Named Entity Aware Transfer Learning for Biomedical Factoid Question Answering

Keqin Peng, Chuantao Yin, Wenge Rong, Chenghua Lin, Deyu Zhou, and Zhang Xiong

Abstract—Biomedical factoid question answering is an important task in biomedical question answering applications. It has attracted much attention because of its reliability. In question answering systems, better representation of words is of great importance, and proper word embedding can significantly improve the performance of the system. With the success of pretrained models in general natural language processing tasks, pretrained models have been widely used in biomedical areas, and many pretrained model-based approaches have been proven effective in biomedical question-answering tasks. In addition to proper word embedding, name entities also provide important information for biomedical question answering. Inspired by the concept of transfer learning, in this study, we developed a mechanism to fine-tune BioBERT with a named entity dataset to improve the question answering performance. Furthermore, we applied BiLSTM to encode the question text to obtain sentence-level information. To better combine the question level and token level information, we use bagging to further improve the overall performance. The proposed framework was evaluated on BioASQ 6b and 7b datasets, and the results have shown that our proposed framework can outperform all baselines.

Index Terms—Biomedical factoid question answering, Transfer learning, Name Entity, Question representation, Ensemble

1 Introduction

With the development of advanced biomedical techniques, biomedical scientific literature has exploded rapidly, making it challenging for researchers to explore large amounts of information. The conventional solution is to use information retrieval (IR) techniques to obtain information from datasets, with researchers hoping to obtain answers directly, instead of a list of articles. Therefore, the question answering (QA) system has gained widespread attention in the community [1]. Many organizations have also started to conduct competitions to promote the development of QA systems in the biomedical domain, and one of the most important competitions is BioASQ [2]. The biomedical QA system usually contains four types of questions: 1) "yes" or "no" questions, 2) factoid questions, 3) list questions, and 4) summary questions [2]. Among them, factoid questions have attracted much attention as QA systems are expected to provide reliable answers.

QA is one of the most fundamental applications of natural language processing. It aims to provide useful and related information for a given question in a natural language. In contrast to IR systems, QA systems use sentences to address information needs [3]. Similar to other natural language processing tasks, better representation of words in QA texts is also essential, as it can help achieve good performance even with a simple neural network. To represent a word, researchers initially used one-hot encoding to encode each word [4]; however, it is not effective because of the sparsity and high dimension of the representation vector. To overcome these problems, Word2Vec, which can learn the representation of each word from a large dataset, was later proposed [5], [6]. Word2Vec has made significant improvements in various natural language processing tasks. However, the Word2Vec model also has some disadvantages, such as its inability to solve the ambiguity of words. Therefore, to learn a good representation that can consider the context information for each word, pretrained models began to emerge [7], [8]. These models have achieved state-of-the-art (SOTA) performances in various natural language processing tasks, and many advanced approaches in QA applications employ pretrained models [9].

In the biomedical field, earlier methods used feature engineering mechanisms to obtain several linguistic and semantic features from the tokens and concepts [10], or used context-independent embedding to represent each word. Inspired by the success of pretrained models in other applications, pretrained models were also adopted in biomedical tasks. However, owing to the different word distributions between general and biomedical texts, directly applying pretrained models to the biomedical domain could not achieve satisfactory performance [11]. Therefore, Lee et al. [11] proposed BioBERT, a pretrained model trained on PubMed articles, that outperforms previous approaches in three biomedical tasks, i.e., named entity recognition (NER), relation extraction, and question answering. Furthermore, based on the BioBERT model, some researchers tried to integrate more external information to achieve better performance [12], [13].

Although BERT/BioBERT-based models have achieved good performance, they still face some challenges. The

- *K. Peng and C. Yin are with State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China, and also with Sino-French Engineer School, Beihang University, Beijing 100191, China. E-mail: {keqin.peng, chuantao.yin}@buaa.edu.cn.*
- *W. Rong and Z. Xiong are with State Key Laboratory of Software Development Environment, Beihang University, Beijing, China, and also School of Computer Science and Engineering, Beihang University, Beijing 100191, China. E-mail: {w.rong, xiongz}@buaa.edu.cn.*
- *C. Lin is with Department of Computer Science, University of Sheffield, Sheffield S1 4DP, United Kingdom. E-mail: c.lin@sheffield.ac.uk.*
- *D. Zhou is with School of Computer Science and Engineering, Southeast University, Nanjing 210096, China. E-mail: d.zhou@seu.edu.cn.*

Manuscript received xxxx, xxxx; revised xxxx, xxxx.
(Corresponding Author: Chuantao Yin.)

BERT/BioBERT is an autoencoder language model, and the embeddings are based on tokens. They assume that the predicted tokens are independent of each other, given the unmasked tokens [14], which makes it less capable of effectively learning the information of phrases, such as named entities, while previous research has proven that the named entity has a positive effect in QA [15]. Compared to the general QA task, biomedical text contain a larger number of abbreviations and domain proper nouns [16], which makes it more important and more difficult to learn named entity information. For example, in the biomedical factoid question, the answer is usually a named entity, e.g., *Name synonym of Acrokeratosis paraneoplastica*; *Orteronel was developed for treatment of which cancer?*

To capture named entity information, several researchers have attempted to merge named entity information into word embeddings. For example, Lamurias et al. introduced NER features to enrich the original QA text [17]. However, because the named entity in the biomedical domain is usually complicated [18], it is difficult to cover all biomedical name entities by directly merging NER information into word vectors. It is therefore interesting to explore alternative mechanisms to utilize the NER information for biomedical QA applications.

Transfer learning is a mechanism that applies knowledge learned in a previous task to another task because it can retain some information of the previous task. The fine-tuning model that uses external and auxiliary data is one of the widely used transfer learning approaches [8], [13]. If the previous task is related to the current work, it usually can improve the performance of the current work [19] [20]. In the biomedical field, the use of transfer learning has attracted much attention. For biomedical QA tasks, the amount of data for training is typically small, which makes transfer learning a good choice for integrating more knowledge from outside. For example, Yoon et al. [12] proposed the fine-tuning of BioBERT in the SQuAD, a popular and large-scale reading comprehension dataset, to learn general knowledge and achieve SOTA performance. Similarly, Jeong et al. [13] proposed an approach that first fine-tuned the BioBERT model in a neural language inference (NLI) task and then in SQuAD for better performance. Inspired by these approaches, in this study, we propose a transfer learning mechanism to merge the information of named entity into the QA model, and we also use the SQuAD dataset to learn the general knowledge that has been proven effective [12].

Another challenge of the BERT/BioBERT architecture is that no independent sentence embeddings are computed [21]. Although some researchers have tried to use the average of outputs of BERT or the special token **CLS** to represent the overall information [22] [23] [24], these methods cannot learn good sentence embeddings [21]. At the same time, sentence information is very important because it usually contains semantic and linguistic properties [25] [26] and may contain some different and valuable information, compared to word embeddings. In QA, the use of word embeddings may be influenced by noisy information under certain circumstances, such as context with lots of single noisy words which are similar to those words in the question but unrelated to the question answering [27], whereas using the sentence information can solve this problem. In particular, the question representation can also strengthen the system's understanding of the problem. BiLSTM [28] has been proven to be effective in properly encoding sentences [29] in an NLI task; hence, in this study, we also try to use BiLSTM to learn a good representation of the overall information of questions to strengthen the question information.

Therefore, in this study, we have token-level QA text information using BioBERT as well as sentence-level QA text information using BiLSTM. To combine the sentence-level and token-level information, we applied the ensemble method [30], which has been effective in improving the performance of simple models [31]. We shared the BioBERT parameters in the two models to learn both the information and reduce the total parameters. Furthermore, ensemble methods can alleviate the problem of unbalanced data [32]. In most QA datasets, the problem of unbalanced data distribution typically exists, which causes the model to obtain very different results in different datasets. In the biomedical domain, the data distribution is also imbalanced. To solve this problem, in the biomedical classification tasks, previous studies usually used random resampling techniques [33], [34], and some researchers used ensemble methods [32]. Hence, in this study, inspired by the classification problem, we applied the bagging mechanism to the proposed framework. We trained the token and sentence-level models simultaneously and then obtained the five most likely answers from each of the models. Afterwards, we used the bagging method to rank the ten answers and get the top-five answers according to their probabilities.

The contributions of this study are as follows: 1) we show that fine-tuning on an NER dataset is effective in answering biomedical factoid questions; 2) we demonstrate that considering overall question information can improve the performance of biomedical factoid QA; and 3) we applied an ensemble mechanism to improve the performance of biomedical factoid QA. The proposed framework was trained and evaluated on the major competition task, BioASQ 6b and 7b, and the results have shown that our proposed framework can outperform all baselines.

2 Related Work

The biomedical QA system is a field of extensive research in the biomedical domain because it can directly produce answers [35] and help researchers quickly find the answer they need rather than browse through a list of articles like in information retrieval (IR) systems. Many organizations hold competitions to promote the development of QA systems, and one of the most popular competitions is the BioASQ[1], which organized public challenges for biomedical semantic indexing and QA. The competition is held once every year and was first held in 2013. Every year, it publishes a dataset, and BioASQ 6b and BioASQ 7b are the datasets released in the 6th and 7th competitions, respectively. The BioASQ competition consists of two tasks: large-scale online biomedical semantic indexing and biomedical semantic QA. Among them, in the QA phase of the second task, there are four types of questions, i.e., 1) "yes" or "no" questions, 2) factoid questions, 3) list questions, and 4) summary questions [2].

1. http://bioasq.org/

The BioASQ competition has become one of the most influential competitions in the biomedical domain, and many researchers are involved in QA research in the biomedical field, especially for factoid questions. For example, Yang et al. [36] proposed the use of supervised models to predict the answer and question type, and then calculated the score of each answer to find the golden answer. However, limited by the amount of data, it used a feature engineering method to extract features from the concepts and named entities, which could not learn a good representation. Later, the proposed AUTH model [37] focused on the process of answer processing. It used word embedding and external resources to represent an answer and obtain its score. This proves that using external information can alleviate the problem of the lack of large datasets, while the feature engineering method can be used to obtain the answers without fully considering the information in the question. The "LabZhu" [38] approach used the knowledge graph method to solve the problem, but in biomedical domain, the knowledge graph is difficult to build. Recently, with the emergence of pretrained models, Google proposed the system "google-gold-input" [39], which used the BERT model to train the BioASQ datasets. However, it did not consider the different data distributions between the general and biomedical domains. To alleviate the influence of different data distributions, Lee et al. [11] proposed BioBERT, which is a pretrained model trained on PubMed articles. The model achieved a remarkable improvement in QA systems. However, owing to the small amount of data available in the biomedical domain, utilizing more useful external information is still challenging. Therefore, Yoon et al. [12] proposed to fine-tune the BioBERT in SQuAD to learn general knowledge. Similarly, Jeong et al. [13] proposed to first fine-tune the BioBERT in an NLI task and then fine-tune the SQuAD to learn external information.

In natural language processing tasks, sentence representation is also an important challenge, and many researchers have proposed diverse solutions to obtain sentence embedding. Because a sentence is composed of a series of tokens, many researchers have used recurrent neural networks (RNNs) to learn sentence information. However, it is difficult to capture the long-term dependencies in a simple RNN architecture because of the vanishing gradient and gradient explosion problems [40]. To overcome this problem, Hochreiter et al. proposed the long short-term memory (LSTM) [41], and Cho et al. proposed a gated recurrent unit [42]. These models focus on one-way information; hence, Schuster et al. further proposed the bidirectional LSTM (BiLSTM) [28] to learn bidirectional information. BiLSTM is often used to learn question embeddings or answer embeddings in QA applications. For example, Tan et al. [43] used BiLSTM to obtain the embeddings of questions and answers for factoid answer selection. Similarly, Li et al. used BiLSTM to rank the answer [44] and achieved better performance, which also demonstrates that question embeddings are useful in QA systems. For biomedical applications, Wiese et al. proposed the use of BiLSTM to learn question embedding for interactions with the answer [45].

In this study, we use the bagging method, a kind of ensemble methods, to combine question-level and token-level information. The ensemble methods include two types of methods: bagging [46] and boosting [47] and they are widely used in solving biomedical classification problems. For biomedical applications, Huang et al. [48] used a bagging classification tree to classify G-protein coupled receptors and achieved good performance. Similarly, Hayder et al. [49] used an adaptive bagging method on a biomedical data stream. In the QA system, the ensemble method is also used to solve sub-problems, such as using the bagging method to learn the relevant label information, to improve the performance of the QA system [50].

3 METHODOLOGY

In this section, we explain the details of the proposed framework for the biomedical factoid QA task. The overall architecture is shown in Fig. 1. First, we provide a definition of the problem and an overview of the framework. Subsequently, we elaborate on the overall process.

3.1 Problem Definition and Architecture Overview

The biomedical factoid question challenge is an extractive QA task, i.e., given a context passage $C = \{c_1, c_2, ..., c_m\}$ and a question $Q = \{q_1, q_2, ..., q_n\}$, there is only one answer $A = \{c_s, c_{s+1}, ..., c_e\}$ in the context passage. In the previous definitions, c_i represents the i-th token in the context, q_j represents the j-th token in the question, m and n are the length of context passage and question, respectively, and s and e are the starting and ending positions of the answer in the context, respectively. The goal of the system is to determine the starting position s and ending position e of the answer in the context passage.

As shown in Fig. 1, our proposed framework is mainly divided into three parts: 1) transfer learning by fine-tuning the BioBERT in the datasets NER and SQuAD, 2) learning the question representation, and 3) applying the ensemble method to combine the two models. First, we adopt a transfer learning mechanism to learn the information of the named entities and general knowledge by fine-tuning the model in the NER dataset and SQuAD, step by step. We encode the question and context together to obtain the embedding of each token in the question and context, then we pass all the embeddings to BioBERT and fine-tune the parameters of BioBERT by the NER task and then SQuAD. Subsequently, for the factoid QA task, we first obtain all the embeddings O_0 of all tokens in the question and context and put them into BioBERT to learn a new representation for each token. Thus, we obtain the new token embeddings O_1, then we put all the new token embeddings of the question into BiLSTM to learn an overall representation for the question and then concatenate the question embeddings with the context embeddings to form new embeddings O_2. Meanwhile, we retain the output embeddings O_1 and pass O_1 and O_2 to two different neural network layers and use the $softmax$ function to obtain two vectors P_1^{pred} and P_2^{pred}, which represent the probabilities of all the tokens as the start position and the end position in the two models. During the prediction step, we obtain the five most likely answers by combining the probability of the start and end positions from each model separately, and finally we use the ensemble method to rank the ten answers and choose

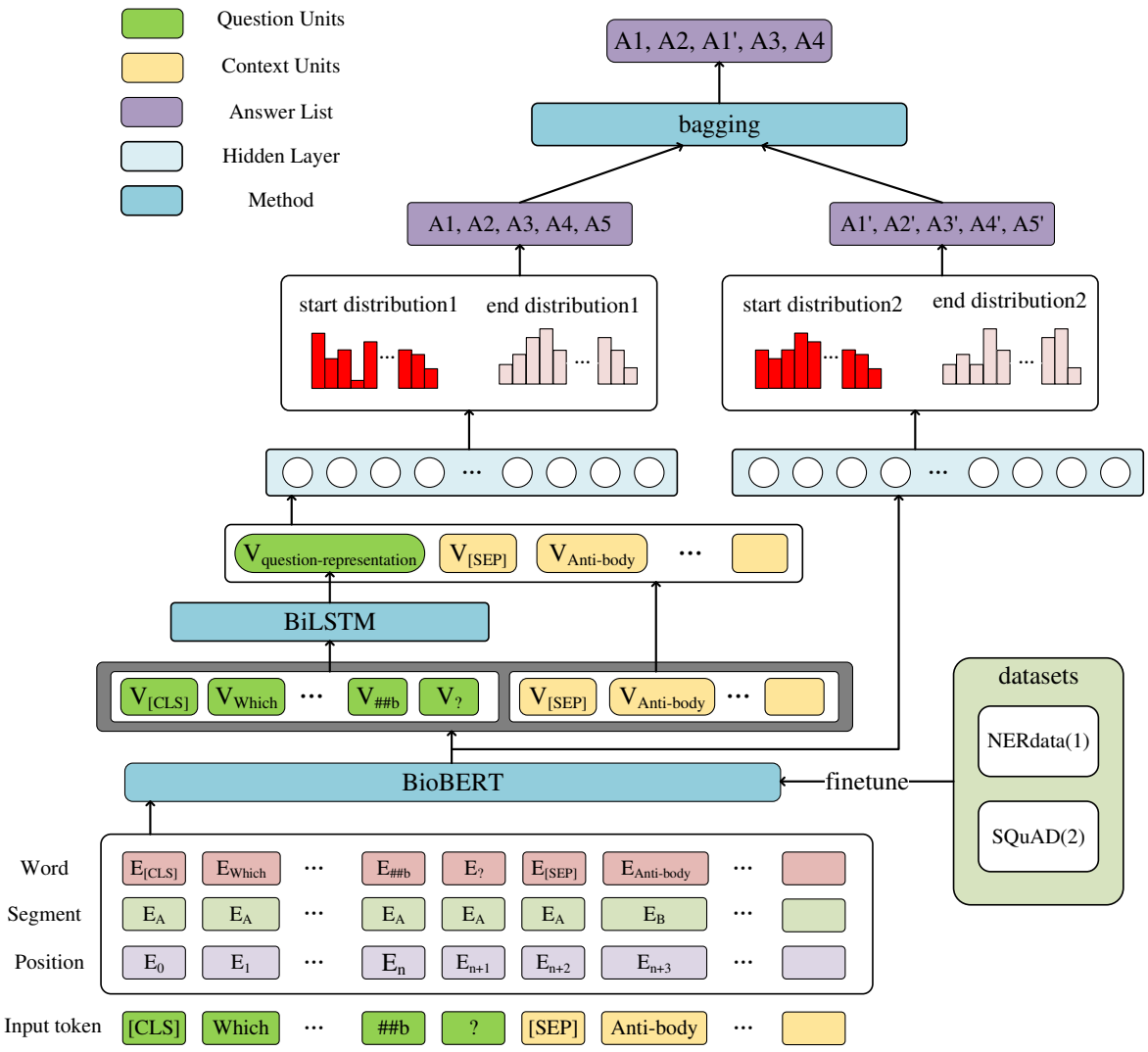

Fig. 1. The Pipeline of Proposed Framework, which consists of three parts: 1. Fine-tuning in NER data and SQuAD; 2. BiLSTM to learn the question representation; 3. Bagging. In the datasets part, the numbers in parentheses represent the order of fine-tune process. The input is "*Which enzyme is targeted by Evolocumab? Antibody therapeutics in Phase 3 ...*" and finally we will get the top-five answers with the highest probabilities.

the five answers with the highest probability as the final answers[2].

3.2 fine-tuned Oriented Transfer Learning

3.2.1 Basic Architecture

Word embeddings are crucial in natural language processing tasks because they contain semantic and syntactic information [51], [52]. In the biomedical domain, traditional models either use the feature engineering method [36] to extract features or use context-independent word embeddings, which cannot accurately represent contextual information. Recently, researchers have begun to use contextualized word representation [7], [8], [53], among which BERT has achieved SOTA performance in various tasks. In the biomedical field, BioBERT [11] has been proposed, which is pretrained on PubMed and outperforms other methods in various biomedical tasks.

For the BioBERT model, an input representation of a token is composed of a token, segment, and position embedding. Becasue BioBERT and BERT have a large number of words, every word is separated into many tokens that can avoid the out-of-vocabulary problem. For the context $C = \{c_1, c_2, ..., c_m\}$ and the question $Q = \{q_1, q_2, ..., q_n\}$, we suppose that the context's input embeddings are $E_C = \{e_{c_1}, e_{c_2}, ..., e_{c_m}\}$ and the input embeddings of the question are $E_Q = \{e_{q_1}, e_{q_2}, ..., e_{q_n}\}$. We can then combine the embeddings of the question and context to form the input of BioBERT $I = [[CLS], Q, [SEP], C, [SEP]]$. Subsequently, in BioBERT, these input tokens will learn the information of the relationship between them using multi-layer transformer encoders [54]. The transformer encoder mainly consists of two parts: multi-head attention and a fully connected feed-forward network. In this architecture, each token has three representations: query, keys, and val-

2. The source code is avaliable at https://github.com/Romainpkq/bioasq_factoid_qa

ues. We represent the three representations of all tokens as matrices Q_1, K, and V, respectively. The equation of multi-head attention is shown in Eq. (1), Eq. (2) and Eq. (3) [54]:

$$Attnetion(Q_1, K, V) = softmax(\frac{Q_1 K^T}{\sqrt{d_k}})V \quad (1)$$

$$MultiHead(Q_1, K, V) = Concat(head_1, ..., head_h)W^O \quad (2)$$

$$head_i = Attention(Q_1 W_i^{Q_1}, K W_i^K, V W_i^V) \quad (3)$$

where $W_i^{Q_1}$, W_i^K, and W_i^V represent the parameter matrices. For the feed-forward networks, the process is shown in Eq. (4):

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2 \quad (4)$$

The architecture of the transformer encoder can learn a good representation for each token, and the BioBERT model is based on this architecture. Although BERT has learned a good representation for each token, it still ignores some important information. First, because the representations of words in the BERT model are based on the tokens and it is an autoencoder model, it cannot learn the information of named entities well. However, named entity is very important to improve the performance of tasks. Hence, we use an NER dataset to fine-tune BioBERT to learn the information of the named entities. At the same time, because of the small amount of data in the biomedical domain, the BioBERT model lacks general knowledge. Therefore, we also use the SQuAD dataset to fine-tune the BioBERT model, which has been proven effective by Yoon et al. [12].

3.2.2 NER Based fine-tuning

In this study, we first fine-tune BioBERT with the NER dataset to learn the named entity information. Here, we suppose that a sample in the dataset is formed as $P = \{(x_1, l_1), (x_2, l_2), ..., (x_m, l_m)\}$, where x_i represents the i-th token in the sample and l_i represents the named entity of the i-th token, and $l_i \in \{label_1, label_2, ..., label_p\}$, where p is the number of named entities. We use the same method to encode the tokens and then input the embeddings into BioBERT to obtain the output embeddings $O_{1_{NER}} = \{o_{x_1}, o_{x_2}, ..., o_{x_m}\}$ for the passage. Subsequently, we input the output embeddings $O_{1_{NER}}$ into a neural network layer; for each token embedding in the output embeddings, we calculate the output as follows:

$$l_i^{pred} = W_1 * o_{x_i} + b_1 \quad (5)$$

where l_i^{pred} represents the output of the i-th token, W_1 and b_1 are the weight matrix and bias, respectively, and the dimensions of $l_i^{pred} = [l_{i_1}^{pred}, l_{i_2}^{pred}, ..., l_{i_p}^{pred}]$ is the number of labels. We use the $softmax$ function on the output of each token to obtain the probabilities of the labels for each token, and we use the cross-entropy function as the loss function. The entire process is as follows.

$$p_{i_r}^{pred} = \frac{exp(l_{i_r}^{pred})}{\sum_{j=1}^{p} exp(l_{i_j}^{pred})} \quad (6)$$

$$loss_i = \sum_{r=1}^{p} p_{i_r} * log(p_{i_r}^{pred}) \quad (7)$$

where $p_{i_r}^{pred}$ represents the possibility that the label of the i-th token is the r-th label in the label set, $l_{i_r}^{pred}$ represents the r-th term of the output of the i-th token and $loss_i$ represents the loss of the i-th token. p_{i_r} represents the real possibility that the label of the i-th token is the j-th label in the label set. It is equal to 1 if the label is the real label, and 0 otherwise.

3.2.3 SQuAD Based fine-tuning

Because of the small number of samples in the biomedical datasets, we also trained BioBERT in the dataset SQuAD 1.1, following the work of Yoon et al. [12]. This process is essentially the same as that of NER. First, we obtain the embedding of each token by merging the word, position, and segment embedding. Then, we pass the embeddings to BioBERT to obtain a new embedding for each token $OS = [os_1, os_2, ..., os_{(m+n+3)}]$. Subsequently, we pass the output of BioBERT to a task layer to predict the answer positions in context. The process is depicted in Eq. (8), Eq. (9), and Eq. (10):

$$f_i = W_2 * os_i + b_2 \quad (8)$$

$$p_i^{(1)} = \frac{exp(f_i^{(1)})}{\sum_{j=1}^{n} exp(f_j^{(1)})} \quad (9)$$

$$p_i^{(2)} = \frac{exp(f_i^{(2)})}{\sum_{j=1}^{n} exp(f_j^{(2)})} \quad (10)$$

where os_i represents the embedding of the i-th token in the BioBERT output, and W_2 and b_2 are the parameter matrix and bias, respectively. f_i is the embedding of the i-th token in the task layer output. $f_i^{(1)}$ and $f_i^{(2)}$ are the first and second elements in f_i, respectively, which represent the start and end probabilities of the token, respectively, and n is the total number of tokens in a sample. The loss is defined as follows:.

$$loss = \frac{1}{2}(\sum_{i=1}^{j}(l_i^{(1)} * log(p_i^{(1)}) + l_i^{(2)} * log(p_i^{(2)}))) \quad (11)$$

where $l_i^{(1)}$ and $l_i^{(2)}$ represent the real probability of the token as the start position and end position, respectively, and $l_i^{(1)}$ equals 1 if the token is the start position and 0 otherwise, $l_i^{(2)}$ equals 1 if the token is the end position and 0 otherwise. After fine-tuning in SQuAD, BioBERT can effectively learn the syntactic and semantic information of the QA dataset.

3.3 BiLSTM Based Question Representation

After continuously fine-tuning BioBERT on the two datasets, i.e., the NER dataset and SQuAD, the BioBERT model can better represent the information of named entities and general knowledge. As shown in section 2.2, the form of input to BioBERT is $O_0 = [[CLS], Q, [SEP], C, [SEP]]$, and the output is $O_1 = [o_{[CLS]}, o_Q, o_{[SEP]}, o_C, o_{[SEP]}]$, where o_Q is the output embedding of the question part of the input, and its form is $o_Q = [o_{q_1}, o_{q_2}, ..., o_{q_n}]$, where o_{q_i} represents the output embedding of the i-th token in the question. o_C is the output embedding of the context part, and its form is $o_C = [o_{c_1}, o_{c_2}, ..., o_{c_m}]$, where o_{c_i} represents the output embedding of the i-th token in the context.

Because the quality of the question representation is important for determining the position of the answer in the context, a better understanding of the question can improve the performance of the model. Hence, to learn a good representation of the question, we pass the output embeddings $[o_{[CLS]}, o_Q]$ into a BiLSTM model, as shown in Fig. 2.The BiLSTM based sentence modeling process is defined as below [28]:

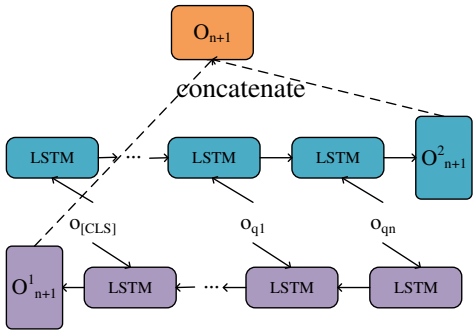

Fig. 2. The Pipeline of BiLSTM in Our Framework

$$i_t = sigmoid(W_{ii} * q_t + b_{ii} + W_{hi} * h_{t-1} + b_{hi}) \quad (12)$$

$$f_t = sigmoid(W_{if} * q_t + b_{if} + W_{hf} * h_{t-1} + b_{hf}) \quad (13)$$

$$g_t = tanh(W_{ig} * q_t + b_{ig} + W_{hg} * h_{t-1} + b_{hg}) \quad (14)$$

$$o_t = sigmoid(W_{io} * q_t + b_{io} + W_{ho} * h_{t-1} + b_{ho}) \quad (15)$$

$$c_t = f_t \cdot q_t + i_t \cdot g_t \quad (16)$$

$$h_t = o_t \cdot tanh(c_t) \quad (17)$$

where h_t is the hidden state at time t, c_t is the cell state at time t, q_t is the input at time t, and q_t is o_{q-1} if $t \neq 1$; otherwise, q_0 is $o_{[CLS]}$. h_{t-1} is the hidden state of the layer at time $t-1$ or the initial hidden state at time 0, and i_t, f_t, g_t, o_t are the input, forget, cell, and output gates, respectively. $sigmoid$ is the sigmoid function whose form is $sigmoid(x) = 1/(1 + exp(-x))$. BiLSTM is a combination of two LSTM models that obtain the input from different directions. Then, we obtain the final outputs o^1_{n+1} and o^2_{n+1}, where o^1_{n+1} represents the output at time $n+1$ for the first LSTM, and o^2_{n+1} is the output at time $n+1$ for the second LSTM. Subsequently, we concatenate the two outputs to obtain the final question representation o_{n+1}. Then, we combine the question representation with the original representation and obtain the output $O_2 = [o_{n+1}, o_{[CLS]}, o_Q, o_{[SEP]}, o_C, o_{[SEP]}]$, and set the vector in $o_{[CLS]}, o_Q$ to 0 to mask the information of the question tokens and obtain $O_2 = [o_{n+1}, 0, .., 0, o_{[SEP]}, o_C, o_{[SEP]}]$, where $n+1$ is the number of zero vectors.

3.4 Ensemble Method Based Classification

The ensemble method is a mechanism that combines existing methods to improve the overall performance, which can effectively improve the stability of the model and solve the problem of unbalanced data distribution. In the proposed framework, we share the parameters of BioBERT for the two different models using the bagging method to combine sentence-level and token-level information. First, we obtain the outputs $O_1 = [o_{[CLS]}, o_Q, o_{[SEP]}, o_C, o_{[SEP]}]$ and $O_2 = [o_{n+1}, 0, .., 0, o_{[SEP]}, o_C, o_{[SEP]}]$ as mentioned above, then we pass the outputs to the two task layers and obtain the final outputs O_{f1} and O_{f2}, respectively. The process is as follows:

$$O^i_{f1} = W_3 * O^i_1 + b_3 \quad (18)$$

$$O^i_{f2} = W_4 * O^i_2 + b_4 \quad (19)$$

where O^i_1 and O^i_2 are the i-th term of the output of the first model O_1 and the second model O_2, respectively; O^i_{f1} and O^i_{f2} are the i-th term of the final outputs O_{f1} and O_{f2}, respectively; $W3$ and $b3$ are the weight and bias of the first model, and $W4$ and $b4$ are the weight and bias of the second model. Then, we obtain O_{f1} and O_{f2}. Because the shape of O_{f1} is (sequence_length+1, output_dim), it has one extra dimension in the first dimension that represents the overall information of the question. We abandon the first dimension of O_{f1} to obtain O_{fn1} whose shape is (sequence_length, output_dim), and then we use $softmax$ to process O_{fn1} and O_{f2}, respectively. Finally, we obtain the possibility P^{pred}_1 and P^{pred}_2. The dimension of P^{pred}_1 and P^{pred}_2 is $(m+n+3, 2)$, where $m+n+3$ is the entire length of the sequence. The last dimension represents the probabilities of the start and end positions of the tokens. We use *cross-entropy* as the loss function, and the process is as follows:

$$loss1 = \sum_{i=1}^{m+n+3} p_{i_1} * log(p1^{pred}_{i_1}) + \sum_{i=1}^{m+n+3} p_{i_1} * log(p2^{pred}_{i_1}) \quad (20)$$

$$loss2 = \sum_{i=1}^{m+n+3} p_{i_2} * log(p1^{pred}_{i_2}) + \sum_{i=1}^{m+n+3} p_{i_2} * log(p2^{pred}_{i_2}) \quad (21)$$

$$loss = (loss1 + loss2)/4 \quad (22)$$

where p_{i_1} and p_{i_2} represent the real probabilities of the i-th tokens as the start and end positions, respectively, for example, p_{i_1} equals to 1 if the i-th token is the start position, else equals to 0. $p1^{pred}_{i_1}$ and $p1^{pred}_{i_2}$ are the results of the two dimensions of the i-th term of the output result P^{pred}_1, $p2^{pred}_{i_1}$ and $p2^{pred}_{i_2}$ are the results of the two dimensions of the i-th term of the output result P^{pred}_2.

In the prediction step, for the two models, we obtain the indexes of the five tokens with the largest starting position probability and the indexes of the five tokens with the largest ending position probability. Subsequently, we combine the start and end indexes to form a whole expression and abandon some situations, such as the start index being greater than the end index. We finally obtain the five most likely answers from each model separately, as follows: $[A1, A2, ..., A5]$ and $[A1', A2', ..., A5']$. Subsequently, we put the whole expressions, including the expressions from the first model and the second model, in the same list and compare the probability of all the expressions to the sum of the start and end probabilities. We rank all the answers by their probabilities, and choose the five highest answers as our final answers. For example, if $P_{A1} > P_{A2} > P_{A1'} > P_{A3} > P_{A4} > ...$, we will get the final answer list $[A1, A2, A1', A3, A4]$.

4 Experimental Study

4.1 Experiment Configuration

In this research, the proposed framework is evaluated on the datasets of BioASQ competition, which is the most formal and influential competition in the biomedical domain. We applied the same step proposed by Yoon et al. [11] to process the BioASQ training data. We use an entire abstract, including the title of an article, as a passage. We first find the given snippet in the abstract, then find the offset of the answer in the snippet, and finally add the offset to the dataset. After data processing, the BioASQ 6b training dataset contained 619 factoid questions and 4772 question-passage pairs, and the BioASQ 7b training dataset contained 779 factoid questions and 5537 question-passage pairs. At the same time, the BioASQ 6b and bioASQ 7b test sets have 161 and 162 questions, respectively, as shown in Table 1.

TABLE 1
Statistics of BioASQ

Version	Train samples	Post-processed question-passage pairs	Test samples
6b	619	4772	161
7b	779	5537	162

We also employed two datasets to fine-tune BioBERT. First, we use the NER dataset, NCBI-disease, which is a collection of 793 PubMed abstracts fully annotated at the mention and concept levels and contains 6892 disease mentions [55], as listed in Table 2. The second dataset is SQuAD 1.1, which is a comprehensive reading dataset that contains more than 100,000 questions. The answer to these questions is either a segment of text or span from the context or it does not exist [56].

TABLE 2
Statistics of the NCBI-disease

	Train set	Dev set	Test set	Total
PubMed citations	593	100	100	793
total disease mentions	5145	787	960	6892
unique disease mentions	1710	368	427	2136
unique concept ID	670	176	203	790

As for the hyperparameters, we first used the NER dataset to fine-tune the BioBERT model, whose dimension of hidden layer is 768. We input the output of the BioBERT to the NER task layer to obtain the fine-tuned parameters, and the number of epochs was 10. Then, we used the SQuAD data to re-fine-tune BioBERT, and the maximum length of the sequence was 384, the number of training epochs was two, and the learning rate was 3e-5. Finally, for the dataset BioASQ 7b, we set the maximum length of the sequence to 384, the maximum length of the question to 64, and the learning rate to 5e-6. We set the number of training epochs to two, and for BioASQ 6b, we set the number of epochs to four. For the question representation, we set the dimension of the hidden layer of LSTM to 384; hence, the dimension of BiLSTM was 768.

4.2 Evaluation Metrics

In the factoid QA system, we aim to determine the start and end positions of the answer in the context and return the final answer. Hence, we need to determine whether the predicted answer was correct. Following the BioASQ competition metrics, the result that we returned was a list, and we used *strict accuracy* (SAcc) and the *lenient accuracy* (LAcc) as the metrics. For SAcc, if the first element in the returned list equals the golden answer, we recorded it as true; otherwise, it was false. For LAcc, if the golden answer was in the returned list, we recorded it as true; otherwise, it was false. At the same time, we used a metric *mean reciprocal rank* (MRR), which can reflect the rank information for all answers in the returned answer list. It is also used to evaluate factoid QA. In this study, we returned a list that contains five predicted answers.

$$SAcc = \frac{c_1}{n} \tag{23}$$

$$LAcc = \frac{c_5}{n} \tag{24}$$

$$MRR = \frac{1}{n} \cdot \sum_{i=1}^{n} \frac{1}{r_i} \tag{25}$$

where c_1 is the number of questions such that the first answer in the predicted answer list is the golden answer, and c_5 is the number of questions such that the golden answer is in the returned answer list, and n is the total number of questions. r_i is the position of the golden answer in the returned answer list and r_i equals j if the golden answer is the j-th answer in the list and $+\infty$ otherwise.

4.3 Baseline Methods

In this study, we compare our method against several recently proposed advanced methods to investigate the potential of the proposed framework, as illustrated below.

1) The AUTH model, which uses the updated version of the BioASQ 6 system and uses the word embeddings and external resources [37], such as MetaMap, BeCAS, and WordNet.

2) "LabZhu" [38] system, which uses two distinct methods based on traditional information retrieval method and knowledge graph based approaches, respectively to find the answer of the factoid questions.

3) The "google-gold-input" system [39], which was developed by Google and uses the BERT model and fine-tuned on the CoQA [57] and the BioASQ datasets.

4) The UNCC [58] system, which uses BioBERT embeddings and fine-tuned on the BioASQ dataset as well as the lexical answer type (LAT) and POS tags to improve performance.

5) The model by Yoon et al. [12], which first uses SQuAD to fine-tune the BioBERT model and then is trained on the BioASQ dataset.

6) The model by Jeong et al. [13], which first uses NLI and then the SQuAD datasets to fine-tune the BioBERT model, and finally is trained in BioASQ dataset.

TABLE 3
Experiment result on BioASQ 6b

Model	6b Factoid QA		
	SAcc	LAcc	MRR
AUTH	0.2015	0.4020	0.2713
LabZhu	0.2387	0.3314	0.2762
BioBERT+SQuAD	0.4286	0.5714	0.4841
BioBERT+MLI+SQuAD	0.4141	0.5740	0.4805
Basic model (BioBERT+NER+SQuAD)	**0.4428**	0.6235	**0.5143**
Basic model with [CLS]	0.4164	0.6298	0.4988
Basic model with BiLSTM	0.4209	0.6298	0.4998
Full model (BiLSTM+bagging)	0.4302	**0.6423**	0.5119

TABLE 4
Experiment result on BioASQ 7b

Model	7b Factoid QA		
	SAcc	LAcc	MRR
AUTH	0.2363	0.3710	0.2898
LabZhu	0.2765	0.3922	0.3252
UNCC	0.3554	0.4922	0.4063
google-gold-input	0.4201	0.5822	0.4798
BioBERT+SQuAD	0.4367	0.6274	0.5115
BioBERT+MLI+SQuAD	0.4510	0.6245	0.5163
Basic model (BioBERT+NER+SQuAD)	0.4697	0.6194	0.5235
Basic model with [CLS]	0.4592	0.6122	0.5207
Basic model with BiLSTM	**0.4790**	0.6191	0.5285
Full model (BiLSTM+bagging)	0.4752	**0.6385**	**0.5323**

4.4 Results and Discussion

We evaluate the proposed framework against the previously proposed models on datasets BioASQ 6b and BioASQ 7b, and the overall results are listed in Table 3 and Table 4.

From these tables it is found that the BioBert with fine-tuning on NER and SQuAD (referred as "Basic model" in the tables and following experiments) can improve the performance of the system in the metric MRR and the metric SAcc compared to the model "BioBERT+SQuAD", which demonstrates that using transfer learning in the NER datasets can learn the information of named entities and improve the strict accuracy. As for the question-level information, from the tables, we can see that BiLSTM often has better performance than [CLS], and when we use BiLSTM to extract question information without bagging, we can see that there are different variations in SAcc and LAcc compared to the Basic model, which means that the two models get different information. However, we also found that in BioASQ 7b, after fine-tuning in NER, the metric LAcc decreases. This may be because the selected answers tend to be more named entities and cause some answers that are not named entities to be excluded, and another reason may be the difference between the name entities in the train dataset and test dataset.

At the same time, from Table 4, we can see that after using the BiLSTM model to extract the question information and the bagging method to combine the two models, our model shows significant improvements in SAcc, LAcc, and MRR, and outperforms the previous best model by 2.4% on SAcc, 1.4% on LAcc, and 1.6% on MRR, which means that the framework can well capture the advantages of the two models, and it also demonstrates that the combination of local and global information can get a better result.

In BioASQ 6b, it also shows the best performance in the metric LAcc, but the metrics SAcc and MRR decrease to some extent. This is possibly caused by the difference in the volume of data in BioASQ 6b and BioASQ 7b, as there are almost 30% more factoid questions in BioASQ 7b than in 6b, and the data in BioASQ 6b is less unbalanced than in BioASQ 7b. Hence, the bagging method cannot achieve a good performance, which demonstrates that the unbalanced data will influence the performance of the model.

We also compared our results for each batch with some systems that participated in the BioASQ 7b competition. The results are listed in Table 5 based on the online leaderboard[3].

TABLE 5
Batch results of the BioASQ 7b Challenge.

Batch	Factoid Model	MRR
1	auth-qa-1	0.2778
	BJUTNLPGroup	0.3483
	Yoon et al.	**0.4637**
	Full model	0.4444
2	transfer-learning	0.3267
	QA1	0.4033
	Yoon et al.	0.5667
	Full model	**0.5913**
3	google-gold-input	0.5023
	QA1/UNCC_QA_1	0.5115
	Yoon et al.	0.4724
	Full model	**0.5201**
4	google-golden-input	0.5495
	FACTOIDS/UNCC_QA_1	0.6103
	Yoon et al.	0.6912
	Full model	**0.7157**
5	UNCC_QA_1	0.3305
	BJUTNLPGroup	0.3381
	Yoon et al.	0.3638
	Full model	**0.3900**

From Table 5, we can see that our proposed framework achieved the best results in four out of five batches. In batch 3, we show an improvement close to 5%, and in batches $2, 4, 5$, we have improved by approximately 2%, which means that our Full model can achieve better results on various data. Then, we investigate the average performance and stability of the proposed framework. Ten experiments were performed in BioASQ 6b and 7b with and without BiLSTM and bagging, respectively, to obtain the average, best, and worst results of the framework. The results of the experiments are listed in Table 7 and Table 6, respectively.

From Table 6 and Table 7, it is observed that our Full model always shows a better performance than the Basic model in LAcc. We also noticed that the Basic model with and without BiLSTM respectively have different variations in SAcc and LAcc which is correspond with the results in Table 3 and Table 4. As the amount of data grows, the Basic model with BiLSTM achieves better performance in MRR, which demonstrates the effectiveness of the BiLSTM. From the results of BioASQ 7b, it is found that using the BiLSTM

3. The official competition results are released by BioASQ

TABLE 6
Experiment Results on BioASQ 6b

Experiments	Full model (BiLSTM+bagging)			Basic model with BiLSTM			Basic model		
	SAcc	LAcc	MRR	SAcc	LAcc	MRR	SAcc	LAcc	MRR
1	0.4116	0.6487	0.5022	0.4101	0.6313	0.4913	0.4154	0.6300	0.5034
2	0.3987	0.6298	0.4929	0.4101	0.6423	0.4994	0.4428	0.6235	0.5143
3	0.4196	0.6423	0.5071	0.4147	0.6298	0.4967	0.4287	0.6171	0.5041
4	0.3987	0.6487	0.4954	0.4147	0.6298	0.4960	0.4218	0.6235	0.5020
5	0.4211	0.6358	0.5068	0.4114	0.6362	0.4960	0.4173	0.6235	0.4966
6	0.4257	0.6298	0.5036	0.4209	0.6188	0.4968	0.4238	0.6300	0.5041
7	0.3987	0.6362	0.4930	0.4147	0.6298	0.4957	0.4192	0.6425	0.5049
8	0.3941	0.6423	0.4941	0.4205	0.6298	0.4998	0.4298	0.6300	0.5091
9	0.4097	0.6423	0.5005	0.4114	0.6298	0.4956	0.4203	0.6325	0.5000
10	0.4302	0.6423	0.5119	0.4114	0.6298	0.4947	0.4302	0.6171	0.5040
average	0.4108	0.6399	0.5008	0.4140	0.6307	0.4962	0.4249	0.6270	0.5043
max	+0.0194	+0.0088	+0.0111	+0.0069	+0.0116	+0.0037	+0.0179	+0.0155	+0.0100
min	-0.0167	-0.0101	-0.0079	-0.0039	-0.0119	-0.0048	-0.0095	-0.0099	-0.0077

TABLE 7
Experiment Results on BioASQ 7b

Experiments	Full model (BiLSTM+bagging)			Basic model with BiLSTM			Basic model		
	SAcc	LAcc	MRR	SAcc	LAcc	MRR	SAcc	LAcc	MRR
1	0.4752	0.6385	0.5323	0.4417	0.6138	0.5084	0.4697	0.6194	0.5235
2	0.4697	0.6336	0.5261	0.4594	0.6248	0.5208	0.4438	0.6320	0.5198
3	0.4695	0.6118	0.5228	0.4754	0.6134	0.5256	0.4435	0.6317	0.5173
4	0.4754	0.6200	0.5363	0.4499	0.6075	0.5094	0.4390	0.6246	0.5109
5	0.4695	0.6182	0.5208	0.4680	0.6139	0.5243	0.4514	0.6311	0.5203
6	0.4558	0.6252	0.5160	0.4600	0.6134	0.5191	0.4390	0.6258	0.5108
7	0.4733	0.6246	0.5262	0.4565	0.6246	0.5182	0.4379	0.6142	0.5061
8	0.4752	0.6326	0.5273	0.4790	0.6191	0.5285	0.4272	0.6246	0.5016
9	0.474	0.6326	0.5299	0.4731	0.6081	0.5252	0.4461	0.6132	0.5106
10	0.4674	0.6179	0.5240	0.4548	0.6139	0.5185	0.4577	0.6317	0.5220
average	0.4705	0.6255	0.5252	0.4618	0.6152	0.5198	0.4455	0.6248	0.5143
max	+0.0049	+0.0130	+0.0071	+0.0172	+0.0096	+0.0087	+0.0242	+0.0072	+0.0092
min	-0.0147	-0.0137	-0.0092	-0.0201	-0.0077	-0.0114	-0.0183	-0.0116	-0.0127

and bagging method results in a more stable performance in SAcc and MRR, and less stable performance in LAcc. This could be because we choose the maximum probability answer from the two models, which means that it considers the information of both the models. Hence, it has higher stability in the SAcc and MRR, but because the bagging method provides more alternative answers, the LAcc metric is less stable. In the results of BioASQ 6b, it is noticed that the SAcc and MRR decrease after using the BiLSTM and bagging, which might be because the BioASQ 6b dataset is smaller and the phenomenon of data imbalance is less significant. Therefore, the bagging method cannot perform as well as the other methods. We can also compare our Full model against the baselines. From the comparison, it is found that most results of our Full model show better performance, and the average results of SAcc and MRR are much higher than those of the previous models.

For the hyperparameters in the framework, we mainly changed the number of epochs to find the best performance. In BioASQ 7b, during the training step, we used different number of epochs, i.e., 1, 2, 3, 4, 5, and we determine the best number of epochs as 2. For BioASQ 6b, we perform the same experiment and determine the best number of epochs to be 4. The results are shown in Table 8 and Table 9.

TABLE 8
Hyperparameter Configuration: Number of Training Epochs in the BioASQ 7b Dataset

Number of Training Epochs	SAcc	LAcc	MRR
1	0.4681	0.6305	0.5218
2	**0.4754**	0.6296	**0.5328**
3	0.4636	**0.6355**	0.5246
4	0.4636	0.6355	0.5246
5	0.4567	0.6355	0.5219

TABLE 9
Hyperparameter Configuration: Number of Training Epochs in the BioASQ 6b Dataset

Number of Training Epochs	SAcc	LAcc	MRR
1	0.3809	0.6200	0.4761
2	0.3858	0.6358	0.4887
3	0.3816	0.6358	0.4820
4	0.3987	**0.6487**	**0.4954**
5	**0.4097**	0.6298	0.4947

4.5 Case Study

In this section, we introduce some test cases to prove the effectiveness of fine-tuning BioBERT on the NER dataset and the effectiveness of question representation and bagging

Example1:
Question: What is the function of the protein Magt1?
Context: ...Recently, patients with a loss-of-function mutation in magnesium transporter 1 (MAGT1) were reported to present a dysregulated Mg homeostasis in T lymphocytes...
Exact answer: Magnesium transporter
Baseline Model (BioBERT+SQuAD): magnesium[1], magnesium transporter[2], magnesium transporter 1[3], magnesium transporter MagT1 have a role in harmonizing the osteogenic differentiation[4], magnesium transport[5]
Our model without BiLSTM and bagging: magnesium transporter[1], magnesium transporter 1 (MAGT1) is a critical regulator of basal intracellular free magnesium[2], critical regulator of basal intracellular free magnesium[3], magnesium transporter MagT1 have a role in harmonizing the osteogenic differentiation[4], magnesium[5]
Our final model: magnesium transporter[1], magnesium transporter 1 (MAGT1) is a critical regulator of basal intracellular free magnesium[2], critical regulator of basal intracellular free magnesium[3], magnesium transporter MagT1 have a role in harmonizing the osteogenic differentiation[4].

Example2:
Question: Rickettsia felis was described as a human pathogen almost two decades ago, what is it's main arthropod vector?
Context: Cat fleas (Ctenocephalides felis) carrying Rickettsia felis and Bartonella species in Hong Kong...
Exact answer: Cat fleas (Ctenocephalides felis)
Baseline model (BioBERT+SQuAD): Bartonella[1], Bartonella and Rickettsia real-time PCR of DNA from 20 C. felis found Bartonella[2], B. clarridgeiae and Bartonella[3], B. clarridgeiae[4], Bartonella henselae[5]
Our model without BiLSTM and bagging: Cat fleas[1], Bartonella[2], cat flea and to be harbouring pathogens of zoonotic potential[3], Cat fleas (Ctenocephalides felis) carrying Rickettsia felis and Bartonella[4], Bartonella and Rickettsia real-time PCR of DNA from 20 C. felis found Bartonella[5]
Our final model: Cat fleas[1], Cat fleas (Ctenocephalides felis) carrying Rickettsia felis and Bartonella[2], Cat fleas (Ctenocephalides felis)[3], fleas[4], Bartonella[5]

Fig. 3. Case Studies: The Number Following the Answer is its Rank in the Alternative Answer List

mechanism. The examples are shown in Fig. 3.

From Fig. 3, we can see that our framework is effective. In the first example, the answer to the question is a biomedical named entity. For the baseline model that is not fine-tuned in the NER dataset, it returns $magnesium$ as the first alternative answer, while in the context, the $magnesium$ and $transporter$ together form a named entity. The baseline could not recognize it well. However, after fine-tuning the model in the NER dataset, it can recognize the named entity, which shows that fine-tuning in the NER dataset is effective. In the second example, the exact answer is not in the alternative answers of the baseline model. It is also not present in our model without BiLSTM and bagging; however, the answer *Cat fleas* appears in the alternative answers, which means the model has noticed the exact answer. In the proposed framework, the exact answer appears in the alternative answers, which means that our ensemble method is effective. At the same time, we can see that the unrelated answer $Bartonella$ is the last answer of the alternative answers, and the other alternative answers are almost irrelevant to the word $Bartonella$, which means that the model has considered the whole question information, and it finds the information related to the question information *main anthropod vector*. From the examples, we can see that our framework has learned the information of named entities, and it considers the overall information of the question.

5 Conclusion and Future Work

Recently, the QA system has attracted a lot of attention because it can quickly obtain the exact answer rather than browsing through a list of articles. In the biomedical QA application, the questions are usually divided into four types: 1) "yes" or "no" questions, 2) factoid questions, 3) list questions, and 4) summary questions. Among them, the answers to the factoid questions can usually be found in the context, which means it has a higher credibility. Therefore, it has been attached much importance in the community.

To provide satisfactory performance for the factoid QA system, it is important to understand the sentence and word in the QA text. Recently, pretrained language models have achieved great success in diverse natural language processing tasks. BioBERT has also shown promising improvement for biomedical applications. Although using the BioBERT model can obtain proper word embedding for biomedical tasks, owing to its autoencoder characteristics, it cannot learn the named entity and the overall question information well. Therefore, inspired by the success of transfer learning in other applications, in this study, we utilized the transfer learning mechanism to learn named entity information and general knowledge. Furthermore, we also applied the BiLSTM to learn the overall information of the question because a proper understanding of sentence information is also essential for the overall performance. Finally, to fully make use of the sentence-level and token-level information, we also applied the bagging mechanism to combine the strength of sentence-level and token-level answer prediction. The proposed framework achieved promising performance in both the BioASQ 6b and 7b shared tasks.

Although the proposed method has significant improvement, there are also limitations that deserve further investigation. From the experiments, it was found that although the bagging method can make the result more stable, the stability is not satisfactory. There seems to be an inconsistency between the metrics SAcc and LAcc, warranting further studies for finding a more stable model. Furthermore, because of the difference between the train dataset and test dataset, how to recognize the name entities that do not appear in the training data is still a problem which deserves further investigation.

At the same time, because transfer learning has proven effective in the biomedical domain, ways to use transfer learning to solve other questions in the biomedical domain and learn more external information can be explored in further studies. In the future, we will also try to research the application of transfer learning to other biomedical problems and solve the instability of the proposed framework.

Acknowledgements

This work was partially supported by the State Key Laboratory of the Software Development Environment of China (No. SKLSDE-2021ZX-16), and the National Natural Science Foundation of China (No. 61977003).

References

[1] M. Neves and U. Leser, "Question answering for biology," *Methods*, vol. 74, pp. 36–46, 2015.

[2] G. Tsatsaronis, G. Balikas, P. Malakasiotis, I. Partalas, M. Zschunke, M. R. Alvers, D. Weissenborn, A. Krithara, S. Petridis, D. Polychronopoulos, Y. Almirantis, J. Pavlopoulos, N. Baskiotis, P. Gallinari, T. Artières, A. N. Ngomo, N. Heino, É. Gaussier, L. Barrio-Alvers, M. Schroeder, I. Androutsopoulos, and G. Paliouras, "An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition," *BMC Bioinformatics*, vol. 16, pp. 138:1–138:28, 2015.

[3] M. W. Bilotti, J. L. Elsas, J. G. Carbonell, and E. Nyberg, "Rank learning for factoid question answering with linguistic and semantic constraints," in *Proceedings of the 19th ACM Conference on Information and Knowledge Management*, 2010, pp. 459–468.

[4] H. Alkharusi, "Categorical variables in regression analysis: A comparison of dummy and effect coding," *International Journal of Education*, vol. 4, no. 2, p. 202, 2012.

[5] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31th International Conference on Machine Learning*, vol. 32, 2014, pp. 1188–1196.

[6] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proceedings of 1st International Conference on Learning Representations Workshop Track*, 2013.

[7] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *CoRR*, vol. abs/1802.05365, 2018.

[8] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019, pp. 4171–4186.

[9] A. Talmor and J. Berant, "Multiqa: An empirical investigation of generalization and transfer in reading comprehension," in *Proceedings of the 57th Conference of the Association for Computational Linguistics*, 2019, pp. 4911–4921.

[10] Z. Yang, Y. Zhou, and E. Nyberg, "Learning to answer biomedical questions: OAQA at BioASQ 4B," in *Proceedings of the 4th BioASQ Workshop*, 2016, pp. 23–37.

[11] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "Biobert: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 2020.

[12] W. Yoon, J. Lee, D. Kim, M. Jeong, and J. Kang, "Pre-trained language model for biomedical question answering," in *Proceedings of International Workshops of ECML PKDD 2019 on Machine Learning and Knowledge Discovery in Databases, Part II*, 2019, pp. 727–740.

[13] M. Jeong, M. Sung, G. Kim, D. Kim, W. Yoon, J. Yoo, and J. Kang, "Transferability of natural language inference to biomedical question answering," *CoRR*, vol. abs/2007.00217, 2020.

[14] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," in *Proceedings of 2019 Annual Conference on Neural Information Processing Systems*, 2019, pp. 5754–5764.

[15] E. Noguera, A. Toral, F. Llopis, and R. Muñoz, "Reducing question answering input data using named entity recognition," in *Proceedings of 8th International Conference on Text, Speech and Dialogue*, 2005, pp. 428–434.

[16] Y. Tateisi and J. Tsujii, "Part-of-speech annotation of biology research abstracts," in *Proceedings of the 4th International Conference on Language Resources and Evaluation*, 2004.

[17] A. Lamurias and F. M. Couto, "Lasigebiotm at MEDIQA 2019: Biomedical question answering using bidirectional transformers and named entity recognition," in *Proceedings of the 18th BioNLP Workshop and Shared Task*, 2019, pp. 523–527.

[18] B. Alshaikhdeeb and K. Ahmad, "Biomedical named entity recognition: A review," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 6, no. 6, pp. 889–895, 2016.

[19] S. Min, M. J. Seo, and H. Hajishirzi, "Question answering through transfer learning from large fine-grained supervision data," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 2017, pp. 510–517.

[20] C. Lee and H. Lee, "Cross-lingual transfer learning for question answering," *CoRR*, vol. abs/1907.06042, 2019.

[21] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 2019, pp. 3980–3990.

[22] C. May, A. Wang, S. Bordia, S. R. Bowman, and R. Rudinger, "On measuring social biases in sentence encoders," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019, pp. 622–628.

[23] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating text generation with BERT," in *Proceedings of 8th International Conference on Learning Representations*, 2020.

[24] Y. Qiao, C. Xiong, Z. Liu, and Z. Liu, "Understanding the behaviors of BERT in ranking," *CoRR*, vol. abs/1904.07531, 2019.

[25] X. Zhu, T. Li, and G. de Melo, "Exploring semantic properties of sentence embeddings," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 2018, pp. 632–637.

[26] A. Conneau, G. Kruszewski, G. Lample, L. Barrault, and M. Baroni, "What you can cram into a single \$&!#* vector: Probing sentence embeddings for linguistic properties," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 2018, pp. 2126–2136.

[27] Y. Hao, X. Liu, J. Wu, and P. Lv, "Exploiting sentence embedding for medical question answering," in *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, 2019, pp. 938–945.

[28] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[29] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes, "Supervised learning of universal sentence representations from natural language inference data," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 670–680.

[30] T. G. Dietterich, "Ensemble methods in machine learning," in *Proceedings of the 1st International Workshop on Multiple Classifier Systems*, 2000, pp. 1–15.

[31] Y. Papanikolaou, D. Dimitriadis, G. Tsoumakas, M. Laliotis, N. Markantonatos, and I. P. Vlahavas, "Ensemble approaches for large-scale multi-label classification and question answering in biomedicine," in *Proceedings of Working Notes for CLEF 2014 Conference and Labs of the Evaluation Forum*, vol. 1180, 2014, pp. 1348–1360.

[32] S. Oh, M. S. Lee, and B. Zhang, "Ensemble learning with active example selection for imbalanced biomedical data classification," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, no. 2, pp. 316–325, 2011.

[33] R. J. B. Dobson, P. B. Munroe, M. J. Caulfield, and M. A. S. Saqi, "Predicting deleterious nssnps: an analysis of sequence and structural attributes," *BMC Bioinformatics*, vol. 7, p. 217, 2006.

[34] G. Li, H. Meng, W. Lu, J. Y. Yang, and M. Q. Yang, "Asymmetric bagging and feature selection for activities prediction of drug molecules," *BMC Bioinformatics*, vol. 9, no. S-6, 2008.

[35] S. J. Athenikos and H. Han, "Biomedical question answering: A survey," *Computer Methods and Programs in Biomedicine*, vol. 99, no. 1, pp. 1–24, 2010.

[36] Z. Yang, N. Gupta, X. Sun, D. Xu, C. Zhang, and E. Nyberg, "Learning to answer biomedical factoid & list questions: OAQA at BioASQ 3B," in *Proceedings of Working Notes of CLEF 2015 Conference and Labs of the Evaluation Forum*, vol. 1391, 2015.

[37] D. Dimitriadis and G. Tsoumakas, "Word embeddings and external resources for answer processing in biomedical factoid question answering," *Journal of Biomedical Informatics*, vol. 92, 2019.

[38] S. Peng, R. You, Z. Xie, B. Wang, Y. Zhang, and S. Zhu, "The fudan participation in the 2015 BioASQ challenge: Large-scale biomedical semantic indexing and question answering," in *Proceedings of Working Notes of CLEF 2015 Conference and Labs of the Evaluation Forum*, vol. 1391, 2015.

[39] S. Hosein, D. Andor, and R. T. McDonald, "Measuring domain portability and error propagation in biomedical QA," in *Proceedings of International Workshops of ECML PKDD 2019 on Machine Learning and Knowledge Discovery in Databases, Part II*, 2019, pp. 686–694.

[40] Y. Bengio, P. Y. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.

[41] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[42] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," in *Proceedings of the 8th Workshop on Syntax, Semantics and Structure in Statistical Translation*, 2014, pp. 103–111.

[43] M. Tan, B. Xiang, and B. Zhou, "Lstm-based deep learning models for non-factoid answer selection," *CoRR*, vol. abs/1511.04108, 2015.
[44] Z. Li, J. Huang, Z. Zhou, H. Zhang, S. Chang, and Z. Huang, "Lstm-based deep learning models for answer ranking," in *Proceedings of the 1st IEEE International Conference on Data Science in Cyberspace*, 2016, pp. 90–97.
[45] G. Wiese, D. Weissenborn, and M. L. Neves, "Neural domain adaptation for biomedical question answering," in *Proceedings of the 21st Conference on Computational Natural Language Learning*, 2017, pp. 281–289.
[46] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.
[47] J. H. Friedman, "Stochastic gradient boosting," *Computational Statistics & Data Analysis*, vol. 38, no. 4, pp. 367–378, 2002.
[48] Y. Huang, J. Cai, L. Ji, and Y. Li, "Classifying g-protein coupled receptors with bagging classification tree," *Computational Biology and Chemistry*, vol. 28, no. 4, pp. 275–280, 2004.
[49] H. K. Fatlawi and A. Kiss, "On robustness of adaptive random forest classifier on biomedical data stream," in *Proceedings of 12th Asian Conference on Intelligent Information and Database Systems*, 2020, pp. 332–344.
[50] Y. Papanikolaou, D. Dimitriadis, G. Tsoumakas, M. Laliotis, N. Markantonatos, and I. P. Vlahavas, "Ensemble approaches for large-scale multi-label classification and question answering in biomedicine," in *Proceedings of Working Notes for CLEF 2014 Conference*, 2014, pp. 1348–1360.
[51] S. Pyysalo, F. Ginter, H. Moen, T. Salakoski, and S. Ananiadou, "Distributional semantics resources for biomedical text processing," in *Proceedings of the 5th International Symposium on Languages in Biology and Medicine*, 2013.
[52] J. P. Turian, L. Ratinov, and Y. Bengio, "Word representations: A simple and general method for semi-supervised learning," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 2010, pp. 384–394.
[53] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.
[54] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of 2017 Annual Conference on Neural Information Processing Systems*, 2017, pp. 5998–6008.
[55] R. I. Dogan, R. Leaman, and Z. Lu, "NCBI disease corpus: A resource for disease name recognition and concept normalization," *Journal of Biomedical Informatics*, vol. 47, pp. 1–10, 2014.
[56] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100, 000+ questions for machine comprehension of text," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 2383–2392.
[57] S. Reddy, D. Chen, and C. D. Manning, "Coqa: A conversational question answering challenge," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 249–266, 2019.
[58] S. K. Telukuntla, A. Kapri, and W. Zadrozny, "UNCC biomedical semantic question answering systems.BioASQ: Task-7B, Phase-B," *CoRR*, vol. abs/2002.01984, 2020.

Keqin Peng received the BSc degree from the Sino-French Engineer School School, Beihang Universiyt, China, in 2019. He is currently working toward the MSc degree in the Sino-French Engineer School, Beihang University. His research interests include machine learning, artificial intelligence, information retrieval, bioinformatics,etc.

Chuantao Yin works as associate professor in Sino-French Engineer school at Beihang University in China. He received his PhD degree on computer science in 2010 from École Centrale de Lyon, France. His research activities mainly focused on machine learning, artficial intelligence, smart education, and etc.

Wenge Rong is professor at School of Computer Science and Engineering, Beihang University, China. He received his PhD from University of Reading, UK, in 2010; MSc from Queen Mary College, UK, in 2003; and BSc from Nanjing University of Science and Technology, China, in 1996. He has many years of working experience as a senior software engineer in numerous research projects and commercial software products. His area of research covers machine learning, natural language processing, and information management.

Chenghua Lin is Senior Lecturer in Natural Language Processing in the Department of Computer Science at the University of Sheffield. He received his PhD in Computer Science from the University of Exeter in 2011. Prior to joining Sheffield, he was a SICSA Senior Lecturer in the Department of Computing Science, University of Aberdeen.

Dayu Zhou received the MS degree in computer science and technology from Nanjing University, Nanjing, China, in 2003 and the PhD degree in computer science and technology from the University of Reading, United Kingdom, in 2008. He is currently a full professor with Southeast University, Nanjing, China. His research interest includes natural language processing, opinion mining, event extraction, sentiment analysis, and bioinformatics.

Zhang Xiong is professor in School of Computer Science of Engineering of Beihang University and director of the Advanced Computer Application Research Engineering Center of National Educational Ministry of China. He has published over 200 referred papers in international journals and conference proceedings and won a National Science and Technology Progress Award. His research interests and publications span from smart cities, knowledge management, information systems and etc.