

Apply deep learning to improve the question analysis model in the Vietnamese question answering system

Dang Thi Phuc¹, Dang Van Nghiem¹, Bui Binh Minh¹, Tran My Linh¹, Dau Sy Hieu²

¹Department of Computer Science, Faculty of Information Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam

²Department of Applied Physics, Faculty of Applied Science, University of Technology-Viet Nam National University HCMC, Ho Chi Minh City, Vietnam

Article Info

Article history:

Received Aug 2, 2022

Revised Sep 9, 2022

Accepted Oct 1, 2022

Keywords:

Best matching 25

Bidirectional encoder representations from transformers

Natural language processing

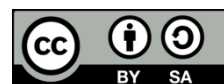
Question answering system

deep learning

ABSTRACT

Question answering (QA) system nowadays is quite popular for automated answering purposes, the meaning analysis of the question plays an important role, directly affecting the accuracy of the system. In this article, we propose an improvement for question-answering models by adding more specific question analysis steps, including contextual characteristic analysis, pos-tag analysis, and question-type analysis built on deep learning network architecture. Weights of extracted words through question analysis steps are combined with the best matching 25 (BM25) algorithm to find the best relevant paragraph of text and incorporated into the QA model to find the best and least noisy answer. The dataset for the question analysis step consists of 19,339 labeled questions covering a variety of topics. Results of the question analysis model are combined to train the question-answering model on the data set related to the learning regulations of Industrial University of Ho Chi Minh City. It includes 17,405 pairs of questions and answers for the training set and 1,600 pairs for the test set, where the robustly optimized BERT pre-training approach (RoBERTa) model has an F1-score accuracy of 74%. The model has improved significantly. For long and complex questions, the model has extracted weights and correctly provided answers based on the question's contents.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Dang Thi Phuc

Department of Computer Science, Faculty of Information Technology, Industrial University of Ho Chi Minh City

12 Nguyen Van Bao, Ward 4, Go Vap District, Ho Chi Minh City 7000, Vietnam

Email: phucdt@iuh.edu.vn

1. INTRODUCTION

Nowadays digitization of documents is going day by day in every single organization for storage or worldwide approach purposes. With text data, somehow the demand on running through contents, extracting the meaning of documents, or searching for a suitable answer for a certain question is playing a big demand. Due to the enlarging of data, the search process now becomes more and more difficult. Therefore, we need a search engine for the fastest and most suitable answers to requested questions. For this purpose, question analysis should be the first element in the architecture of a question-answering (QA) system. It is responsible for finding the necessary information as input for the next steps as per extracting documents or extracting expected answers. That's why the question analysis has an important role and directly affects the operation of the QA system. The question analysis step mostly focuses on question classification with two widely used approaches: rule and statistical probability [1]–[5]. The rule approach is that linguists will provide the rules,

regular expressions, and keywords for each question, but the grammar problem uses to be very difficult to control and depend on the characterization of each linguistic [6], [7]. Modeling for this method, thus, is very time-consuming and labor-intensive, requiring the cooperation of experts in the field of linguistics when building question patterns and grammar for each of those types of questions. When expanding or changing the question dataset, the previous rules must be rebuilt, which makes the system very difficult to scale. The statistical probability approach was synthesized including two main approaches, which are language modeling and machine learning. Method using statistical language models to estimate the distribution of natural languages as accurately as possible [8]. In addition, in combination with the method of extracting keywords where an important word is extracted using the term frequency-inverse document frequency (TF-IDF) algorithm, we can achieve better results [9]. The machine learning approach currently plays a great interest to many researchers due to less human effort required, is highly portable, and is easily applied to many different applications [10]. Commonly used algorithms are support vector machines (SVM) [11], K-nearest neighbors (k-NN) [12], naïve Bayes (NB) [13], artificial neural network (ANN) [14], and deep learning model such as convolutional neural network (CNN) [15], recurrent neural network (RNN) [16]. RNN achieves many good results in question analysis problems, especially the combination of long short-term memory (LSTM), and gated recurrent unit (GRU) with Word2vec techniques leads to significant efficiencies for extracting features [17]–[20]. here are disadvantages to the above method, but the Attention mechanism has partly contributed to overcoming them. The emergence of the attention mechanism replaced RNN, the recent models almost eliminated RNN in its architecture [21], [22]. The current prominent model is the transformer model [23] and bidirectional encoder representations from transformers (BERT) [24].

In this paper, we build the analysis and QA system based on the documents provided. The author's automatic question-answering system will analyze the question and extract the keywords which high weight in the question and based on these keywords find the most relevant text (containing those keywords) in the documentation provided. The deep learning model is applied to extract the corresponding answer in documents.

2. RESEARCH METHOD

2.1. Related algorithms

The question analysis is quite important for finding the correct answer later. A method used to use for extracting keywords and important words from questions is the TF-IDF, however, in many complex questions, the word weights evaluated by TF-IDF are not effective. In recent years, with the development of deep learning technology, the big data problem has been significantly improved, remarkable developments are CNN and RNN combined with the attention mechanism which is highly effective for the problem of natural language processing (NLP). For example, transformer and BERT are outstanding efficiency models for NLP. In this paper, we propose a deep learning model to improve the question analysis problem. After finding the important words using the question analysis model, we use best matching 25 (BM25) algorithms to find relevant documents and then apply deep learning models to extract answers.

In BM25 algorithms, Okapi BM25 [25], [26] is a ranking function used by search engines to rank documents by relevance to a given query. This ranking function is based on a probabilistic model, invented in 1970-1980. This BM25 used to be called Okapi BM25 because it was used first in the Okapi search system. BM is a ranking system based on TF-IDF, its search result is based on a combination of words and ranks documents based on the query words in the document, regardless of the relationship of these words within the text content. This is also the disadvantage that BM25 encounters when in the query and the document of words with the same meaning but different spellings. Given a query Q , containing keywords q_1, \dots, q_n the BM25 score of document D is defined as (1) [25].

$$\text{score}(D, Q) = \sum_i^n \text{IDF}(q_i) \frac{f(q_i, D) \times (k_1 + 1)}{f(q_i, D) + k_1 \times (1 - b + b \times \frac{\text{fieldLen}}{\text{avgFieldLen}})} \quad (1)$$

where $\text{IDF}(q_i)$ is inverse document frequency weight of the query term q_i , D is the document, b is constant ($b=0.75$), $k_1 \in [1.2, 2.0]$, fieldLen is the length of the document, f is the frequency of occurrence of word q_i in the document, avgFieldLen is the average length of the document. Formula of $\text{IDF}(q_i)$ is defined as (2) [25].

$$\text{IDF}(q_i) = \ln \left(1 + \frac{(\text{docCount} - f(q_i) + 0.5)}{f(q_i) + 0.5} \right) \quad (2)$$

where docCount is the total number of documents and $f(q_i)$ is the number of documents contain q_i .

To improve the efficiency of the algorithm BM25, we combined the word weights extracted from the question analysis model using (1) and (2),

$$IDF(q_i) = \ln\left(1 + \frac{(docCount - f(q_i) + 0.5 + w_i)}{f(q_i) + 0.5}\right) \tag{3}$$

$$BM25 = \sum_i^n IDF(q_i) \frac{f(q_i, D) \times (k_1 + 1 + w_i)}{f(q_i, D) + k_1 \times \left(1 - b + b \times \frac{FieldLen}{avgFieldLen}\right)} \tag{4}$$

where w_i is weight of word q_i .

BERT stands for the phrase “bidirectional encoder representation from transformer”, which is a new architecture for the language processing problem, created by Google and published in 2018 by Devlin [24] and in studies [27], [28]. A distinctive feature of BERT is that it can balance the semantic context in both the left and right directions. The mechanism of BERT is transmitting all the words of the sentence simultaneously into the model, but not right-to-left or left-to-right as in other directional models. That means we can consider BERT as bidirectional. BERT defines hidden layers, size of hidden layers and the number of heads at attention as L, H, and A respectively. There are two BERT architectures. BERT base has L=12, H=768, A=12, and BERT large has L=24, H=1024, A=16.

Pre-training BERT is BERT that is trained with two tasks: masked language model and next sentence prediction. Masked language model (MLM), with a self-supervised pretraining objective, is used in many NLP for learning word representations. Before feedforward to model, approximately 15% of tokens are replaced with mask tokens, which will represent the hidden word. The model will be based on the words not covered by the mask to predict the hidden word. With that, the words around the mask will also create a context for the hidden word. To calculate the probability of the output word, we add a fully connected layer after the last layer, Softmax function will calculate the probability distribution. Next sentence prediction is a supervised classification task with two labels input. The input data for the model will be a collection of pairs of sentences such that 50% of the data consists of pairs that the selected second sentence is precisely next to the first sentence, and the rest 50% consists of pairs that the second sentence is randomly selected from the corpus with no relation to the first ones. The model’s label will correspond to *isNext*, which is the following sentence, and *notNext*, which is the pair of non-consecutive sentences. The architecture of BERT is shown in Figure 1.

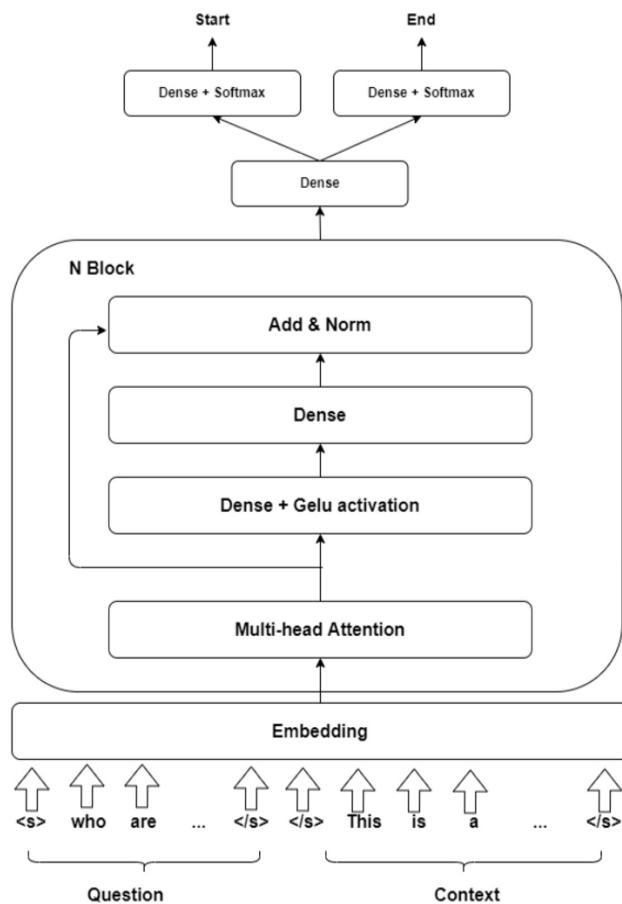


Figure 1. Architecture of BERT

The BERT model will be detailedly described in the process of building a QA system. In this paper, in addition to the BERT model, we propose more improved models of BERT, which are robustly optimized BERT pre-training approach (RoBERTa) and DistilBERT. RoBERTa [29]–[31] is improved by removing next sentence prediction during training and giving dynamic masking that will keep changing, longer training time will go with larger batch sizes. RoBERTa is proposed to improve the accuracy of the BERT model. DistilBERT [32], [33] uses the distillation technique and the Bayesian approximation algorithm as Kullback-Leibler, which approximates large neural network model architectures using networks of smaller architectures. DistilBERT has a reduced architecture compared to BERT 40%. The DistilBERT model is proposed to increase the calculation speed while preserving the accuracy and efficiency of the model.

2.2. Question answering system architecture

The question-answering model includes steps: question analysis, retrieving related documents from the dataset, and extracting the answer from relevant documents. Compared with the traditional QA models [34], we have improved the question analysis step to increase the accuracy of the weight extraction of the words in the question. The system architecture is shown in Figure 2.

To find the weight of each word in question, we focus on improving the question analysis model. The training model has an architecture as shown in Figure 3, in which the question is analyzed through two steps: pos-tag analysis and analyze the question type extracted through the deep learning model, which evaluates the weight of each word in the question.

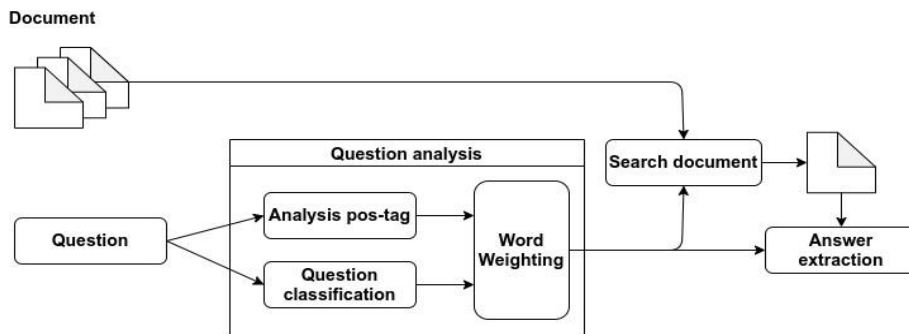


Figure 2. Model system architecture to answer questions from files and documents

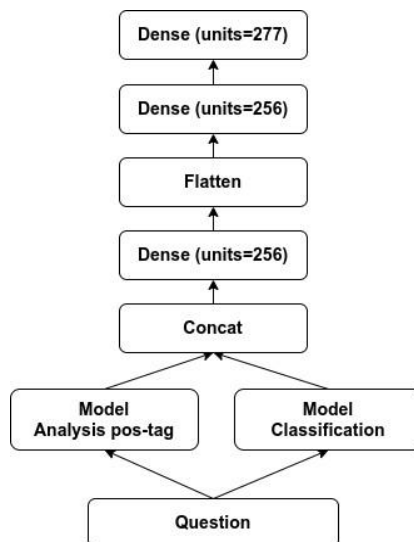


Figure 3. Weight model architecture

Pos-tag analysis, as represented in Figure 4(a), is combined in parallel from two branches. The left branch will reuse two layers of the question classification model: embedding and position embedding, which

help to reduce the model size and avoid unnecessary training. The next layer is multi-head attention with heads=6 used for finding the context vector for question. There will be a connection from position embedding to the output of multi-head attention to add previous information to avoid loss of information during attention calculation; after that is a normalization layer, and then it goes through a dropout layer with a ratio of 20%.

The last layer is a feedforward layer with two consecutively fully connected layers and the activation function (sigmoid) in the last layer for removing unimportant features when using the connection from the first layer (it contains too much irrelevant information). Formular at layer feedforward is defined as (5) and (6) [35].

$$O_1 = (x \times W_1 + b_1) \times P(X \leq x) \text{ with } P(x) \sim N(0,1) \tag{5}$$

$$O_2 = \frac{1}{1 + e^{-(o_1 \times W_2 + b_2)}} \tag{6}$$

where W_1, W_2 are weights collected in the network; b_1, b_2 are biases; x is input, P is activation function; and $N(0,1)$ is normal distribution.

The right branch will use the feedforward layer from the question with the input as a matrix where each sentence with the size of ($maxlen \times num_pos_tag$), where $maxlen$ is the maximum length of question padding with a value of 0; num_pos_tag : is several types of pos-tag including twenty-five types. Each word one-hot to vector has a value of 0 or 1, where 1 is the pos-tag position of the question in the pos-tag list. After that is multi-head attention with the value of pos-tag, query, and key, which are results of the right branch finding the relationship between the context and the pos-tag of the words, the output will be added to the output of the right branch and then passed through 20% dropout and feedforward.

The question classification, as shown in Figure 4(b), plays an important role in question analyzing. Categorize question is labeled as who, what, animal, how, number, why, time point, period, plant, yes/no, and location. The question classification layer is an external pre-trained model and is added to the question analysis model. This classification model also removed all RNN layers in favor of multi-head attention. Output after passing through the embedding layer will go through the layers in turn: Normalization, multi-head attention (heads=3), fully connected with unit=1 to remove the last dimension ($batch_size \times maxlen$), and finally through layers: fully connected (units=d), dropout (0.2), and fully connected (units=11) with the softmax activation function, giving the probability of the classified object.

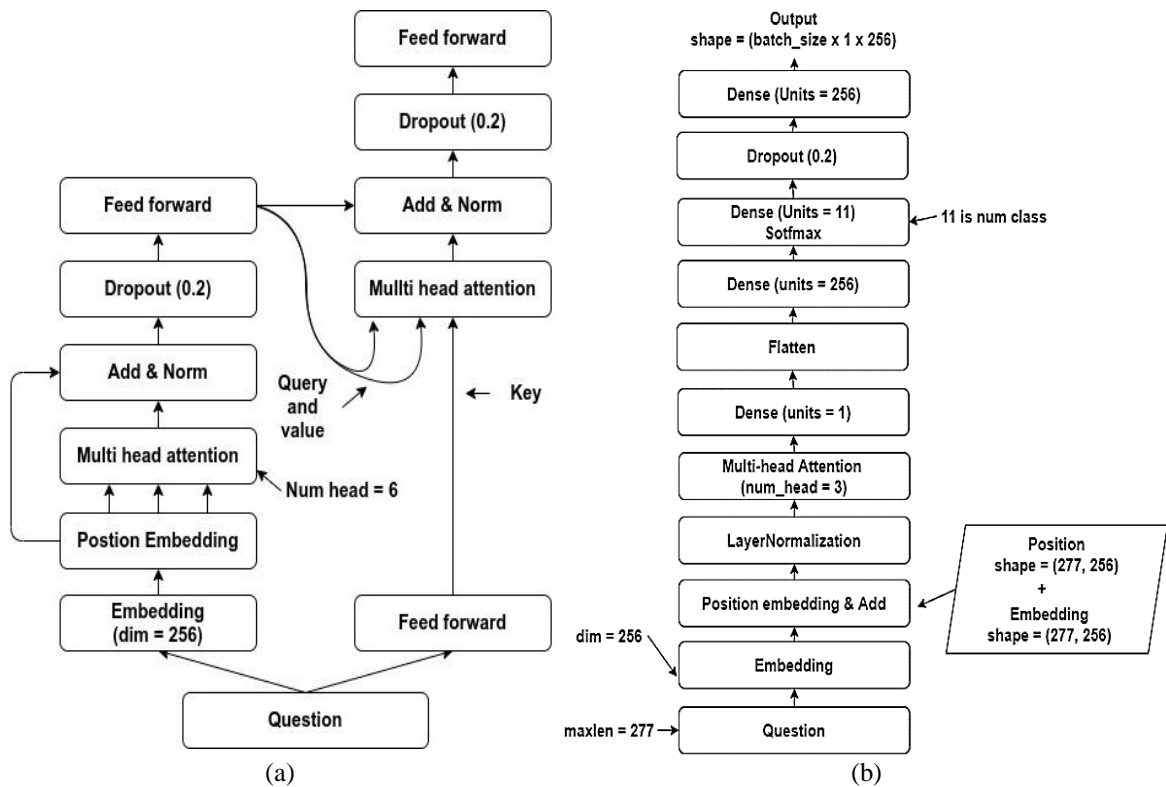


Figure 4. Weight model detail (a) pos-tag analysis model and (b) question classification model

In retrieving related documents combining word weights, the question after the processed query will be included in extracting answers or corresponding paragraphs to search for the document related to the question by using an improved BM25 algorithm. When extracting the answer, the passage after extracting from the above steps, the passage will be sent to the answer extraction model. The answer will be extracted from the passage itself. Here we will apply some BERT architectures and fine-tune them to be able to apply word weights. The fine-tuning model architecture is depicted in Figure 5 through the Roberta model. Input after converting into tokens will be put into embedding layers and converted to word vectors with a dimension of 256 and will be marked using position embedding. The question will be put into the weight model to calculate their weights to mark important words. Output at a question analysis model has a size of $batch_size \times maxlen$. To be able to multiply into the output of the Encoder, we will expand the last dimension to become the size of $batch_size \times maxlen \times embedding_dim$. The decoder includes two inputs: the first is the passage after embedding, and the other is the output from the Encoder after multiplying with word weights. The model has two outputs for determining the start and end points of the answer, and the result will be a probability vector of size (1×800) .

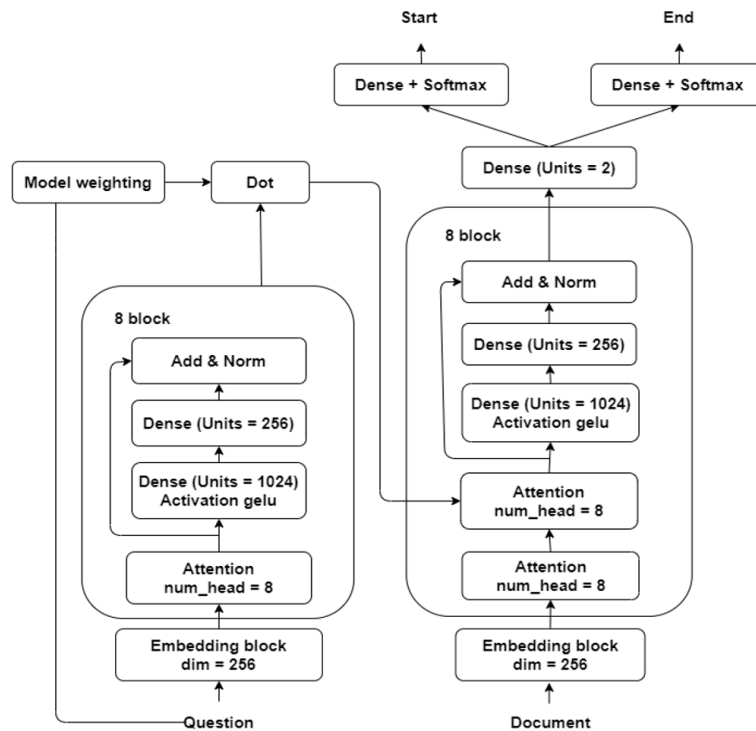


Figure 5. Fine-tune model RoBERTa

2.3. Model evaluation

We evaluate the model's effectiveness through the training time and the number of iterations to achieve accuracy. The model's accuracy is evaluated based on 2 values: F1-score and EM. The EM value is calculated according to the rule: for each pair of questions and answers, if the characters in the model's prediction sentence exactly match the characters of one of the correct answers, then $EM=1$, otherwise $EM=0$. The F1-score value is defined by (7) to (9) [36].

Precision is the ratio of the number of correctly classified correct answers to the number of the classified sentences.

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

Recall is the ratio of the number of correct answers classified correctly to the number of sentences that are correct for that answer.

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

where TP is the true positive when the predicted label and the actual label are both 1, FP is false positive when the predicted label is 1, the actual label is 0, FN is false negative when the predicted label is 0, the actual label is 1, and TN is true negatives when the predicted label and the actual label are both 0. F1-score is the harmonic mean of precision and recall calculated by (9).

$$F1 = 2 * \left(\frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}} \right) \quad (9)$$

3. RESULTS AND DISCUSSION

3.1. Dataset

The dataset for the question classification model includes many different topics collected and labeled including 19,339 questions. They are labeled as question types: who, what, animal, how, number, why, time point, period, plant, yes/no, and location. The dataset of questions is formatted according to Stanford Answering Dataset [37], which includes corresponded 21,073 questions and 21,073 answers.

The data source is built from the text files of the Industrial University of Ho Chi Minh City including credit regulations, scholarship encouraging study regulations, training regulations, student affairs, and admissions. The dataset is described in Table 1. This dataset is used to train the model for finding word weights in questions and the model for extracting answers. Data preprocessing: removing characters (“1.”, “2.”, “a”), “b”, special characters, uppercase, lowercase.

Table 1. Question answer data

Document	Question	Answer
<i>Tổ hợp 3 môn xét tuyển khối A00 toán, lý, hóa khối A01 toán, tiếng anh, vật lý khối B00 toán, hóa...</i> (The combination of 3 subjects for A00 math, physics, chemistry, A01 math, English, physics; B00 math, chemistry...)	<i>Xin chào thầy, cho em hỏi khối A00 gồm những môn gì, thầy giải đáp ạ?</i> (Hello teacher, can I ask what subjects are in A00, please answer?)	<i>Khối A00 Toán Vật lý Hóa học</i> (A00 math, physics, chemistry)
<i>Tổ hợp 3 môn xét tuyển khối A00 toán, lý, hóa khối A01 toán, tiếng anh, vật lý khối B00 toán, hóa...</i> (The combination of 3 subjects for A00 math, physics, chemistry, A01 math, English, physics, B00 math, chemistry...)	<i>Xin chào thầy, cho em hỏi khối A00 gồm những môn nào ạ, em xin cảm ơn.</i> (Hello teacher, can I ask what subjects are included in A00, thank you)	<i>Khối A00 Toán Vật lý Hóa học</i> (A00 math, physics, chemistry)
<i>Nhóm ngành công nghệ thông tin gồm 05 ngành: chuyên ngành công nghệ thông tin, kỹ thuật phần mềm, khoa học máy tính, hệ thống thông tin, khoa học dữ liệu...</i> (The group of information technology industries includes 05 majors: information technology, software engineering, computer science, information system, data science, ...)	<i>Bạn vui lòng cho tôi biết những tổ hợp nào được xét cho chuyên ngành khoa học máy tính</i> (Could you please tell me what combinations are considered for the computer science major?)	<i>Nhóm ngành công nghệ thông tin gồm 05 ngành: chuyên ngành công nghệ thông tin, kỹ thuật phần mềm...</i> (The group of information technology industries includes 05 majors: information technology, software engineering...)

3.2. Results and analysis

Result of the classification model: The model is trained on the training set of 17405 questions and a test set of 1,934 questions, which is trained using Python with the TensorFlow library. The training process has been done using Google Colab Pro, GPU Tesla P100-PCIE-16 GB, and RAM 12.6 GB. The model is evaluated based on accuracy and F1-score and compared with the model using bidirectional LSTM (Bi-LSTM) with the same model parameters. The results in Table 2 show that the model using multi-head attention has high efficiency in terms of training speed and accuracy compared to the model using Bi-LSTM.

Result of weighting model: The model is trained with the IUH question data set. Important words will be labeled and set weight from 0.8 and 0.95, unimportant words will be labeled and set weight between 0.1 to 0.3. The results show that important words are evaluated with high accuracy. When it is compared with the pos-tags assigned word from the pyvi library in Tables 3 and 4; it shows that the high weights are equivalent to the words being nouns, verbs, and adjectives. Furthermore, the value of word weight indicates the importance of that word in the sentence, thereby extracting better answers.

Table 2. Question classification model comparison table

Model	Epochs	Time per epoch	Evaluate loss	Evaluate accuracy	F1-score
Bi-LSTM	30	33 s	0.42	0.88	0.85
Multi-head attention	30	18 s	0.46	0.84	0.83

Table 3. Compare the weight model with the pyvi library

Question	weight	Pos-tag pyvi
<i>Quy chế</i> (Regulation)	0.86	<i>Danh từ</i> (Noun)
<i>Đào tạo</i> (educate)	0.31	<i>Động từ</i> (Verb)
<i>Của</i> (of)	0.26	<i>Giới từ</i> (Preposition)
<i>Hệ thống</i> (system)	0.73	<i>Danh từ</i> (Noun)
<i>Tín chỉ</i> (credit)	0.82	<i>Danh từ</i> (Noun)
<i>Là gì</i> (what is)	0.21	<i>Đại từ</i> (Pronoun)

Table 4. Compare the weight model with the pyvi library

Question	weight	Pos-tag pyvi
<i>Trách nhiệm</i> (responsibilities)	0.91	<i>Danh từ</i> (Noun)
<i>Của</i> (of)	0.21	<i>Giới từ</i> (Preposition)
<i>Phòng</i> (Department)	0.84	<i>Danh từ</i> (Noun)
<i>Công tác</i> (activities)	0.72	<i>Động từ</i> (Verb)
<i>Sinh viên</i> (student)	0.88	<i>Danh từ</i> (Noun)
<i>Là gì</i> (what are)	0.11	<i>Đại từ</i> (Pronoun)

For the question: “*Quy chế đào tạo theo hệ thống tín chỉ là gì? (What is the education regulation using credit system?)*”, word weights are shown in Table 3. For the question “*Trách nhiệm của phòng công tác sinh viên là gì?*” (*What are the responsibilities of the student activities Department?*), the result is shown in Table 4. We use Nguyen’s dataset [38], [39] including 18,108 questions and paragraphs for the improved BM25 algorithm. The results of comparisons between the two models are shown in Table 5. We can see that the improved BM25 algorithm has higher accuracy.

Result of QA model: The pre-trained model is trained with the IUH QA dataset with a training set of more than 17,405 QA data and a test set of 1,600 QA data. The model is built in Python language and implemented on the Google Collab Pro platform, GPU Tesla P100-PCIe-16 GB, RAM 12.6 GB. The model architecture is fine-tuned with the parameters shown in Table 6.

The results of training on different models and evaluating the accuracy are shown in Table 7. We evaluate the model’s effectiveness through the training time and the number of iterations to achieve accuracy. The model’s accuracy is evaluated based on two values, F1-score, and EM. Results show that the RoBERTa model has the best results in terms of both execution speed and accuracy.

Table 5. Comparison table of improved BM25 and BM25 techniques

	Accuracy (%)	F1-score (%)
BM25	83.34	90.92
BM25 improved	85.87	92.40

Table 6. Model architecture parameters

Model	Size of hidden layers (H)	Number of multi-head layers (A)	Number of blocks in transformer (L)
BERT	512	8	8
BERT + weight	256	8	8
RoBERTa	256	8	8
RoBERTa + weight	256	8	8
DistilBERT	256	8	8
DistilBERT + weight	256	8	8

Table 7. Modeling efficiency and accuracy

Model	Time per Epoch	Epoch	Batch size	F1-Score (%)	EM (%)
BERT	812s	20	8	65	61
BERT + weight	782s	20	16	69	65
RoBERTa	845s	20	16	74	74
RoBERTa + weight	922s	30	16	75	73
DistilBERT	892s	20	8	72	71
DistilBERT + weight	1027s	20	16	62	58

We chose RoBERTa+weight model for the experiment. Some results are obtained.

- For the question: “*Trách nhiệm của phòng tổ chức hành chính là gì?*” (*What are the responsibilities of the administrative department?*). Corresponding weight value: “*Trách nhiệm*”: 0.87, “*của*”: 0.24, “*phòng*”: 0.86, “*tổ chức hành chính*”: 0.59, “*là gì*”: 0.11. Relevant paragraph: “*Trách nhiệm của Phòng Tổ chức Hành chính. 1. Giải quyết các công việc hành chính có liên quan cho sinh viên; sao y các văn bản, chứng chỉ của Trường. 2. Tiếp nhận và phân phối bưu phẩm, thư cho sinh viên.*” and the answer is: “*1. Giải quyết các công việc hành chính có liên quan cho sinh viên; sao y các văn bản, chứng chỉ của Trường. 2. Tiếp nhận và phân phối bưu phẩm, thư cho sinh viên*” (*1. Solve related administrative problems for students as per copy, notarize all student’s documents from university. 2. Receive and distribute mail and letters to students.*).
- For the question: “*Yêu cầu nhiệm vụ của sinh viên là gì?*” (*What are the student tasks?*). Corresponding weight value: “*Yêu cầu*”: 0.18, “*nhiệm vụ*”: 0.40, “*của*”: 0.31, “*sinh viên*”: 0.85, “*là gì*”: 0.18. Relevant paragraph: “*Yêu cầu của công tác sinh viên. 1. Sinh viên là nhân vật trung tâm trong Nhà trường, được Nhà trường bảo đảm điều kiện thực hiện đầy đủ nhiệm vụ và quyền trong quá trình học tập và rèn luyện tại Trường. 2. Công tác sinh viên phải thực hiện đúng đường lối, chính sách của Đảng, pháp luật của Nhà*”

nước và các quy chế, quy định của Bộ Giáo dục và Đào tạo. Công tác sinh viên phải bảo đảm dân chủ, khách quan, công bằng, công khai, minh bạch trong các vấn đề có liên quan đến sinh viên.” and the answer is: “1. Sinh viên là nhân vật trung tâm trong Nhà trường, được Nhà trường bảo đảm điều kiện thực hiện đầy đủ nhiệm vụ và quyền trong quá trình học tập và rèn luyện tại Trường. 2, Công tác sinh viên phải thực hiện đúng đường lối, chính sách của Đảng, pháp luật của Nhà nước và các quy chế, quy định của Bộ Giáo dục và Đào tạo. Công tác sinh viên phải bảo đảm dân chủ, khách quan, công bằng, công khai, minh bạch trong các vấn đề có liên quan đến sinh viên.” (1 student is a central object in the University and University guarantees the conditions to fully perform all the duties and rights for Student's studying and training process at the University 2 the student's work must be done to comply with the Party's policies, laws of the state and regulations of the ministry of education and training, student work must ensure democracy, objectivity, fairness, publicity, transparency).

- For the question: “Tổ hợp xét tuyển ngành kỹ thuật ô tô tại thành phố Hồ Chí Minh như thế nào?” (How are the combinations for the automotive engineering industry in Ho Chi Minh City?). Corresponding weight value: “Tổ hợp”: 0.73, “xét tuyển”: 0.50, “ngành”: 0.64, “kỹ thuật”: 0.82, “ô tô”: 0.88, “tại”: 0.36, “thành phố”: 0.78, “Hồ Chí Minh”: 0.36, “như thế nào”: 0.17. Relevant paragraph: “Tên ngành công nghệ kỹ thuật ô tô mã ngành đối với hệ đại trà 7510205 các tổ hợp xét tuyển A00, A01, C01, D90” and the answer is: “tên ngành công nghệ kỹ thuật ô tô mã ngành đối với hệ đại trà 7510205 các tổ hợp xét tuyển A00, A01s, C01, D90” (The Major is Automotive Engineering Technology, the Major's code for Normal Student Group is 7510205; selection combinations are A00, A01, C01, D90).
- For the question: “Quy chế đào tạo của trường đại học công nghiệp thành phố Hồ Chí Minh đến năm 2020 như thế nào” (What is the training regulation of the Industrial University of Ho Chi Minh City until 2020?). Corresponding weight value: “Quy chế”: 0.76, “đào tạo”: 0.63, “của”: 0.48, “trường”: 0.89, “đại học”: 0.83, “công nghiệp”: 0.84, “thành phố”: 0.61, “Hồ Chí Minh”: 0.50, “đến”: 0.43, “năm”: 0.65, “2020”: 0.60. “như thế nào”: 0.38. Relevant paragraph: “Quy chế đào tạo theo hệ thống tín chỉ là tập hợp những quy định về phương thức đào tạo thực hiện theo hình thức tích lũy tín chỉ; trong đó sinh viên chủ động lựa chọn học từng học phần (tuân theo một số ràng buộc được quy định trước) nhằm tích lũy từng phần kiến thức và tiến tới hoàn thành toàn bộ chương trình đào tạo để được cấp văn bằng tốt nghiệp.” and the answer is: “đào tạo theo hệ thống tín chỉ” (Training is using credit system).

4. CONCLUSION




The question analyzing process in the QA problem is a significant step, in this paper, we improve the problem of QA from the text data by adding steps of contextual feature analysis, pos-tag analysis, and question type analysis in the question analysis model. These steps are built using a deep learning network model to search for high-weighted words in questions. Our models are trained on real data sets and achieve reliable accuracy. The weighted results are applied to improve the BM25 algorithm and then applied to the QA model to find the most suitable answer. Compared with the previous method using only TF-IDF weights, the combination of word weights in the BM25 algorithm improves the accuracy of the paragraph's search results. At the same time, the combination of word weights in the step of finding an answer also helps to eliminate the noise and returns the correct answer. The results show that for questions with more semantic complexity, the weight model has given high weight to important words, increasing the accuracy of the BM25 algorithm and the sentence position prediction model, answer. For questions that are too long and complex, the extraction model is redundant or omits important words. In the upcoming development, we will add data with more diverse topics and improve the algorithm to make the model more accurate and efficient.

REFERENCES




- [1] K. S. D. Ishwari, A. K. R. R. Aneze, S. Sudheesan, H. J. D. A. Karunaratne, A. Nugaliyadde, and Y. Mallawarachchi, “Advances in natural language question answering: A review,” *Prepr. arXiv1904.05276*, Apr. 2019.
- [2] D. Metzler and W. B. Croft, “Analysis of statistical question classification for fact-based questions,” *Information Retrieval*, vol. 8, no. 3, pp. 481–504, Jan. 2005, doi: 10.1007/s10791-005-6995-3.
- [3] H. T. Madabushi and M. Lee, “High accuracy rule-based question classification using question syntax and semantics,” in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2016, pp. 1220–1230.
- [4] B. Loni, G. van Tulder, P. Wiggers, D. M. J. Tax, and M. Loog, “Question classification by weighted combination of lexical, syntactic and semantic features,” in *Text, Speech and Dialogue*, 2011, pp. 243–250, doi: 10.1007/978-3-642-23538-2_31.
- [5] E. Riloff and M. Thelen, “A rule-based question answering system for reading comprehension tests,” in *NLP/NAACL 2000 Workshop on Reading comprehension tests as evaluation for computer-based language understanding systems*, 2000, vol. 6, pp. 13–19, doi: 10.3115/1117595.1117598.
- [6] M. Mishra, V. K. Mishra, and S. H.R., “Question classification using semantic, syntactic and lexical features,” *International journal of Web & Semantic Technology*, vol. 4, no. 3, pp. 39–47, Jul. 2013, doi: 10.5121/ijwest.2013.4304.
- [7] P. Biswas, A. Sharan, and R. Kumar, “Question classification using syntactic and rule based approach,” in *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Sep. 2014, pp. 1033–1038, doi: 10.1109/ICACCI.2014.6968434.

- [8] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of Machine Learning Research*, vol. 3, pp. 1137–1155, Apr. 2003.
- [9] Z. Alami Merrouni, B. Frikh, and B. Ouhbi, "Automatic keyphrase extraction: a survey and trends," *Journal of Intelligent Information Systems*, vol. 54, no. 2, pp. 391–424, Apr. 2020, doi: 10.1007/s10844-019-00558-9.
- [10] R. Arora, P. Singh, H. Goyal, S. Singhal, and S. Vijayvargiya, "Comparative question answering system based on natural language processing and machine learning," in *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, Mar. 2021, pp. 373–378, doi: 10.1109/ICAIS50930.2021.9396015.
- [11] S. K. Mishra, P. Kumar, and S. K. Saha, "A support vector machine based system for technical question classification," in *Proceedings of the Third International Conference on Mining Intelligence and Knowledge Exploration*, 2015, pp. 640–649, doi: 10.1007/978-3-319-26832-3_60.
- [12] X.-P. Yu and X.-G. Yu, "Novel text classification based on K-nearest neighbor," in *2007 International Conference on Machine Learning and Cybernetics*, 2007, pp. 3425–3430, doi: 10.1109/ICMLC.2007.4370740.
- [13] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Machine Learning*, vol. 29, pp. 131–163, 1997, doi: 10.1023/A:1007465528199.
- [14] T. Fei, W. J. Heng, K. C. Toh, and Tian Qi, "Question classification for E-learning by artificial neural network," in *Fourth International Conference on Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint*, 2003, vol. 3, pp. 1757–1761, doi: 10.1109/ICICS.2003.1292768.
- [15] S. Zhou, "Research on design of automatic question answering system based on convolutional neural network," in *2020 2nd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, Oct. 2020, pp. 204–207, doi: 10.1109/MLBDBI51377.2020.00045.
- [16] S. Yilmaz and S. Toklu, "A deep learning analysis on question classification task using Word2vec representations," *Neural Computing and Applications*, vol. 32, no. 7, pp. 2909–2928, Apr. 2020, doi: 10.1007/s00521-020-04725-w.
- [17] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
- [18] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *Prepr. arXiv1412.3555*, Dec. 2014.
- [19] S. V. Moravvej, M. J. M. Kahaki, M. S. Sartakhti, and A. Mirzaei, "A method based on attention mechanism using bidirectional long-short term memory (BLSTM) for question answering," in *2021 29th Iranian Conference on Electrical Engineering (ICEE)*, May 2021, pp. 460–464, doi: 10.1109/ICEE52715.2021.9544258.
- [20] W. Huang, X. Dong, W. Shang, and W. Lin, "Research on man-machine conversation system based on GRU seq2seq model," in *2019 6th International Conference on Dependable Systems and Their Applications (DSA)*, Jan. 2020, pp. 413–418, doi: 10.1109/DSA.2019.00064.
- [21] A. Vaswani *et al.*, "Attention is all you need," *Prepr. arXiv1706.03762*, Jun. 2017.
- [22] S. Modak, S. Chaudhury, A. Rawat, and S. Deb, "Improving performance of recurrent neural networks for question-answering with attention-based context reduction," in *2021 IEEE Mysore Sub Section International Conference (MysuruCon)*, Oct. 2021, pp. 723–728, doi: 10.1109/MysuruCon52639.2021.9641626.
- [23] T. Shao, Y. Guo, H. Chen, and Z. Hao, "Transformer-based neural network for answer selection in question answering," *IEEE Access*, vol. 7, pp. 26146–26156, 2019, doi: 10.1109/ACCESS.2019.2900753.
- [24] J. Devlin, M.-W. 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*, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.
- [25] S. Robertson and H. Zaragoza, "The probabilistic relevance framework: BM25 and beyond," *Foundations and Trends® in Information Retrieval*, vol. 3, no. 4, pp. 333–389, 2009, doi: 10.1561/15000000019.
- [26] A. Trotman, A. Puurula, and B. Burgess, "Improvements to BM25 and language models examined," in *Proceedings of the 2014 Australasian Document Computing Symposium on - ADCS '14*, 2014, pp. 58–65, doi: 10.1145/2682862.2682863.
- [27] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, "Transformers in vision: A survey," *ACM Computing Surveys*, vol. 54, no. 10s, pp. 1–41, Jan. 2022, doi: 10.1145/3505244.
- [28] D. A. Navastara, I. I. I. Anand, and A. Z. Arifin, "Bilingual question answering system using bidirectional encoder representations from transformers and best matching method," in *2021 13th International Conference on Information & Communication Technology and System (ICTS)*, Oct. 2021, pp. 360–364, doi: 10.1109/ICTS52701.2021.9608905.
- [29] Y. Liu *et al.*, "RoBERTa: A robustly optimized BERT pretraining approach," *Prepr. arXiv1907.11692*, Jul. 2019.
- [30] W. Suwarningsih, R. A. Pratama, F. Yusuf Rahadika, and M. H. Albar Purnomo, "Self-attention mechanism of RoBERTa to improve QAS for e-health education," in *2021 4th International Conference of Computer and Informatics Engineering (IC2IE)*, Sep. 2021, pp. 221–225, doi: 10.1109/IC2IE53219.2021.9649363.
- [31] K. L. Tan, C. P. Lee, K. S. M. Anbananthen, and K. M. Lim, "RoBERTa-LSTM: A hybrid model for sentiment analysis with transformer and recurrent neural network," *IEEE Access*, vol. 10, pp. 21517–21525, 2022, doi: 10.1109/ACCESS.2022.3152828.
- [32] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *Prepr. arXiv1910.01108*, Oct. 2019.
- [33] A. F. Adoma, N.-M. Henry, and W. Chen, "Comparative analyses of bert, roberta, distilbert, and xlnet for text-based emotion recognition," in *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, Dec. 2020, pp. 117–121, doi: 10.1109/ICCWAMTIP51612.2020.9317379.
- [34] H. A. Pandya and B. S. Bhatt, "Question answering survey: directions, challenges, datasets, evaluation matrices," *Prepr. arXiv2112.03572*, Dec. 2021.
- [35] G. Bebis and M. Georgiopoulos, "Feed-forward neural networks," *IEEE Potentials*, vol. 13, no. 4, pp. 27–31, Oct. 1994, doi: 10.1109/45.329294.
- [36] C. Goutte and E. Gaussier, "A probabilistic interpretation of precision, recall and F-score, with implication for evaluation," in *ECIR 2005: Advances in Information Retrieval*, 2005, pp. 345–359, doi: 10.1007/978-3-540-31865-1_25.
- [37] 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, doi: 10.18653/v1/D16-1264.
- [38] K. Nguyen, V. Nguyen, A. Nguyen, and N. Nguyen, "A vietnamese dataset for evaluating machine reading comprehension," in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 2595–2605, doi: 10.18653/v1/2020.coling-main.233.
- [39] D. T. Phuc, T. Q. Trieu, N. Van Tinh, and D. S. Hieu, "Video captioning in Vietnamese using deep learning," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 3, pp. 3092–3103, Jun. 2022, doi: 10.11591/ijece.v12i3.pp3092-3103.




BIOGRAPHIES OF AUTHORS

Dang Thi Phuc    received a specialist degree from the Lomonosov Moscow State University, Moscow, Russia in 2008. In 2018, she received a Ph.D. degree in system analysis, control, and information processing from Peoples' Friendship University of Russia, Moscow, Russia. Since 2018, she is a lecturer at Faculty of Information Technology, Industry University of Ho Chi Minh City, Vietnam. Her research interests include machine learning, computer vision, NLP, deep learning, and operator network. She can be contacted at email: phucdt@iuh.edu.vn.






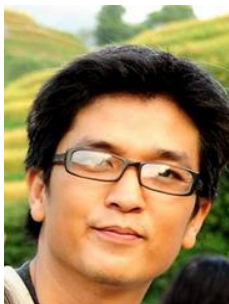
Dang Van Nghiem    is a student at Industrial University of Ho Chi Minh City, Vietnam. His study interests include machine learning, computer vision, NLP, and deep learning. He can be contacted at email: vanngkiem848@gmail.com.






Bui Binh Minh    is a student at Industrial University of Ho Chi Minh City, Vietnam. His study interests include machine learning, computer vision, NLP, and deep learning. He can be contacted at email: buiminhh.k14@gmail.com.



Tran My Linh    is a student at Industrial University of Ho Chi Minh City, Vietnam. Her study interests include machine learning, computer vision, NLP, and deep learning. She can be contacted at email: tranmylinh26042000@gmail.com.



Dau Sy Hieu    received a specialist degree from the Lomonosov Moscow State University, Moscow, Russia in 2009. In 2015 he received a Ph.D. degree in Physical Condensation State from Peoples' Friendship University of Russia, Moscow, Russia. Since 2009, he is a lecturer Faculty of Applied Science, University of Technology-Viet Nam National University HCM City, Vietnam. His research interests include condensation state, optical system design, machine learning, computer vision, NLP, and deep learning. He can be contacted at email: dausyhiu@hcmut.edu.vn.