AN APPROACH FOR BUILDING A CHATBOT SYSTEM FOR THE ADMISSION PROCESS OF DA LAT UNIVERSITY

Phan Thi Thanh Nga, Nguyen Thi Luong, Ta Hoang Thang, Thai Duy Quy*
Dalat University

ARTICLE INFO

Received: 25/5/2022
Revised: 22/8/2022
Published: 23/8/2022

KEYWORDS

Chatbot
Virtual assistant
BERT
University admission
Representation language models

ABSTRACT

A chatbot is a computer program designed for the chat interaction between robots and humans automatically using natural language techniques. This program is developed to be a virtual assistant which lures the human into the thought that they are talking to a real person. In this paper, we develop a chatbot system for the admission process of Da Lat university that allows the staff to answer questions immediately and automatically from users anytime. The important feature of any chatbot is to understand the user's questions and to respond with appropriate answers. Our approach builds a chatbot application that adapts to the university's needs. We apply some BERT-based representation language models to predict the answer from the input question. The experimental results show that the salti/bert-base-multilingual-cased-finetuned-squad is a suitable model for our chatbot application since its F1 and EM scores for dev_set are remarkably high, accounting for 88.6% and 79.6%, respectively. For the intent classification, we achieve the accuracies of 99.9% and 100% for the validate accuracy and the test accuracy.

MỘT CÁCH TIẾP CẬN XÂY DỰNG ỨNG DỤNG CHATBOT TƯ VẤN TUYỂN SINH TRƯỜNG ĐẠI HỌC ĐÀ LẠT

Phan Thị Thanh Nga, Nguyễn Thị Lương, Tạ Hoàng Thắng, Thái Duy Quý*
Trường Đại học Đà Lạt

THÔNG TIN BÀI BÁO

Ngày nhận bài: 25/5/2022
Ngày hoàn thiện: 22/8/2022
Ngày đăng: 23/8/2022

TỪ KHÓA

Chatbot
Trợ lý ảo
BERT
Tuyển sinh đại học
Mô hình biểu diễn ngôn ngữ

TÓM TẮT

Chatbot là ứng dụng được xây dựng nhằm tương tác với con người một cách tự động bằng cách sử dụng các kỹ thuật ngôn ngữ tự nhiên. Chương trình này đóng vai trò như một trợ lý ảo, trò chuyện với con người và khiến họ nghĩ rằng họ đang nói chuyện với một người thật. Trong bài báo này, chúng tôi sẽ phát triển một hệ thống chatbot hỗ trợ quy trình tuyển sinh đại học, tự động trả lời ngay lập tức tất cả các câu hỏi từ người dùng bất cứ lúc nào, ngay cả ngoài giờ hành chính. Tính năng quan trọng của một ứng dụng chatbot là hiểu câu hỏi của người dùng và đưa ra câu trả lời thích hợp. Vì vậy, chúng tôi đề xuất phương pháp xây dựng ứng dụng chatbot phù hợp với nhu cầu của trường đại học. Chúng tôi áp dụng một số mô hình biểu diễn ngôn ngữ dựa trên BERT để dự đoán câu trả lời từ câu hỏi đầu vào. Thử nghiệm cho thấy salti/ bert-base-multilingual-cased-finetuned-team là mô hình phù hợp cho ứng dụng chatbot của chúng tôi vì điểm F1 và EM trên tập dữ liệu thử nghiệm cao đáng kể, lần lượt chiếm 88,6% và 79,6%. Đối với chức năng phân lớp ý định, chúng tôi đạt được 99,9% và 100% trên tập dữ liệu thử nghiệm và tập dữ liệu kiểm tra.

DOI: https://doi.org/10.34238/tnu-jst.6056

* Corresponding author. *Email: quytd@dlu.edu.vn*

1. Introduction

Thanks to the advance in natural language processing and machine learning techniques, the usage of intelligent chatbots has become more and more popular in many organizations in recent years. The key benefit of using chatbots is to offer the ability to be active 24/7 and respond to user requests automatically and immediately. Chatbots also help to diminish the dependence on manpower in today's world of automation. In addition, chatbots may bring out higher work performance than humans when dealing with multiple conversations coincidentally. Nowadays, due to the speedy development of advanced technologies, more intelligent systems have appeared using complex knowledge-based models or artificial intelligence (AI) algorithms to understand the questions and give expected responses [1], [2]. Because chatbots bring a lot of benefits to an organization, we want to develop an intelligent virtual assistant chatbot for the admission process of Da Lat university by making use of some pre-trained BERT models, which can be finetuned with just one additional output layer to create state-of-the-art models for a wide range of NLP tasks, including question answering.

1.1. Motivation

These days, when the competition of attracting new students between universities is becoming more and more intense, thus the task of university admission consulting becomes one of the most important tasks. Therefore, if a university utilizes its chatbot application as a virtual assistant to give university admission information to interested people instantly, the number of enrolled students may be increased. However, although modern chatbots utilize the power of artificial intelligence to understand and respond to users' complex questions, the current state-of-the-art systems are still a long way from being able to have coherent, contextual, and natural conversations with humans [1]. BERT is proved to be conceptually simple and empirically powerful because it obtains new state-of-the-art results on eleven natural language processing tasks [3] so we decide to make use of a the pre-trained BERT model from huggingface.co to build our intelligent chatbot.

1.2. Literature Survey

1.2.1. Chatbot

Chatbot technology is not a new topic. It was started in 1966 when the first chatbot program named ELIZA was published. Hussain et al. [1] proposed that chatbot programs could be classified based on their Interaction Mode such as Text-based or Voice/Speech-based, Chatbot Application such as Task-Oriented or Non- Task-Oriented, Rule-based or AI, and Domain-specific or Open-Domain. There are also multiple techniques employed for building a chatbot program including Parsing, Pattern Matching, AIML, Chatscript, Ontologies, and Markov Chain Model. In addition, Artificial Neural Networks Models such as Recurrent Neural Networks (RNNs), Sequence to Sequence Neural Model, and Long Short-Term Memory Networks (LSTMs) are the latest advances in machine learning which have made it possible to develop more intelligent chatbots. As a result, many chatbot applications for the educational domain have also employed a variety of techniques which are mentioned above. First, Artificial Intelligence Markup Language (AIML) and Latent Semantic Analysis (LSA) are utilized to design a chatbot, which provides an efficient and accurate answer for any query based on the dataset of FAQs [4] Second, Pattern matching, Artificial intelligence, and machine learning are also the best choices for an automatic response giving system which will give a reply to the student's questions [5]. Third, in [6], the authors have presented a design of a textual communication application namely chatbot in the educational domain which had an average F-score of 0.870. They used an ensemble learning approach known as random forest or random decision forest. In the domain of Vietnamese chatbot, Nguyen and Truong [7] proposed a solution to build a semi-automatic

consultancy system (a semi-automatic question-answering system) using VnTokenizer for word separation and stop word removing and the SVM model for question classification. Moreover, Bao [8] proposed a model for building a question-answering system using the seq2seq model and LSTM in his master thesis. Besides, Thuy [9] proposed a chatbot application that answered automatically all questions about the services supported by Vietnam Airlines. The author utilized BoW and TF-IDF (Term Frequency – Inverse Document Frequency) to create word vectors, used a multi-class SVM algorithm for classification, and got a precision of about 87.5%. Finally, the authors in [10] presented a healthcare-supporting chatbot that utilized the DIET model for classifying the user's intents, and achieved a precision score of 95%.

1.2.2. Embeddings: Word2Vec, Skip-gram, GloVe

As we know the machine learning models cannot process text so we need to figure out a way to convert these textual data into numerical data. Word embeddings are a form of word representation that bridges the human understanding of language to that of a machine. They have learned representations of text in an n-dimensional space where words that have the same meaning have a similar representation. Word2vec and GloVe are the two most popular algorithms for word embeddings that bring out the semantic similarity of words. However, Word2vec and Glove are different because Word2vec embeddings are based on training a shallow feedforward neural network while GloVe embeddings are learned based on matrix factorization techniques. Word2vec is a method to efficiently create word embeddings by using a two-layer neural network. This simple neural network was developed in [11] and [12] as a response to make the neural-network-based training of the embedding more efficient and since then has become the de facto standard for developing pre-trained word embedding. Word2vec is not a single algorithm but a combination of two techniques – CBOW (Continuous bag of words) and Skip-gram model [11]. Skip-gram model is an efficient method for learning high-quality vector representations of words from large amounts of unstructured text data. The Skip-gram architecture is similar to CBOW, but instead of predicting the current word based on the context, it tries to maximize the classification of a word based on another word in the same sentence. GloVe stands for Global Vectors because the global corpus statistics are captured directly by the model. The result by Pennington, Socher, and Manning [13] shows that GloVe is a new global log-bilinear regression model for the unsupervised learning of word representations that outperforms other models on word analogy, word similarity, and named entity recognition tasks.

1.2.3. BERT

BERT (Bidirectional Encoder Representations from Transformers) was published by Google in 2018 and has recently achieved great performance in a wide range of NLP tasks, including question answering and language inference. BERT is proved to be conceptually simple and empirically powerful because it obtains new state-of-the-art results on eleven natural language processing tasks [3]. The difference between BERT and other previous language representation models is that it is designed to pre-train deep bidirectional representations from the unlabeled text by jointly conditioning on both left and right context in all layers then the pre-trained BERT model can be finetuned with just one additional output layer to create state-of-the-art models for a wide range of NLP tasks, such as question answering and language inference, without substantial task-specific architecture modifications [3]. In detail, BERT is first pre-trained using two unsupervised tasks including Masked LM and Next Sentence Prediction (NSP). After that, a pre-trained BERT model can be finetuned to model many downstream tasks whether they involve single text or text pairs.

The model is trained on unlabeled data over different pre-training tasks in pre-training steps. In contrast, in finetuning steps, the BERT model is first initialized with the pre-trained parameters, and all of the parameters are fine-tuned using labeled data from the downstream

tasks. These two steps are presented in Figure 1 by running an example of question-answering. The BERT model architecture is a multi-layer bidirectional transformer encoder, and it is discussed in detail in [3].

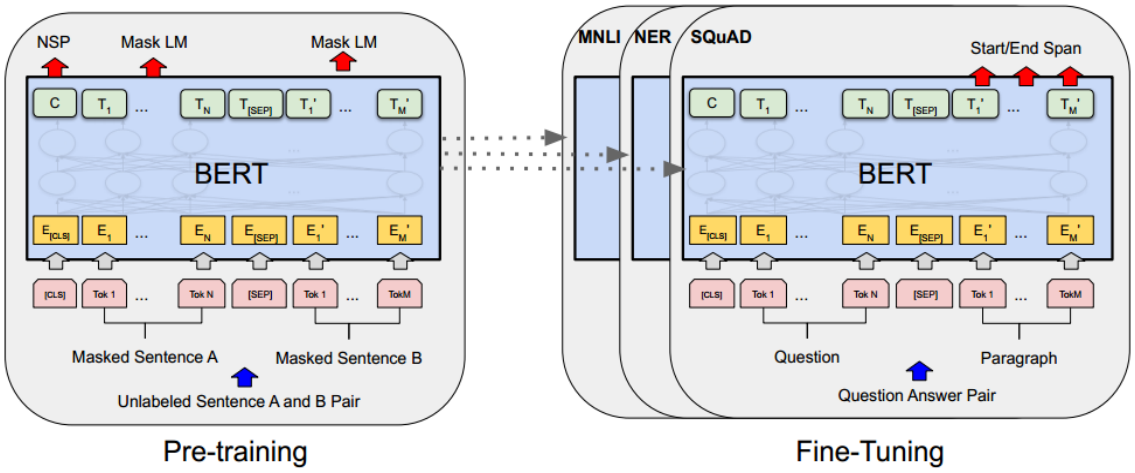

Figure 1. *Overall pre-training and fine-tuning procedures for BERT* [3]

To make BERT handle different downstream tasks, a "sequence" is used as the input token sequence to BERT, which may be a single sentence or two sentences packed together. The input representation of each token is the sum of tokens, segments, and position embeddings. A visualization of this construction can be seen in Figure 2. The reading comprehension question answering system aims to answer questions given passage or document. For this reason, each input question and paragraph must be represented as a single packed sequence (question with A segmentation embeddings and paragraph with B segmentation embeddings).

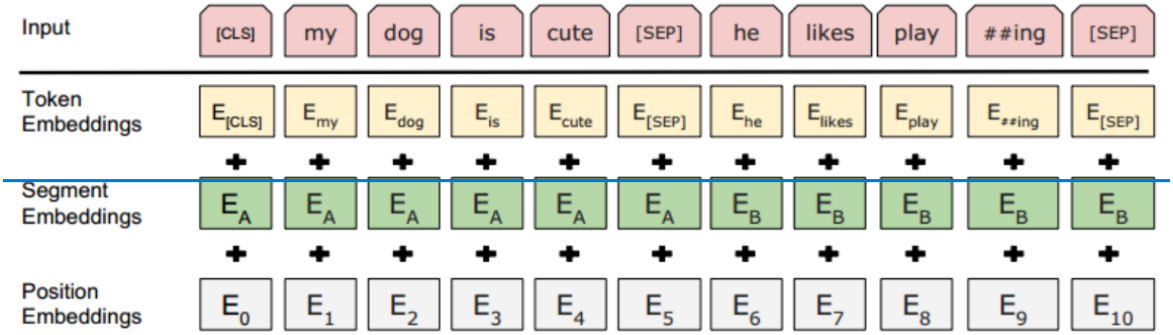

Figure 2. *BERT input representation* [3]

Due to the (BERT) has recently achieved state-of-the-art performance in question answering tasks, in this paper, we will pick up some pre-trained BERT models from huggingface.co, which can support multilingual, including Vietnamese to find which models are suitable for our chatbot application.

1.3. Our approach

To have an appropriate dataset for all pre-trained BERT-based models, three main steps have been taken which are data collection, data preprocessing, and data standardization. Firstly, and most importantly, the dataset of question-and-answer pairs about university admission must be collected. We have collected about 800 questions and their corresponding answers. In the following step, we apply some techniques for normalizing and standardizing those questions such

as cleaning data, adding Vietnamese accents, finding and replacing all the abbreviations with normal form words, and removing stop words. After this step, we prepared around 800 pairs of questions and answers for our experiment. This preprocessing step can be considered as a fine-tune to enhance the learning performance. Next, we utilize word embedding algorithms to automatically create the contexts for all pairs of questions and answers. Besides pairs of question and answer, the paragraphs or the contexts that contain the information for the answer to each question are also provided as input. We create the contexts from the university's brochure or by joining several answers which have similar semantic meanings. To find this group of these answers we utilize embedding algorithms such as Word2vec or GloVe.

The contribution of the paper is to propose the process of data collection, data preprocessing, and data standardization. Besides, we also examined several popular pre-trained BERT-based language models for Vietnamese QA systems to find the most suitable model for our application. Except for this section, the overall structure of the paper is as follows. Section 2 and Section 3 describe our approach and experiment. Section 4 describes the concluding remarks and future scope of the research.

2. Proposed Methodology

In this paper, we use a method with two phases. In the first phase, we performed some steps in Figure 3 to train data for the chatbot such as data preparation, data preprocessing, creating a context for a question, and training with BERT models in Figure 3. In the second phase, we use the model in the first phase to predict an answer to a question in Figure 4.

2.1. Training phase

2.1.1. Data preparation

Data labeling is an essential step in building an automated chatbot application, in which we could present a question-answering system. We initially collected about 250 conversations from Da Lat university's fan page. After filtering and selecting, we had around 260 admission-related pairs of questions and answers from this source. We have also applied some steps to add Vietnamese accents and find and replace all the abbreviations with normal form words.

In addition, we have collected admission information from our university's brochure. Thereafter, we preprocessed the crawled data, converted it to structured data, and then create about 850 pairs of questions and answers. In addition, questions are classified into 27 intents. Our experimental dataset contains around 800 pairs of unique questions and answers.

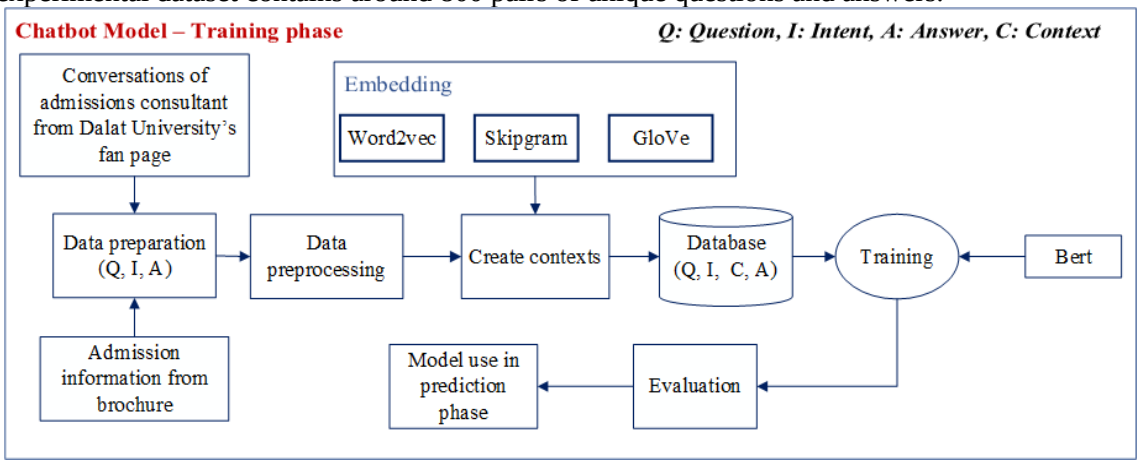

Figure 3. *The model in the training phase*

Figure 4 illustrates an example for our corpus, including four main components: question, intent, context, and answer. The intent is labeled by an integer number from 0 to 26, so we do have 27 intents at this moment. In the future, we will have more intents to cover more plentiful data of all questions and answers.


```
{
"id":0210,
"question": "Cho em hỏi thêm, tổ hợp xét tuyển của ngành CNTT là gì?",
"intent": 18,
"context": "Ngành đào tạo kỹ sư Công nghệ Thông tin có 3 chuyên ngành:
Mạng máy tính, Kỹ thuật phần mềm và Khoa học dữ liệu. Tổ hợp xét tuyển
của ngành Công nghệ Thông tin là A00, A01, A12, D90.",
"answers": "A00, A01, A12, D90"
}
```


Figure 4. *An example data*

From a given question as in the example above, we may have more than one answer. This depends on how we build the corpus and the order of the answers.

2.1.2. Data preprocessing

We applied several steps of preprocessing to the dataset to improve the quality of the data before training the models. We added the context for all the questions and answers of the dataset used for all BERT-based training models. Moreover, we used a stop-words list from [14], ViTokenizer, and ViPosTagger [15] for word separation and POS tagging.

2.1.3. Creating contexts

BERT is a language representation model which has attracted lots of attention due to its great performance in a wide range of NLP tasks, especially in machine reading comprehension and question-answering tasks [3], [16]. The input token to BERT includes questions and paragraphs or the context which contain the answer. These contexts can play an important role in increasing the EM and F1 rates because the context contains lots of useful information needed for answering all user's questions that the system is not trained. Two methods are applied for creating these contexts. First, we use the information supported in the university's brochure for all related questions. Second, for questions asked for numeric data, we can join several answers which have similar information by employing some embedding algorithms such as Skip, Word2vec, or GloVe. To create word representations using these algorithms, we utilized a dataset consisting of 6.1 GB of text from 1.8 million articles collected through the Vietnamese news portal at http://www.baomoi.com. The text is first normalized to lower case and all special characters are removed except these common symbols: commas, semicolons, colons, full stops, and percentage signs.

2.1.4. Training model

In this section, we choose some pre-trained BERT models from huggingface.co, which can support multilingual, including English and Vietnamese. Unfortunately, all models designed only for the Vietnamese language such as PhoBERT [17] do not produce the results we expect. Therefore, in this section we keep only models which have the highest scores of F1 and EM in the experiment as follows:

- *Model1: salti/bert-base-multilingual-cased-finetuned-squad;*
- *Model2: mrm8488/bert-tiny-5-finetuned-squadv2;*
- *Model3: deepset/xlm-roberta-large-squad2;*
- *Model4: bert-large-uncased-whole-word-masking-finetuned-squad.*

Except Model 3 with 256 tokens, other models support the maximum question length up to 512 tokens. Meanwhile, the maximum answer length of all models is 64 tokens. These lengths are good enough to capture short pairs of question-answer in our dataset, without doing content truncation. All models are based on BERT architecture, which originally trained on 2 objects: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). We apply transfer learning from these pre-trained models, extracting embeddings to represent our inputs for the training.

2.2. Prediction phase

Due to the escalation of the corpus, more questions may be created in the future; we outline the prediction phase in Figure 5, also be an optional plan. We will consider using this phase whether depends on the scale of our corpus. In this phase, from a given question, at first, we predict its intents or categories. Then, the question will be matched with other questions in the corpus which have the same intents before choosing the most similar question (or candidate question) or proper context to get the correct answer. Also, in this way, we downsize the time of searching the similar questions and help to improve the performance in general. In scientific eyes, this can be viewed as an attention mechanism where we only focus on the problem that we believe to have the highest probability to get answers.

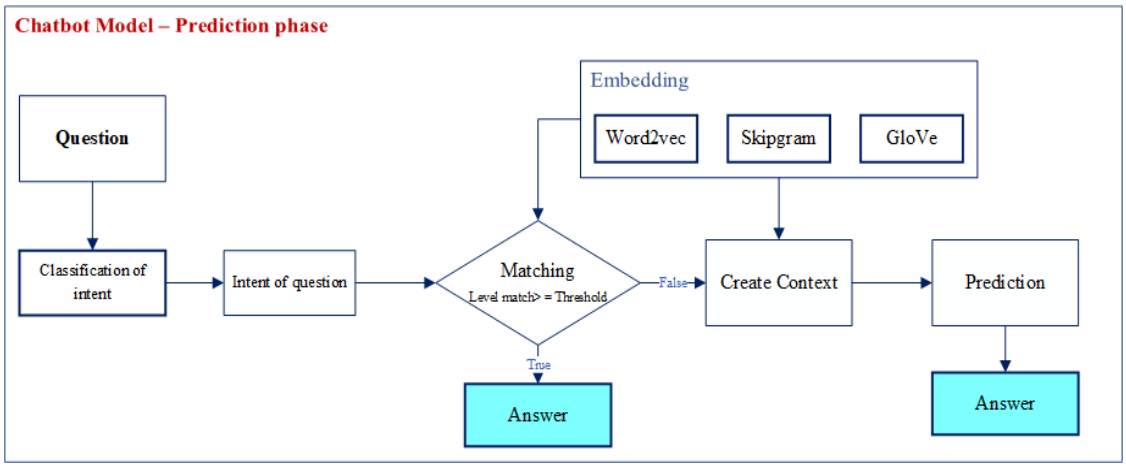

Figure 5. *The model in the prediction phase*

We use some word embeddings (Word2vec, Skipgram, GloVe) to calculate the similarity between that given question and those in the same intents. We set a threshold (i.e. 0.9) to allow the system to know whether to choose a direct answer from a similar question. If this fails, the model will switch to find the proper context in the corpus-based on several information retrieval techniques and reply to the BERT models to produce the answer.

3. Experiments

Table 1. *The scores of F1 and EM over 4 models and 3 methods of catching semantic similarity*

Models	None		Skip-gram		GloVe		Word2Vec	
	F1 (%)	EM (%)	F1 (%)	EM (%)	F1 (%)	EM (%)	F1 (%)	EM (%)
Model 1	53.48	45.43	85.20	79.60	85.90	79.60	88.60	79.60
Model 2	53.19	21.44	82.30	78.00	82.80	78.00	85.50	78.00
Model 3	34.63	28.60	79.10	40.80	79.10	40.80	79.10	44.20
Model 4	20.80	17.97	53.30	50.30	52.80	50.30	54.30	50.30

We randomly split our dataset of 800 questions into 2 sets (*dev_set* and *train_set*) with a proportion of 2:8. Then, we use *dev_set* for the testing process. To improve the performance, for

each question, we use some techniques such as Skip-gram, GloVe, and Word2vec to capture in our dataset all contexts which have the highest semantic similarity to that question (Table 1). In this way, we widen the chance to retrieve the answers to an input question.

Table 1 presents F1 and EM scores over four models combining four methods, which are three different word embedding methods (Skip-gram, GloVe, and Word2vec) to capture semantic similarity in the sentence-level and one method without using any word embeddings called None. Obviously, the performance is low when we do not apply any word embedding methods over four models, in result as the highest F1 and EM scores of None are only 53.48% and 45.43%. In other words, the word embedding methods outperform method None where the highest delta differences of F1 and EM between these methods are 44.47% and 56.56%.

Word2vec shows a slightly better performance compared to Skip-gram and GloVe while Skip-gram and GloVe do not indicate so much difference. Model1 + Word2vec obtains the highest scores for F1 and EM, 88.6% and 79.6% while the lowest F1 score is 20.80% and the lowest EM score is 17.97%. All of the lowest scores belong to method None. In some combinations (methods + models), the big gaps between F1 and EM scores may infer that there is a problem from the model or from the way that we build the dataset.

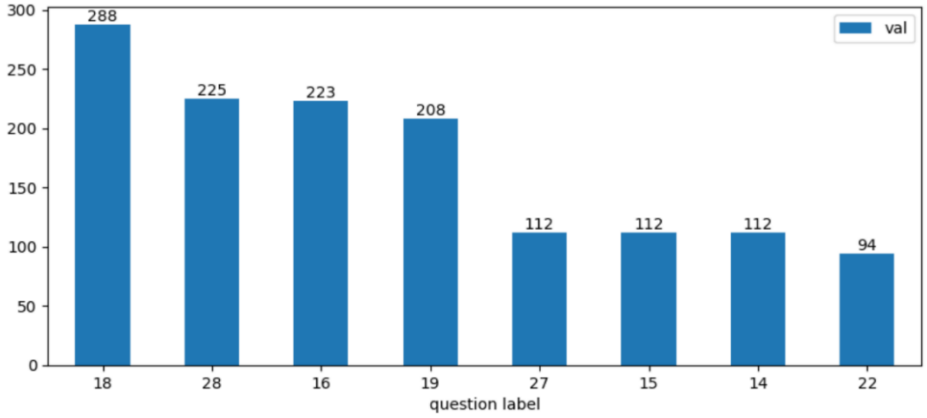

Figure 6. *The number of questions by intents (labels)*

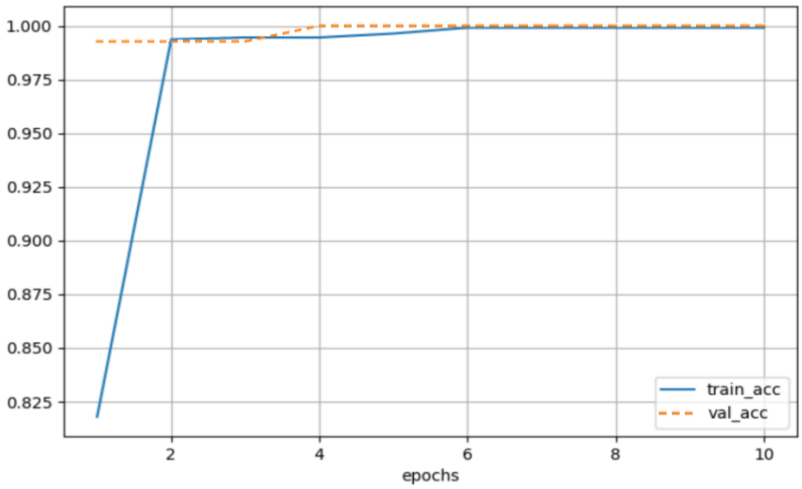

Figure 7. *The train accuracy and validation accuracy in 10 epochs*

We also extend our work on the intent classification, which has various methods. However, since the BERT models (bert-base-cased) outperform other methods in the experiment, we only apply this method to our intent classification. After using some techniques to map terms in

questions to Wikidata and use their aliases to form new sentences, we increase our corpus from 800 to 1374 questions, then use them to train over 8 intents showed in Figure 6. Also, the figure points out the imbalanced problem happening in our corpus. Fortunately, BERT models can deal well with this problem.

In the training process, we divide the corpus into 3 parts: training, testing and validating sets with the ratio of 8:1:1. After 10 epochs in Figure 7, we gain the best model with a training accuracy of 100%, validation accuracy of 99.9%. The accuracy values are extremely high because we work with a small corpus and a small number of intents, but this also reflects the prominent performance of BERT models that we found in a similar work by Jin et al. [18]. The testing accuracy is 100% when we choose the best model from epoch 8 which the validation accuracy gains the highest value.

4. Conclusion

In this paper, we have proposed a method for building a dataset in a chatbot application, which has three basic steps, including data collecting, data preprocessing, and context building. In the collecting data step, we have collected around 800 pairs of questions and answer in the university's admission domains from nearly 260 conversations and our university's brochure. In preprocessing data, we have applied several techniques for the Vietnamese dataset such as adding Vietnamese accents, finding and replacing all the abbreviations with normal form words, removing stop words, using ViTokenizer and ViTagger for word separation, and POS tagging. Thereafter, we use embedding techniques such as Skip-gram, Word2vec, or GloVe to search for similar contexts for enlarging the chance to find the appropriate answers in the models. The experimental results show that these embeddings could improve significantly the F1 and EM scores of the BERT-based models. After obtaining the dataset, we applied some available pre-trained multilingual BERT-based models from huggingface.co to our dataset and realize that *salti/bert-base-multilingual-cased-finetuned-squad* model is the most suitable model among four models examined because it achieved the F1 score of 88.6% and EM score of 79.6%. We also do our extra work on intent classification and realize that the BERT models outperform other methods with the validation accuracy and train accuracy are 99.9% and 100%. In future work, we intend to fine-tune these pre-trained BERT representations with additional architectures to apply to specific tasks. Furthermore, we also develop the corpus size by using oversampling and under sampling methods to control the corpus distribution by intents; as well as organizing our corpus under knowledge graphs to easier manage and integrate it to other research.

REFERENCES

[1] S. Hussain, O. A. Sianaki, and N. Ababneh, "A survey on conversational agents/chatbots classification and design techniques," in *Primate life histories, sex roles, and adaptability*, U. Kalbitzer, A. M. Jack, and M. Katharine, Eds. Berlin: Springer, 2019, pp. 946-956.

[2] N. N. Khin and K. M. Soe, "University chatbot using artificial intelligence markup language," in *The IEEE Conference on Computer Applications (ICCA)*, Myanmar, 2020, pp. 103-107.

[3] D. Jacob, W. C. Ming, L. Kenton, and T. Kristina, "BERT: Pre-training of deep bidirectional transformers for language understanding," Cornell University, October 11, 2018. [Online]. Available: https://arxiv.org/abs/1810.04805. [Accessed Apr. 28, 2020].

[4] B. R. Ranoliya, N. Raghuwanshi, and S. Singh, "Chatbot for university related FAQs," in *The 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, India, 2017, pp. 125-131.

[5] G. Hiremath, A. Hajare, P. Bhosale, and R. Nanaware, "Chatbot for education system," *International Journal of Advance Research in Technology*, vol. 4, no. 3, pp. 37-43, 2020.

[6] A. Mondal, M. Dey, D. Das, S. Nagpal, and K. Garda, "Chatbot: An automated conversation system for the educational domain," in *The International Joint Symposium on Artificial Intelligence and Natural Language Processing*, Thailand, 2018, pp. 103-109.

[7] T. N. Nguyen, and Q. D. Truong, "Support system for college admissions counseling," (in Vietnamese), *Can Tho University Journal of Science*, no. 15, pp. 152-159, 2015.

[8] V. B. Nguyen, "Building a dialogue model for Vietnamese in the open domain based on the sequential learning method," (in Vietnamese), MSC. Thesis, VNU Hanoi-University of Engineering and Technology, Hanoi, 2016.

[9] T. T. Nguyen, "Application of multi-class svm supervised learning algorithm in building a Vietnamese Q&A chatbot system," (in Vietnamese), in *The National scientific conference on IT and applications in various fields*, Vietnam, 2018, pp. 98-105.

[10] M. T. Vi, V. M. Do, D. N. Tran, and T. A. Nguyen, "Building a Chatbot solution to support healthcare on Vietnamese domain," (in Vietnamese), in *The National Workshop on Application of High Technology in Practice*, Vietnam, 2021, pp. 87-95.

[11] T. Mikolov, G. Corrado, K. Chen, and J. Dean, "Efficient estimation of word representations in vector space," in *The International Conference on Learning Representations*, USA, 2013, pp. 202-210.

[12] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," Cornell University, October 16, 2013. [Online]. Available: https://arxiv.org/abs/1310.4546. [Accessed Oct. 15, 2020].

[13] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *The Conference on empirical methods in natural language processing*, Quatar, 2014, pp. 80-89.

[14] V. D. Le, "Vietnamese-stopwords," March 15, 2019. [Online]. Available: https://github.com/stopwords/vietnamese-stopwords. [Accessed April 25, 2020].

[15] V. T. Tran, "Python Vietnamese toolkit," Pyvi 0.1.1, Jun 30, 2021. [Online]. Available: https://pypi.org/project/pyvi. [Accessed Sept. 20, 2021].

[16] H. Zhiheng, X. Peng, L, Davis, M. Ajay, and X. Bing, "TRANS-BLSTM: Transformer with Bidirectional LSTM for Language Understanding," Cornell University, 2003. [Online]. Available: https://arxiv.org/abs/2003.07000. [Accessed Sept. 15, 2020].

[17] Q. D. Nguyen, and T. A. Nguyen, "PhoBERT: Pre-trained language models for Vietnamese," Cornell University, 2020. [Online]. Available: https://arxiv.org/abs/1310.4546. [Accessed Dec. 19, 2020].

[18] D. Jin, Z. Jin, J. T. Zhou, and P. Szolovits, "Is bert really robust? a strong baseline for natural language attack on text classification and entailment," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 5, pp. 8018-8025, 2020.

[19] H. Zhiheng, X. Wei, and Y. Kai, "Bidirectional lstm-crf models for sequence tagging," Cornell University, 2020. [Online]. Available: https://arxiv.org/abs/1508.019912015. [Accessed Dec. 19, 2020].