Deep Learning-based Multi-Chatbot Broker for Q&A Improvement of Video Tutoring Assistant

Ozoda Makhkamova
Department of Software
Konkuk University
Seoul, South Korea
crchoc@gmail.com

Kang-Hee Lee
Department of Software
Konkuk University
Seoul, South Korea
khlee91@konkuk.ac.kr

KyoungHwa Do
Department of Software
Konkuk University
Seoul, South Korea
doda0905@gmail.com

Doohyun Kim
Department of Software
Konkuk University
Seoul, South Korea
doohyun@konkuk.ac.kr

***Abstract*—Chatbot is software for conversations where the opponent is a program instead of a human. Marketing, business, education, healthcare, and other fields are using chatbots for the convenience of users. The functionality of the chatbot is like a virtual assistant for users to help in purchasing and idea generations, resulting in better services and faster response. This paper presents the fundamental components and working principle of a multi-chatbot broker for video tutoring assistant for e-learning as an online video lecture assistant. The proposed broker implements a deep learning approach for a single video having multiple chatbots to select relative answers for a different perspective of each user, thereby improving the responsiveness of feedback and its quality. This means reducing the responding time of video tutoring assistant and increasing the quality of Q&A data by adding multiple chatbots from different viewpoints for a single video.**

Keywords—chatbot, deep learning, multi-chatbot, video tutoring, broker

I. Introduction

Due to the flourishment and development in Internet technologies, developers try to make services better and faster, such as the Mobile Internet, online banking, electronic documentation, etc. Moreover, tutoring is one of the important services that need to be improved. Learning approaches can be classified into offline and online learning. Offline learning is an approach where a mentor teaches a group of people in an educational organization such as a school or university. The advantage of this approach is the use of visual aids, a discussion of the material covered with the students. Contrary to that, online learning is an approach where a student searches for knowledge via the Internet, where there is a huge number of articles, blogs, video tutorials. The features of online learning include saving money and time [1], choosing an area of interest and a course of study individually. The disadvantage of this approach is using a long time to get feedback [2] on questions due to the nature of online learning, leading to cause a low percentage of students who watch the lecture to the end [3]. For this reason, it is practical to use a chatbot to get answers to the questions quicker.

As commonly known, a chatbot is a program that functions as an assistant to improve a service for users of the software supported by artificial intelligence [4]. Chatbots are used in many services as ordering food, supplying information, or just for communication. An intelligent system in the travel domain on the Echo platform that gathers user preferences and based on this data constructs a recommendation system for users is presented in [5]. A chatbot for university-related FAQs, which gives an accurate and efficient answer for any query based on the dataset of FAQs using Artificial Intelligence Markup Language (AIML) and Latent Semantic Analysis (LSA) is introduced in [6]. Intelligent conversational Bot for Massive Online Open Courses, MOOC-bot, is developed and integrated into MOOCs website to respond to the learner's inquiries using text or speech input [7].

All these examples have a single chatbot that causes low responsiveness of the systems. This article presents a multi-chatbot broker for video tutoring assistant to improve the responsiveness of feedback, which is a relative answer for a different respective of each user in video tutoring, thereby increasing the rate of learners who successfully finished the course. This broker function as an assistant that responds to questions of the listener in video tutoring. A feature of this multi-chatbot broker is that anyone can attach their chatbot for a specific video that leads to having more data to train broker and improve quality of Q&A sets. When the listener sends a query, the broker identifies which of the chatbots has a more suitable response and higher priority and sends it to the user. Throughout the experiments, the accuracy of a proper response to the question by defining the desired chatbot was revealed.

This rest of the paper is organized as follows: Section II introduces issues and approach of using the multi-chatbot broker for Q&A quality improving, Section III detailed the proposed system along with the architecture. Experimental results of the implementation are covered in section IV. The paper is concluded in Section V with future research plans.

II. Multi-Chatbot

According to [8], online education has grown from Internet propelled distance education to online higher education enrollments, and [9] shows predictions on chatbot usage from 2014-2025. These bring to the idea of uniting online video tutoring and chatbot technologies to make online education more efficient by immediate reply to all possible questions

2375-9356/20/$31.00 ©2020 IEEE
DOI 10.1109/BigComp48618.2020.00-71

related to the video. However, a pair of single video and single chatbot can have few disadvantages. First, one chatbot takes a long time to respond to a user's query and has difficulty in finding an answer in various information. The second disadvantage is that a lecturer and a listener can have a difference in the level of knowledge that leads to the fact that the listener may have questions about which the lecturer did not cover. The other one, the amount of question and answer sets (Q&A) that one chatbot can learn is less. For a chatbot to learn more Q&A sets from different viewpoints and shorten the time of responding, it is necessary to build a system with multiple chatbots. While a pair of single video and single chatbot has tight coverage of Q&A collections, the set of one video with multiple chatbots can have extensive coverage of Q&A with various information from different views of understanding. For this, a broker, which plays a leading role in connecting video and multiple chatbots, can be used to improve the quality and coverage of Q&A sets (Figure 1).

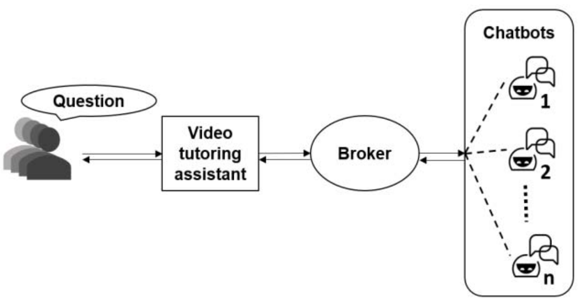

Figure 1. Multi-chatbot

The broker is a program that makes a connection between sender and receiver and works as a middleware software. The primary purpose of a broker is to take incoming messages from the client and perform some actions on them. In the case of multi-chatbot, an intermediary server, which is the broker, will be required even for the ability to send questions back to the chatbots. The proposed broker is a middleware tool to connect the video with different chatbots and find a chatbot, which potentially has a more relevant answer to the question of the user. The broker learns question sets of every chatbot in advance, using deep learning algorithms, to choose a proper chatbot as accurately and correctly as possible. To do this role, the user question is compared with learned question sets of those chatbots to find a more relevant chatbot and to give a response. As a result, the multi-chatbot broker expands the coverage of Q&A in video tutoring to make online learning more practical and usable.

III. PROPOSED SYSTEM

The main goal of the proposed system is to increase the quality of response of chatbots with more than one chatbot for one video by the multi-chatbot broker. The usage of various chatbots causes possibility to have extensive Q&A sets in a different field and from different viewpoints. This system aims to shorten the time of getting feedback, achieved by using multiple chatbots. Chatbot developers develop each chatbot from their own viewpoints with the different level of knowledge and understanding and extend Q&A coverage. The main components of the proposed system are: 1) a user, who watches a video for education; 2) video tutoring assistant, which helps in classifying user's requests; 3) a broker, which aims to choose proper chatbot corresponding to user's query; and 4) chatbots, which can be made by different developers with Q&A sets for video. The target of this system is the people, who use this e-learning to learn, especially with video tutoring, and want to get answers from the tutor as faster as possible.

A. Modules and interaction

As mentioned above, this paper proposes the system with the multi-chatbot broker for users, who use online video tutoring as a way of getting an education, to help video tutors to provide faster feedback to the user and increase the number of students who use online video tutoring. User can use the multi-chatbot broker for video tutoring assistant to find answers to questions which can appear while watching the video.

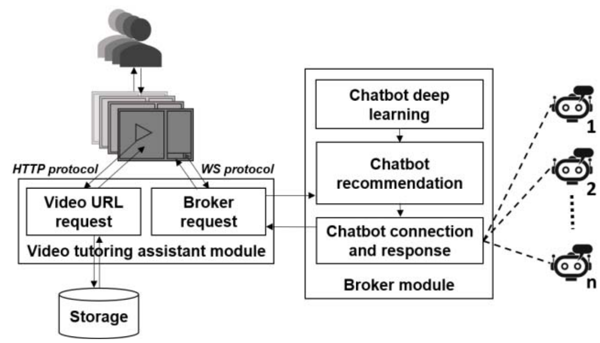

Figure 2. System architecture

Figure 2 shows the system architecture of the proposed system and sets aside main components as video tutoring assistant module, broker module, and chatbots, which are connected using socket technology to exchange messages between each module.

Video tutoring assistant module is part of the system which handles presenting the interface of the system and classifying the user's request. The interface of this system is a webpage, where user can see video and chatbot window, and inquire chatbot when a question arises while watching the video. Request classification is an action which dependents on the type of protocol divides requests into video URL request and broker request. If the user clicks on the video, HTTP protocol requests URL of the video from storage and displays the video to the user. If the user requires chatbot, WS (WebSocket) protocol calls a broker module for query classification and connection to the chatbot with the most relevant question and answer set.

As it was explained earlier, the broker is a middleware tool for connecting sender and receiver and managing transactions of entities between two parts. In this system, the broker is an instrument for maintaining the connection of video and its multiple chatbots. The main aim of the broker is to evaluate chatbot priority for a specific video and to choose relevant chatbot with needed Q&A set. Broker can have more than one chatbot, which makes coverage of questions more extensive and increases responsiveness for questions.

The connection between modules is made with the socket to transfer such data as queries and responds, video name and chatbots IP. Video name and chatbots IP are used to help broker for connecting with chatbots of the specific video. As one video can own multiple chatbots, chatbot information is necessary for managing such a connection. Question data of all chatbots are used for training broker to predict chatbot, which has the relevant question to the user's query and send the answer to the user. More explanation on the broker module can be found in the next part.

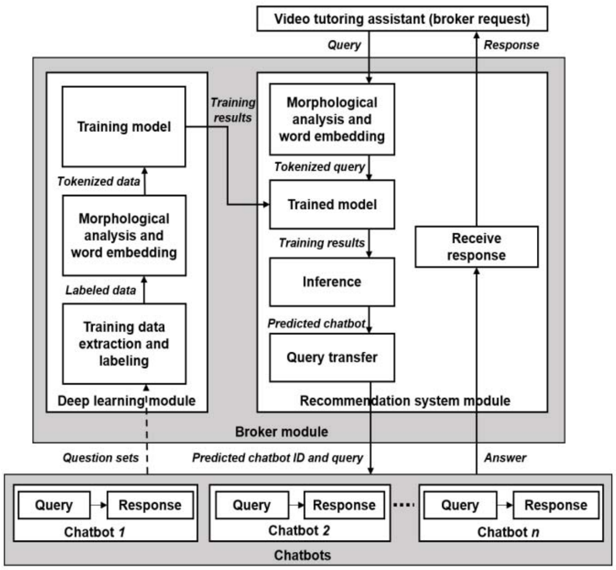

Figure 3. Broker module architecture

B. Broker and deep learning

The main aim of the broker is achieved by using deep learning algorithms for classifying user's query using trained model and evaluation of priority of chatbots depending on the question data of all chatbots data. It makes the broker manage chatbots depending on priority and to predict chatbot, which has a related question to the user's query.

Broker module architecture (Figure 3) has two main parts: the deep learning module and chatbot recommendation system module. The deep learning module is responsible for training question data sets from each chatbot to increase the quality of responsiveness of video tutoring assistant. At the deep learning module, training data passes three actions: extracting and labeling training data of each chatbot, morphological analysis and word embedding, and training model with tokenized question data from all chatbots. There are few techniques for word embedding, such as Word2Vec, FastText, Glova, etc. This system uses Word2Vec[9] technique for word embedding, which infers that the semantic relation intimacy through distance similarity between words is used. Word2Vec has two methods for training word embedding: CBOW and skip-gram. CBOW model indicates the middle word through words preceding and following. On the other hand, the skip-gram model uses one word to infer the appearance of words around. For word embedding in this system, skip-gram model of the Word2Vec technique is used. The trained model is used for predicting chatbot, which will produce an answer. This system uses four classifiers to conduct experiment, such as Naive Bayes Classifier (NBC), Stochastic Gradient Descent (SGD), Bidirectional Long Short-Term Memory (Bi-LSTM), and Convolutional Neural Network (CNN). These classifiers are used to train question data from all chatbots.

The chatbot recommendation system module is responsible for recommending the chatbot with the answer, which the user is searching for. For achieving its goal, the recommendation system module uses a trained model from deep learning module to predict a proper chatbot for current user's query. User's query goes through morphological analysis and word embedding to make prediction faster and convenient. As the chatbot recommendation system module predicts chatbot, the query transfer module connects with the server of the predicted chatbot and sends the query to the chatbot, and the recommendation system receives the answer from chatbot to show it to the user.

Algorithm 1. Deep learning on question data from all chatbots and predicting relevant chatbot

Input: Question
1: for training { Get-chatbot-data(chatbots' question sets) → *Get question sets from all chatbots for training*
2: Data-labeling(chatbots' question sets) → *label all questions*
3: Word2Vector(chatbots' question sets) → *Word embedding*
4: nltk.Text(chatbots' question sets, tokens) → *divide chatbots' question sets into tokens }* end for
5: pipeline(question vector, chatbots' question sets) → *sending input question to chatbots' question sets*
6: model(question vector, chatbots' question sets vector) → *drawing a model of training data*
7: predict(selected chatbot, answer)→ *predict chatbot based on trained model*
10: **Output**: answer for a question

Algorithm 1 is to analyze and train question data of all chatbots and to select more relevant chatbot to give the correct answer for the user's query. First, question data sets from all chatbots are taken for the training phase. As for the training phase vector data of words is needed, question data is labeled and goes through word embedding to convert it to vector. Next step is to divide question data into tokens. The training model of question data is drawn using the input question vector and question sets vector. When a program predicts the chatbot with a more relevant question, it connects with selected chatbot and prints answer to the user. The output of this algorithm is the answer to the user's query.

IV. Experiment and Analysis

The experiment is aimed to evaluate the performance of the proposed system. Since the user's query is not sent directly to a chatbot but passes through the broker, it causes a possibility of having delays. However, it is beneficial in terms of improving accuracy and constructing a system that accommodates various chatbots. Since training of broker is not related to the actual execution time, performance deterioration by the broker in the actual execution will not be significant.

For the evaluation of the performance of the proposed system, four classification algorithms were used. As an experimental data, KorQuAD database with 66181 of Korean Q&A sets (60407 sets – training data, 5774 pairs – test data) is used, but test data is not open source. Data were divided into 57386 pairs for training data and 3021 pairs for test data followed by two test cases for experimental analysis. The first case (T1) is using simple questions with only nouns like "today weather", while the second case (T2) involved questions, which are not significantly written in order like "weather tomorrow is how".

TABLE I. PERFORMANCE COMARISON FOR EACH CLASSIFIER ACCURACY

	Naïve Bayes classifier	**SGD classifier**	**Bi-LSTM classifier**	**CNN classifier**
Train	0.892	0.849	0.997	0.996
Test	0.801	0.779	0.631	0.725
T1	0.862	0.846	0.629	0.561
T2	0.89	0.823	0.449	0.567
Average	0.861	0.824	0.677	0.712

Naive Bayes Classifier showed more than 80% of accuracy in training. As NBC considers only the frequency of occurrence of specific words per class, not the order of words, it shows high accuracy in T2. In the case of T1, as only nouns were extracted from the sentence and the frequency of appearance of other morphemes could not be considered, it shows low accuracy. For the SGD classifier, the generated accuracy is close to 80% where training data, T1, and T2 have the accuracy of more than 80%, test data - less than 80%, showing lower accuracy in comparison to than NBC. In the case of Bi-LSTM classifier, training data accuracy is close to 100%, but test data and T1 accuracy are just higher than 60%. Because the Bi-LSTM classifier uses RNN structure, it considers the order of the sentence bringing low accuracy in T2 - less than 50%. For the CNN classifier, the accuracy is high for training data, but for test data lower than 80% and 60% accuracy is generated for T1 and T2, respectively.

The accuracy of all classifiers is specified in table 1 showing average best performance for naïve Bayes classifier with an accuracy of 86%. Whereas, the training data does not show higher accuracy than the CNN and Bi-LSTM models showing a tradeoff. On the other hand, CNN and Bi-LSTM classifiers show the high performance of nearly 100% accuracy of the training data. However, it shows low accuracy on test data and T1 case with only nouns and T2 case mixed with all morphemes. Overall, the SGD classifier is less accurate than the naïve Bayes classifier. This experiment shows that naïve Bayer classifier is outstanding and has higher accuracy compared other classifiers.

V. CONCLUSION AND FUTURE WORK

This paper presents deep learning-based multi-chatbot broker for improving quality of Q&A in video tutoring assistant. The goal of the proposed system is to increase the number of Q&A sets from different chatbots, shorten the time of giving feedback to the user and improve the quality of Q&A by uniting different chatbots in one broker. In conclusion, the proposed system is designed to help developers to increase the responsiveness of chatbot by adding several chatbots for a single video, and users to have better and faster feedback on their question in online video tutoring. The experiments show the difference in performance accuracy of the different classifier, such as Naïve Bayes Classifier, SGD, Bi-LSTM, and CNN. As a result, Naïve Bayes classifier has high average accuracy compare to others.

For the future work, there is still much work, one can focus on adding personalized recommendation system of related video courses for the current video based on a user's query to extend the functionality of video tutoring assistant. It can help the broker to know that the user's often asked questions and recommend other courses which are useful and necessary to him that will lead to increasing the usage of e-learning more effectively.

ACKNOWLEDGMENT

This research was supported by the MSIT(Ministry of Science, ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2019-2016-0-00465) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation).

REFERENCES

[1] Trespalacios, J & Rand, J., "Using asynchronous activities to promote sense of community and learning in an online course," International Journal of Online Pedagogy and Course Design, 5(4), pp. 67-81, 2015.

[2] K.M. Alraimi, H. Zo, A.P. Ciganek, "Understanding the MOOCs continuance: the role of openness and reputation," Journal of Computer & Education, 80, pp. 28-38, 2015.

[3] Anna Espasa, Julio Meneses, "Analysing feedback processes in an online teaching and learning environment: an exploratory study," Journal of Higher Education, pp 277–292, 2010.

[4] Shawar, Bayan Abu and Eric Atwell, "Using dialogue corpora to train a chatbot," Proceedings of the Corpus Linguistics 2003 conference Lancaster University, p. 681–690, March 2003.

[5] A. Argal, S. Gupta, A. Modi, P. Pandey, S. Shim and C. Choo, "Intelligent travel chatbot for predictive recommendation in echo broker," 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, pp. 176-183.R, 2018.

[6] B. R. Ranoliya, N. Raghuwanshi and S. Singh, "Chatbot for university related FAQs," 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, 2017, pp. 1525-1530.

[7] S.L. Lim, O.S. Goh, "Intelligent conversational bot for massive online open courses (MOOCs)," Universiti Teknikal Malaysia Melaka, Malaysia.

[8] S.Palvia, P.Aeron, P.Gupta, D.Mahapatra, R.Parida, R.Rosner and S.Sindhi, "Online Education: Worldwide Status, Challenges, Trends, and Implications" Journal of Global Information Technology Management, 2018, VOL. 21, pp. 233-241.

[9] Chatbot Market Size And Share Analysis, Industry Report, 2014 - 2025. https://www.grandviewresearch.com/ industry-analysis/chatbot-market. Accessed: 2018-05- 07.