23	81.26	81.71	82.03	81.86	75.33	76.81	75.44	76.11
BiLSTM	85.03	85.75	85.00	85.37	75.18	75.42	75.63	75.52	69.01	68.22	69.02	68.61
BiLSTM+Att	92.63	92.61	92.60	92.60	82.43	82.37	82.45	82.40	75.21	75.43	75.03	75.23
Transfomer	92.21	92.22	92.20	92.20	83.17	83.36	83.30	83.32	62.76	60.48	61.77	61.12
Trans + BiGRU	93.22	92.88	93.20	93.03	-	-	-	-	-	-	-	-
TBGA	-	-	-	-	86.12	86.23	86.20	86.21	80.08	82.62	80.12	81.35

The model TBGA achieves ideal results on questions containing answer information. In addition, the addition of an attention mechanism is indeed effective, which can capture the text characteristics to be represented. Especially on the OQA dataset, compared with the traditional CNN method, the accuracy has been improved by 4.75%. Transformer has the worst effect. The main reason is that the amount of data offline is relatively high, and the effect of longer question and answer sentences containing answers is not good. This is also caused by insufficient long-distance dependence. LSTM is not effective, insufficient feature extraction is the main reason. Transformer is used as a strong feature extractor, and long-distance dependence is supplemented by improved Bi-GRU. At the same time, more features are extracted through the hidden layer, and, finally, the attention mechanism is used to highlight important question information and feature fusion so as to give play to their respective advantages and complement each other to improve the classification effect. In summary, on the English question dataset and on the Chinese question and answer dataset, the method proposed in this paper has significant advantages. The improvement of the effect on different datasets also shows the effectiveness and comparison of the method, with strong generalization ability.

In summary, the method proposed in this paper has achieved significant improvement in both the English and Chinese datasets, which not only verifies the effectiveness of the method but also reflects the generalization ability of the method from the side.

In addition, as shown in Figure 10, by comparing the F1 values of each category on the TREC dataset for commonly used text classification models: convolutional network and RNN network plus attention mechanism and multi-head attention network, it can be found that the abbreviated category (ABB) question has the most effect. A good model is Transformer. On the statement type (DES) question, the CNN model is relatively effective. The model that works best in ENT (entity class) is also the CNN network. For the character category (HUM) question, the Transformer model has the best effect. For the place name category (LOC) question, the RNN plus attention model has the best effect. On the number category (NUM) question, the Transformer model has the best effect. It is found through experiments that different networks have different performances in different categories. This has a certain enlightening effect on the classification of problems in specific domain categories with corresponding models.

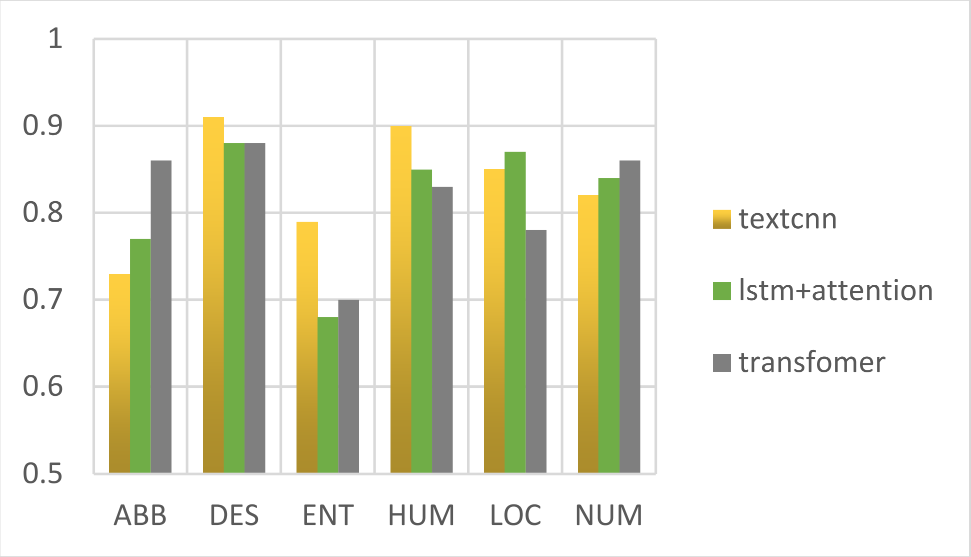

Figure 10. F1 values of each category on the TREC.

5. Conclusions and Future Work

In recent years, the rapid development of the Internet has made more intelligent information technology imminent. As a hot research direction in recent years, question and answer systems can satisfy users' needs for accurate information acquisition with the background of increasing information. Demand and the combination of deep learning and question answering system tasks is the current trend. Question classification, as the primary link of the question and answer system, can see the range of candidate answers, which plays a role in the direction of follow-up research. Deep learning technology is undoubtedly a new research hotspot in the branch of machine learning, and it has become a powerful force to promote the rapid development of machine learning. The development of deep learning provides strong technical support for the research of question classification and also plays a guiding role in the development of future question answering systems. Therefore, the focus of this paper was the combination of deep learning technology and question classification tasks.

We analyzed the current situation of question classification and the successful application of machine learning and deep learning in this research field and analyzed and compared different machine learning and deep learning question classification models. We attempted to use a variety of deep learning methods to apply and research question classification. This method is mainly based on the question classification method of deep neural network-Transformer and Bi-GRU and adds the attention mechanism layer, which is characterized by: on the one hand, it uses the combination of Transformer and Bi-GRU to simultaneously obtain the essential characteristics and timing characteristics of the question so as to maximize the extraction of the effective information in the question to be classified. On the other hand, an attention mechanism is added to the model to make full use of the answer information of the question sentence to enhance the expression of the question sentence and capture the effective information in the question sentence, grasp the semantic key points in order to retain the most critical information, and filter out the redundant information. The experimental results show that the question classification method proposed in this paper has a higher accuracy rate than traditional machine learning methods and a single neural network structure on both the Chinese and English datasets.

In the next work, we will consider how to better optimize the pre-processing part of the interrogative sentences so as to reduce the noisy data. In addition, for the problem of the insufficient Chinese interrogative corpus, in addition to collecting and labeling more corpora, a combination of traditional feature extraction methods and deep learning can be used to further improve the classification accuracy on top of the existing corpus, and, at the same time, different neural network models can be attempted to be improved and fused appropriately to obtain better information of interrogative sentences so as to further improve the classification accuracy of interrogative sentences.

Author Contributions: Investigation, D.H.; methodology, D.H. and T.T.; experiment, D.H.; supervision, A.H. and T.T.; validation, D.H., A.H. and T.T.; writing—original draft, D.H.; writing—review and editing, D.H., A.H. and T.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Natural Science Foundation of Xinjiang, China (2021D01C076).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Data Age 2025 of IDC. Available online: https://www.seagate.com/cn/zh/our-story/data-age-2025/ (accessed on 2 March 2022).
2. Green, B.F.; Wolf, A.K.; Chomsky, C.; Laughery, K. Baseball: An Automatic Question Answerer. In Proceedings of the Western Joint IRE-AIEE-ACM Computer Conference, New York, NY, USA, 9–11 May 1961; pp. 219–224.
3. Woods, W.A. Lunar rocks in natural English: Explorations in natural language question answering. *Linguist. Struct. Processing* **1977**, *5*, 521–569.
4. Wilensky, R.; Chin, D.N.; Luria, M.; Martin, J.; Mayfield, J.; Wu, D. The Berkeley UNIX consultant project. *Comput. Linguist.* **1988**, *14*, 35–84.
5. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to Sequence Learning with Neural Networks. *Comput. Lang.* **2014**, *2*, 3104–3112.
6. Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *Comput. Sci.* **2014**, *19*, 25–27.
7. Bahdanau, D.; Cho, K.; Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. *Comput. Sci.* **2014**, *20*, 61–63.
8. Sarker, S.; Monisha, S.T.A.; Nahid, M.M.H. Classification of Bengali Questions towards a Factoid Question Answering System. In Proceedings of the International Conference on Advances in Science Engineering and Robotics Technology, Dhaka, Bangladesh, 3–5 May 2019; pp. 1–5.
9. Moldovan, D.; Pasca, M.; Harabagiu, S.; Surdeanu, M. Performance issues and error analysis in an open-domain question answering system. *ACM Trans. Inf. Syst.* **2003**, *21*, 133–154. [CrossRef]

10. Faiz, A.; Manna, R.; Laskar, S.R.; Pakray, P.; Das, D.; Bandyopadhyay, S.; Gelbukh, A. Question Classification and Answer Extraction for Developing a Cooking QA System. *Comput. Sist.* **2020**, *24*, 24.
11. Jia, K.; Fan, X.Z.; Xu, J.Z. Chinese qusetion classification based on KNN. *Microellectronics Comput.* **2008**, 162–164.
12. Hacioglu, K.; Ward, W. Question Classification with Support Vector Machines and Error Correcting Codes. In Proceedings of the Companion Volume of the Proceedings of HLT-NAACL 2003-Short Papers, Edmonton, AB, Canada, 27 May 2003; pp. 28–30.
13. Silva, J.; Luísa, C.; Mendes, A.C.; Wichert, A. From symbolic to sub-symbolic information in question classification. *Artif. Intell. Rev.* **2011**, *35*, 137–154. [CrossRef]
14. Wen, X.; Zhang, Y.; Liu, T.; Ma, J.-S. Syntactic structure parsing based chinese question classification. *J. Chin. Inf. Processing* **2006**, *20*, 35–41.
15. Zhang, Y.; Liu, T. Research progress of open domain question answering technology. *J. Electron.* **2009**, *37*, 1058–1067.
16. Hovy, E.; Gerber, L.; Hermjakob, U.; Lin, C.-Y.; Ravichandran, D. Toward Semantics-Based Answer Pinpointing. In Proceedings of the Second International Conference on Human Language Technology Research, San Diego, CA, USA, 18–22 March 2001; pp. 1–7.
17. Zhang, D.; Lee, W.S. Question Classification Using Support Vector Machines. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Toronto, ON, Canada, 28 July–1 August 2003; pp. 26–32.
18. Sundblad, H.A. Re-Examination of Question Classification. In Proceedings of the 16th Nordic Conference of Computational Linguistics, Tartu, Estonia, 24–26 May 2007; pp. 394–397.
19. Li, X.; Huang, X.J.; Wu, L.D. Combination classifier based on error- driven algorithm and its application in problem classification. *J. Comput. Res. Dev.* **2008**, *45*, 535–541.
20. Huang, Z.; Thint, M.; Qin, Z. Question Classification Using Head Words and Their Hypernyms. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Honolulu, HI, USA, 25–27 October 2008; pp. 927–936.
21. Peng, X.Y.; Zhong, D. Survey of Cross-Lingual Word Embedding. *J. Chin. Inf. Processing* **2020**, *34*, 1–15, 26.
22. Kim, Y. Convolutional Neural Networks for Sentence Classification. In Proceedings of the Empirical Methods in Natural Language Processing, Doha, Qatar, 25–29 October 2014; pp. 1746–1751.
23. Komninos, A.; Manandhar, S. Dependency Based Embeddings for Sentence Classification Tasks. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, 26–31 July 2015; pp. 174–179.
24. Xu, J.; Zhang, D.; Li, S.; Wang, H. Research on question classification via bilingual information. *J. Chin. Inf. Processing* **2017**, *31*, 171–177.
25. Shi, M.F.; Yang, Y.; He, L. Community Q&A question classification method based on Bi-LSTM and CNN and including attention mechanis. *Comput. Syst. Appl.* **2018**, *27*, 157–162.
26. Yu, Z.T.; Fan, X.Z.; Guo, J.Y. Classification of Chinese Questions Based on Support Vector Machine. *J. South China Univ. Technol. Nat. Sci. Ed.* **2005**, *33*, 25–27.
27. Tian, W.D.; Gao, Y.Y.; Zu, Y.L. Question classification based on self-learning rules and modified Bayes. *Appl. Res. Comput.* **2010**, *27*, 75–77.
28. Li, R.; Song, X.X.; Wang, W.J. Chinese question classification based on Chinese Frame Net. *Comput. Eng. Appl.* **2009**, *45*, 111–115.
29. Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S.R. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. *arXiv* **2018**, arXiv:1804.07461.
30. Liu, Y.H.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**, arXiv:1907.11692.
31. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter. *arXiv* **2019**, arXiv:1910.01108.
32. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.; Le, Q.V. Xlnet: Generalized autoregressive pretraining for language understanding. In Proceedings of the 33th Conference and Workshop on Neural Information Processing Systems, Vancouver, BC, Canada, 10–12 December 2019; pp. 5754–5764.
33. Ezen, C.A. A Comparison of LSTM and BERT for Small Corpus. *arXiv* **2020**, arXiv:200 9.05451.
34. Khandelwal, U.; He, H.; Qi, P.; Jurafsky, D. Sharp Nearby, Fuzzy Far Away: How Neural Language Models Use Context. *arXiv* **2018**, arXiv:1805.04623.
35. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. *arXiv* **2013**, arXiv:1301.3781.
36. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and Their Compositionality. *arXiv* **2015**, arXiv:1508.06669.
37. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In Proceedings of the Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
38. Chung, J.; Gulcehre, C.; Cho, K.H.; Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv* **2014**, arXiv:1412.3555.
39. González, J.; Gómez, J. TREC: Experiment and evaluation in information retrieval. *J. Am. Soc. Inf. Sci. Technol.* **2007**, *58*, 910–911.

40. Choi, E.; Kitzie, V.; Shah, C. A Machine Learning-Based Approach to Predicting Success of Questions on Social Question Answering. In Proceedings of the iConference 2013, Fort Worth, TX, USA, 12–15 February 2013.
41. Adamic, L.A.; Zhang, J.; Bakshy, E.; Ackerman, M.S. Knowledge Sharing and Yahoo Answers: Everyone Knows Something. In Proceedings of the International Conference on World Wide Web ACM, Beijing, China, 21–25 April 2008; pp. 665–674.