
SMARTBOT FOR LEARNING LANGUAGES

Animesh Kumar Jha*¹, Shivam Pawar*², Abhinav Singh*³,

Aman Panwar*⁴, Vishal Patil*⁵

*^{1,2,3,4}Student, Department Of MIT School Of Engineering, MIT ADT University,
Pune, Maharashtra, India.

*⁵Assistant Professor, Department Of MIT School Of Engineering, MIT ADT University,
Pune, Maharashtra, India.

ABSTRACT

Chatbots (conversational interfaces) showcases modern methods for human beings to communicate with Artificial Intelligence bots. Conventionally, to find results to our queries we utilize software programs involving a search engine or call customer care. Chatbot enables a user to speak with a bot in the same traditional manner of speaking to another human being. Many online portals, home, and handheld devices have started incorporating chatbots. Artificial Intelligence is the root of the innovative application of chatbots. Recent advances in machine learning have greatly improved the accuracy and effectiveness of making chatbots a viable option for many organizations. This improvement in machine learning has created a spark for further research which would undoubtedly usher in continued enhancements in the accuracy and efficiency of future chatbots.

Keywords: Machine Learning, Artificial Intelligence (AI), Chatbot, Web Application.

I. INTRODUCTION

A chatbot is an application that speaks or talks with its user on topics in a wide range of domains being business, leisure, or education, communication takes place in a conventional manner or text or speech. Chatbots are utilized across almost every domain, namely application control, technical support, customer care, website navigation, marketing, education, and training. As users adapt to interacting with digital entities the development of this domain of applications will create many more features and eliminate technical barriers to incorporating new users. The creation of a digital assistant that can manage our tasks, answer all our queries and effectively solve our problems is the goal of this domain of application. These digital assistants make use of machine learning applications and can manage a few everyday tasks just like a traditional human assistant (such as spam detection, flagging important content, and conversation) to ease the user's workload and increase efficiency. Text-based chatbots generally keep to a set of custom guidelines or flow for answering the queries of the end-user. These guidelines help them to converse or manage tasks effectively requested in its specific domain, though if the chatbot is not trained to answer a question the bot is not effective enough to make responses, providing only a generic default message. Since the 1970s, Intelligent Tutoring systems have been developed in the educational domain, and hence have a history of being used for teaching in educational centers. Artificial intelligence is used for applications in the education field to increase and individualize automation in tutoring. Knowledge of the research and graphical interface is used to create an eye-catching, accurate, and resourceful teaching bot that along with making the most of the technical advancements in the artificial intelligence field, also understands emotional, cognitive, and social educational concerns. The incorporation of chatbots into the educational area over the last decade implies an increase in interest in how chatbots might be implemented for teaching and learning. Useful chatbot systems can provide benefits of instant availability and the ability to respond naturally through a conversational interface with the same advantages as an interview. Additionally, chatbots demonstrate the ability to create easy-going interactions with users so that they can be leveraged to support engagement, as well as set out goals, strategies, and outcomes of learning and training. The purpose of this study was to explore the use of a chatbot as a support system for the learning of a new language of choice using an instant personalized messaging application.

II. METHODOLOGY

This chatbot project will help the teacher to make them more comfortable with the languages. Usually, when a teacher teaches in the classroom after the end of the lecture most of the students don't remember what has been taught on the lecture after day. This chatbot will help a teacher to upload content in the chatbot or to give

it to the person who will upload it in the chatbot. In this way, after the updating of the chatbot, one can ask or talk to a chatbot to have a quick recap of what learned during classes and make learning more fun. In the future development of this project would bring easier to learn approaches and if any university wants, they can build in such a way that it can test students learning and would help in further growth. We have used Web Development Languages for designing the chatbot for interactive GUI and easy accessibility of the application for the students, as they can access the Web Application from any platform or device of their choice Chatbot responses to user messages should be large enough to continue the conversation. Sometimes it is not necessary for the chatbot to understand and remember all the details of a sentence; depending on the requirements we need to modify different parts / items and use one, but the flow of basic communication remains unchanged. However, choosing the right format depends on the type of chatbot domain. Sometimes in fun conversations, the composition varies, where you can use it to measure user input pattern or save chat history to gain an understanding and determine the type of response. The purpose of entertainment bots is to increase the average time spent chatting with a user through a bot. If time is limited, conversation will likely be brief; which is why chatbot is boring. Chatbot for business is usually very popular and works with job applications, which means they are designed for a specific purpose or to accomplish a specific purpose. It should be very important for the chatbot developer so that the purpose is prepared and defined in advance. By that definition it means one has to be prepared for a lot of needs before growing a real chatbot. For example, a travel chatbot provides information about flights, hotels, and trips and helps you find the best package according to user terms and can make the reservation accordingly. Also, Google Assistant easily provides the information requested by the user and Uber bot takes the request for a ride. Chat business conversations are short, less than 15 minutes. Each conversation has a specific purpose and the quality of the conversation can be determined by how many users reach the goal as follows: Did the user get the information they were looking for? Has the user successfully booked a flight with a hotel? Has the user purchased products that help solve a quick problem? Maybe some chatbots don't fit in this category, but it should be good enough to work on most bots currently developed. So in order to better understand the models I will explain the simplest first and then increase the complexity. Remember, there is no suitable way to build a chatbot, you can use any method that suits your operating case. Also, it is not wise to use a complex method with a simple use case. Since you can't use a pestle to kill ants, it will only make your life more difficult.

Rule-Base chatbot

With the rule-based chat approach, we can create both entertainment and business chatbots. In this type, we need to specify the possible input question that the user will ask and its associated response. Refer to the example below: What is your NAME When a rule-based bot receives a query, it checks all the user-defined patterns up till it finds the pattern which matches the question. If a match is found, the chatbot uses the correct response template to generate the response. In other words, the developer is defining a set of rules or patterns as conditions to give a response to the user. Heuristics for selecting the right response can be engineered in many different ways, sometimes a general if-else condition is implied whereas sometimes a machine learning algorithm is trained on these input patterns. This approach is very popular for entertainment bots. AIML is a widely used language for writing patterns and responses.

Intent Classification Architecture

The challenge with the pattern-based or rule based approach is that, the patterns should be coded manually, and it is not an easy task. Imagine, if we try to increase the capability of the chatbot, then we need to hardcode every condition the chatbot can answer. This is extremely difficult to maintain and can cause a lot of overlapping confusion between the patterns. This can possibly reduce the accuracy of the chatbot. Also as mentioned earlier single question can be asked in multiple ways. Therefore, it is not easy for a human to define and find pattern by natural language understanding, whereas computers can do this easily. In other words, for narrow domains a pattern matching architecture would be the ideal choice. However, for chatbots that deal with multiple domains or multiple services, broader domains, sophisticated, state-of-the-art neural network architectures, such as Long Short-Term Memory (LSTMs) and reinforcement learning agents are the best options. Machine learning can be applied on intent classification algorithm to classify and find patterns in the natural language, thanks to word embedding. You just need to provide training set of a few hundred or

thousands of examples, and it will pick up patterns in data and classify the intent accurately and in fairly less amount of time. Such machine learning algorithm can be built using any popular machine learning library like Sci-kit learn, Tensorflow or PyTorch. Another option is to use one of cloud API: wit.ai, api.ai, Microsoft LUIS.

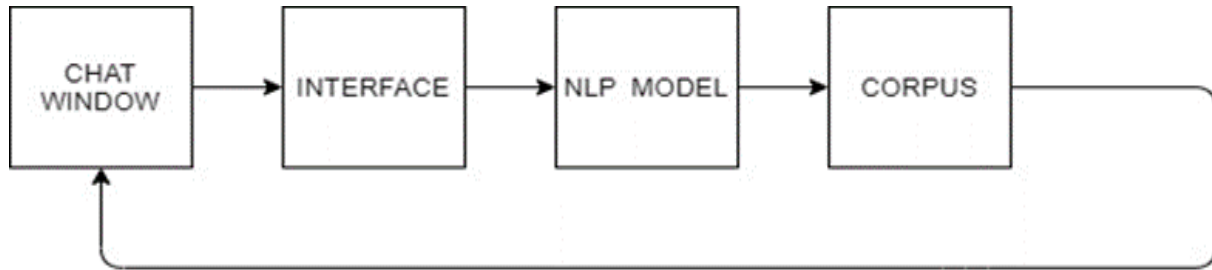


Figure 1: System Architecture Diagram.

Response Generation

Pattern matching, intent classification and context extraction helps to understand what user message means. Whenever the chatbot gets the intent and the context of message, it shall generate a response. But now the question arises how? You can approach it differently based on the type of chatbot you are building. Simplest way is to response a static message by placing some of the values variable, based on the entity and processing that you undergo. This processing can be a Query to a database or calling some 3rd party API. ChatScript provide a very handy solution to this kind of problems. For Example: Q: What is the age of Obama? A: Age of \$('Entity') is \$('Query result') Apart from this, different kinds of chatbots offer different processing and response mechanism. For example, a medical chatbot will have a predefined set of Symptoms Model for different diseases and for generating the response it will apply statistical modelling and probabilistic approach to generate similarity with symptoms model to predict the disease based on the user input to chatbot questions and gives response accordingly. A question answering chatbot will dig into the knowledge graph or a database to query the request and generate the best answer score to give the correct response. On the other hand, a weather based chatbot will call a 3rd party API's to get the right data and place it into fixed messages to give the response. This concept of generating the responses based on user message is also called Response modelling. There are two major types of modelling: 1) Generative Modelling 2) Retrieval based Modelling This post mostly covers the Retrieval based models. They are much easier to build and provide more predictable results. The chatbot uses the intent and context of conversation for selecting the best response from a predefined list of bot messages. You probably won't get 100% accuracy of responses, but at least you know all possible responses and can make sure that there are no inappropriate or grammatically incorrect responses. Retrieval-based models are more practical at the moment. Also many algorithms and APIs are readily available for developers. Generative Response Model is the future of chatbots where the output not only depends on the current input, but to a series of input given in the past. I will write a separate post on Generative response model, but for now it is out of the scope of this post. However, 90% of the chatbot in the market today is built using Retrieval based modelling because of its huge capability and ability to solve maximum problem and making life easy for people.

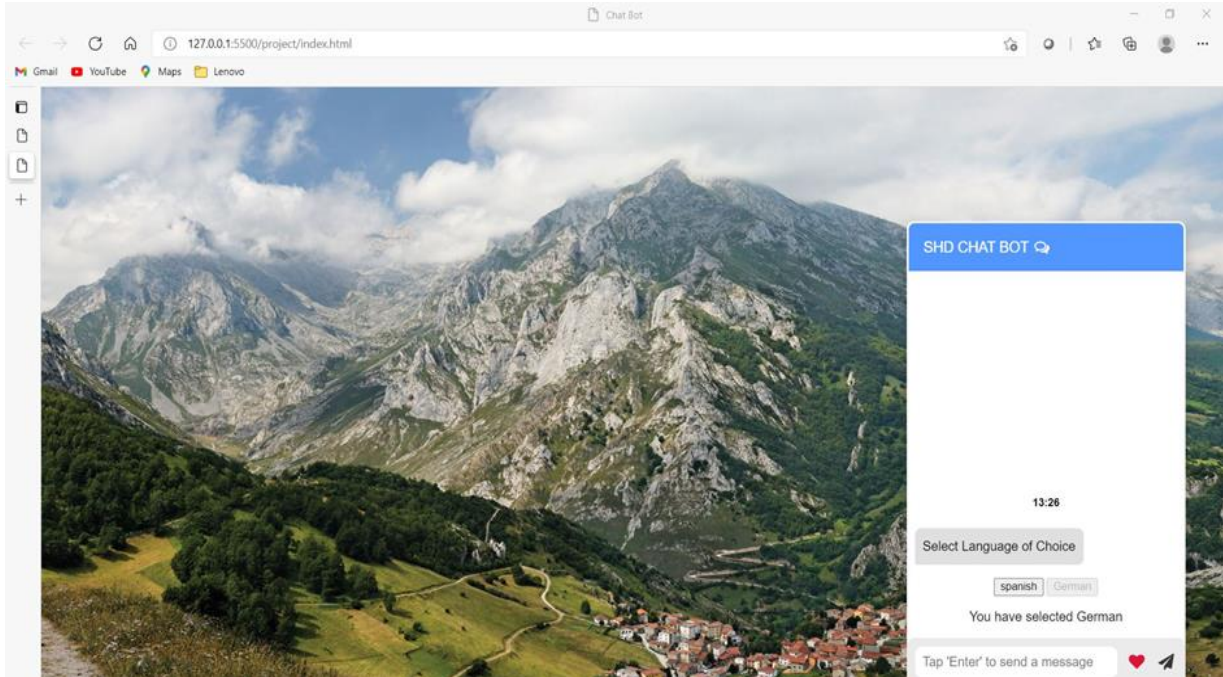


Figure 2: Selection of language to learn.

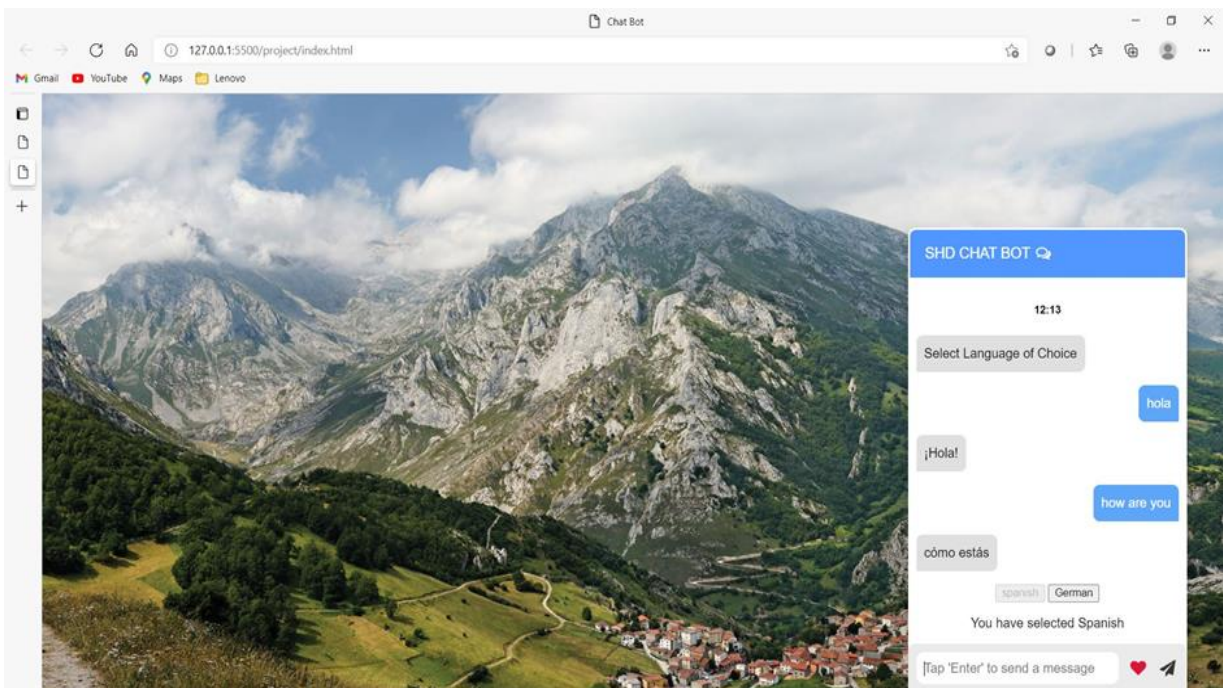


Figure 3: Spanish Language Chatbot.

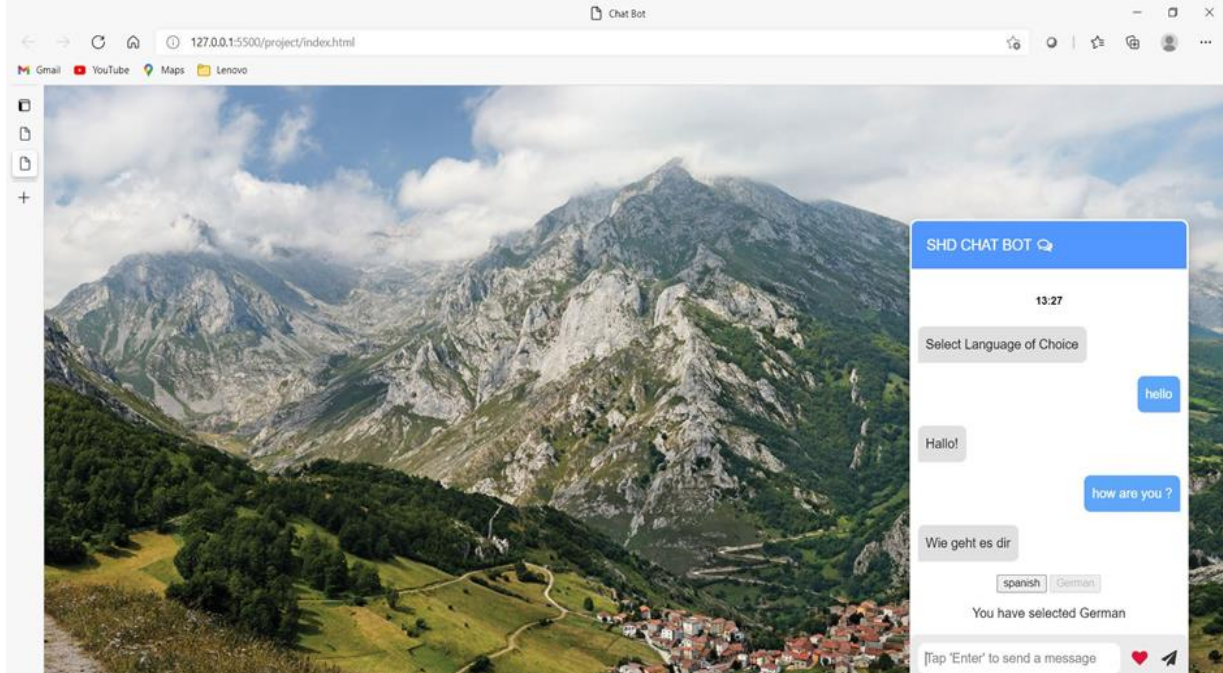


Figure 4: German Language Chatbot.

A chatbot is a man-made intelligence that converses with a user through apps or messaging. Chatbots are the longer term of AI, as they supply the closest to a natural conversation between humans and machines. Chatbots have been there for a long time; the first-ever chatbot was Eliza, but it wasn't as complex as the chatbots we have nowadays, such as Alexa or Siri. Due to the new ML and AI techniques, chatbots are expected to complete 90% of our tasks in the future, but there is still a long way to go. The progress we've made with chatbots so far is, nonetheless, astonishing. A few rules need to be considered on which we will base the design of our chatbot. Before we start implementing a chatbot web application, it is important to decide what type of chatbot we want to implement: rule-based, self-learning or a mixture of the two. Self-learning chatbots use ML and AI techniques and algorithms to save the inputs from the users and use them later. Rule-based chatbots have predefined rules that they follow. We must know the audience that we are making our chatbot. Keeping in mind the natural language of the users of the chatbot we should create the language that the application should use incorporating it for our target audience. We also need to provide appropriate responses to initial questions to the chatbot. Create a chatbot: This is done using the function. The function will take the name of the bot as an input argument. This function returns an object (called bot) which will be further used in the program. You may set the name of your choice. In our case, we've set it to Hola. Train the chatbot: this is often done using the train all data function. The data we are training for the chatbot is shown here. The input argument of this function is that the bot. Train with custom data We train the chatbot with custom data using the function. The two input arguments of this function are the bot itself, and the custom data we want to train. This custom data takes the first element of the list as the question and the second element as its answer. We train the chatbot with custom data for the language the user has chosen to learn.

III. MODELING AND ANALYSIS

The Analytic Hierarchy Process (Saaty, the way to make a decision: The analytic hierarchy process, Saaty, 1990) is often described as a way of decomposition of unstructured situations into easier ones. This method assigns a worth to every variant supported by the utilization of the subjective evaluation of the pairwise comparison. Subsequent synthesis identifies the choice with the very best priority. The aim of the authors is to incorporate into the model the maximum amount of relevant information as possible, own goals, criteria, and variants, and to work out the optimal solution. The target of our research is to match the Facebook Messenger chatbots suitable for learning within the selected areas. The methodology of the chatbot selection is described in Sections Coding of chatbots, Evaluating the quality of chatbots. The 47 chatbots that were selected were defined as variations of our model. We assigned a rating to every variant (47 chatbots) for all formulated quality

attributes (. The evaluation was conducted as a chat conversation: a single evaluator rated on a 1–100 scale answered by the chatbot. The evaluator had scripted questions for the chatbots, but he had to adapt them during the conversation consistent with the subject or chatbot sort of dialogue (e.g., whether the chatbot was ready to answer open questions or simply used predefined navigation buttons). When the SHD ChatBot is opened, the user is asked to choose between two languages, namely, Spanish and German. After the user chooses the particular language of his preference, he can input his message. The user inputted message is translated into the language that he had previously selected and he gets the translated message back in the form of a chat/text message.

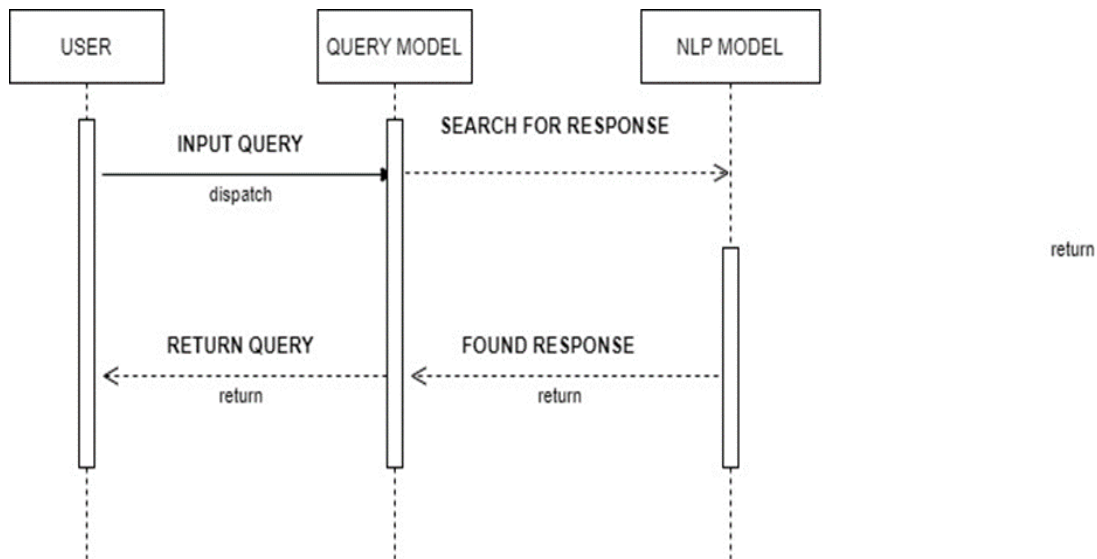


Figure 5: Sequence Diagram of the chatbot.

IV. RESULTS AND DISCUSSION

This application is an educational chatbot of the instant messaging type web-based chatbot with a focus to identify discoverability and characteristics such as language, subject matter, and developer's platform. This application was developed with the help of artificial intelligence for the personalized learning of languages of each student, measuring the amount of usage and progress (understanding of words and phrases) of the chatbot application. Educational chatbots vary from the basic level of sending personalized messages, to recommending learning content. The application created consists of features such as artificial intelligence teaching assistants answering typical queries by learners, setting learning goals, and monitoring learning progress. The fields this chatbot has indulged in are, to focus on the developer's support to create and offer tools that allow any teacher to integrate chatbots into their classes without difficulty, and provide educational chatbot guidelines to successfully support coaching methods and students' learning and content analysis of the actual conversations with students. Our application supports two foreign languages Spanish and German, taught along with English explanations for beginners of the languages. Artificial intelligence is used to train the chatbot for responses and also measure the ability of the students in the usage of the chatbot, extensively shifting the language to a purer conversion in the foreign language of choice of the student.

V. CONCLUSION

The web-based application does provide quite a bit of flexibility in the creation and usage, two different languages have been added to the same chatbot and it is trained for each of them separately to respond by providing the answers in the language chosen by the user. Even though it's a push in a new domain with vastly new ideologies, there are many if not the few advances listed below to be made in the application. As users can't only learn by asking queries from the chatbot, a few quiz rounds, select the correct answer, fill in the blanks, etc. type exercises would be added to the chatbot for enhancing engagement with the application. Even the queries can have better engagement and learning opportunities, in the future along with the results of the query about the words or phrases the application will provide pictures of the result along with two sentences of examples of how the word is used in the chosen foreign language. As artificial intelligence systems are developed further the

accuracy of conversation and the application's ability to judge the question and provide results will become more humane, to create a more natural teaching process. Small application advances such as a higher number of languages and languages with different alphabets such as Japanese can be incorporated. Voice input and recognition and hence voice output with correct pronunciation can be an integral push to a rather natural manner of learning a tongue. The GUI can assimilate tutors along with just the artificial intelligence which can test the system create exercises for students, even create live classes on the website and provide their expert results on queries that the students post on the website for them.

VI. REFERENCES

- [1] Smutný, Pavel & Schreiberova, Petra. (2020). Chatbots for learning: A review of educational chatbots for the Facebook Messenger. *Computers & Education*. 151. 103862. 10.1016/j.compedu.2020.103862.
- [2] Curry, Collette. (2011). *The Design, Production and Implementation of a storytelling chatbot* - MSc thesis 2011. Park, Mina & Aiken, Milam. (2019). Automated Translation Preference with a Bilingual Chatbot. *INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY*. 19. 7502-7509. 10.24297/ijct.v19i0.8343. Borges, Lucas. (2015). Analysis of the Wisconsin Breast Cancer Dataset and Machine Learning for Breast Cancer Detection.
- [3] N. Fatima, L. Liu, S. Hong and H. Ahmed, "Prediction of Breast Cancer, Comparative Review of Machine Learning Techniques, and Their Analysis," in *IEEE Access*, vol. 8, pp. 150360-150376, 2020, doi: 10.1109/ACCESS.2020.3016715.
- [4] A. Nuez Ezquerro, "Implementing ChatBots using Neural Machine Translation techniques," *Treball Final de Grau, UPC, Escola Tècnica Superior d'Enginyeria de Telecomunicació de Barcelona, Departament de Teoria del Senyal i Comunicacions*, 2018.
- [5] Tiha, Anjana. (2018). Intelligent Chatbot using Deep Learning. 10.13140/RG.2.2.14006.75841. Boeri, C, Chiappa, C, Galli, F, et al. Machine Learning techniques in breast cancer prognosis prediction: A primary evaluation. *Cancer Med*. 2020; 9: 3234– 3243. <https://doi.org/10.1002/cam4.2811>