International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol. 9, Issue.7, July 2021 E-ISSN: 2347-2693

Design of Chatbot System for College Website

Shivani Pravin Rashinkar[1*], Neha Dilip Wanjol[2], Shivani Nandkumar Rane[3], Pushkar P. Shinde[4], Sopan A. Talekar[5]

[1,2,3,4,5]Department of Computer Engineering, MVP's KBT College of Engineering, Pune University, Nashik, India

**Corresponding Author: rashinkarshivani173@gmail.com, Phone no.:+918805871670*

DOI: https://doi.org/10.26438/ijcse/v9i7.4145 | Available online at: www.ijcseonline.org

Received: 14/Jul/2021, Accepted: 20/Jul/2021, Published: 31/Jul/2021

***Abstract*—** Most of the time, Students need to visit college administration office to collect various information regarding college such as Tuition fees, Term Schedule, etc. during admission process or as per their daily needs. Hence, to overcome this problem, a chatbot can be designed and developed which can be easily integrated with any college website to provide necessary information regarding college. The goal of AI based chatbot is to make an efficient conversation between human and machine via auditory or textual methods. This project uses Natural language processing to process the user's query and generate a meaningful response. Based on the information stored in the database, bot itself determines appropriate response of a particular query fired by user. The Chatbot is based on an Artificial Intelligence algorithm, which analyses user's question and responds with a Naive Bayes' algorithm. This system will be a Web Application and can reduces work of college administration providing information to students. It also reduces the workload on the staff to answer all the queries of the students.

***Keywords*—** Chatbot, Query, Graphical User Interface, Natural Language Processing, Artificial Intelligence.

I. INTRODUCTION

Chatbot is a computer application that mimics human conversations in its natural format along with textual or voice communication. In the utilization of AI techniques together with natural language processing (NLP) [1],[3] chatbot for college website can be designed. This system will be a web application, so it can give solutions to the analysed queries of the user. User simply need to raise the query to the chatbot and the system will reply to the users through the powerful Graphical User Interface (GUI) which is similar to messaging application interface, and provides a friendly environment to the user as they are much aware of operating messaging application[1],[2]. The user can submit the question about the college-related information or activities such as admission process, contact information, address, annual day, sports day, intake and other cultural activities of colleges. Developing a chatbot solves the problems that can arises in gathering required college information. This system can be accessible from anywhere and anytime. Chatbot will deliver efficient and relevant response to the user corresponding to their entered message. Chatbot system will be beneficial for students, parents, teaching and non-teaching staff as well. Presently, there are various chatbots available for the students like UNIBOT, ALICE etc. UNIBOT is designed for the students to ask university related question. For this system a new algorithm is developed to deliver an appropriate response to the user corresponding to their entered message[2]. ALICE is a rule-based chatbot based on the Artificial Intelligence Markup Language (AIML). This System uses NLP and Pattern Matching Algorithm [3],[6],[7] to process user's query[8].

This paper is divided into several sections, where Section I contains the introduction of chatbot system, Section II contains Related Work of chatbot system, Section III explains the Methodology with architecture diagram and flow chart, Section IV contains Results and Section V describes Conclusion and Future Work.

II. RELATED WORK

K. Bala, M. Kumar, S. Hulawale, and S. Pandit et al. [1] Project on Chatbot for college management is developed with the help of AI algorithms which can analyse user's queries. This is a web application that will give answers to the analysed queries of the user. Users will simply need to select the class for queries and ask the question to the bot. In this paper, they have used Porter Stemmer algorithm to answer the user' queries. The Users should register and login to the system. Once login, user can access the various helping pages through which the user can ask queries related to college activities.

P. Nikhila, G. Jyothi, K. Mounika, Mr.K. K. Reddy et al [2] The chatbot named UNIBOT is designed for the students to ask university related question. This system uses the concept of Artificial Intelligence and Machine Learning. The System uses PHP Language for the development of UNIBOT. The query is given as an input to the algorithm, which processes it and gives the corresponding response to the user. The GUI is similar to a Messaging Application. It delivers efficient and relevant response to the user corresponding to their entered query. New algorithm is developed for UNIBOT. It is very efficient, requires less memory and has minimal database hits.

B.Setiaji and F. W. Wibowo et al. [3] Chatterbot is designed with a powerful pattern matching algorithm. This project uses Indonesian conversational pattern and MySQL database. This application is based on a knowledge base which is maintained by admin. It can be miss in defining a sentence and how to response it while connecting chat application to the database. In the pattern-matching operation, knowledge representation and implementation of SQL are important. A data that has been created which is based on the pattern of the conversation must be tested by the help of a series of scenarios. The conversation should be crosschecked to the basic pattern so that it allows you to add some knowledge to the database which is not added before. If the input record in the database does not match, then it will be remodeled.

K. Shivam, K. Saud, M. Sharma, S. Vashishth, and S. Patil et al. [4] In this paper, for designing chatbot, Facebook Messenger is used which is source and uses artificial intelligence to communicate with the user and provide the required information. This Facebook API is integrated with Python backend, webhook is used to deliver the query of the user to the server. This system has used WIT and AI as a pre-trained artificial intelligence module so that one could use its pre-trained neural networks to answer the user's query with efficiency and accuracy.

E. Haller and T. Rebedea et al. [5] This paper describes the concept of identifying vital facts in texts describing the life of a historical figure for building a conversational agent that could be used in middle-school CSCL scenario. This paper presents a way for building a chat-bot that may simulate a historical figure. The system can receive "input" as an understandable text or a web page about the historical figure and has as "output" a trained conversational agent that is in a position to answer all reasonably questions about the life expertise of that user.

S. B. Sonawane, A. S.Badwar, R. H. Dalvi, G. N. More and S. A. Talekar et al. [6] This paper describes the concept of AI based chatbot which is designed for Student Counselling for career guidance. This system uses NLP and Keyword Matching Algorithm to process user's query. This System uses a modular architecture to respond to user input. Each module contains knowledge based initialization mechanism, and logic to handle user requests. The students are able to get proper guidance for career in the field of their choice, also the college list for the same is provided as per requirement.

III. METHODOLOGY

The proposed chatbot system is a web application which gives reply to the question of the user. This system is utilized for talking. A chatbot project is built using artificial algorithms i.e. Naive Bayes' algorithm that analyse user requests and understand the user's message[9]. The system uses Natural Language Processing (NLP) and built-in artificial intelligence to answer the queries asked by the user. For the chatbot development, we have used Python programming language using Django framework and Chatterbot library. It makes it easy to generate automated responses to a user's input with the help of a machine learning algorithm to produce different types of responses. Students just have to query through the bot which is used for chatting purpose. Chatbot will reply to the query with the help of artificial intelligence.

The proposed system will reduce the administration burden and will be able to provide necessary details to students and parents online. Students do not have to visit college administration for every inquiry. Students will get their queries resolved without any hassle to reach out the college administration office. The System will be available for 24/7 to all students and parents [4].

1. User Login:

User just needs to submit his/her queries regarding the college to the bot. User can be student, teacher as well as parents.

2. Chatbot Responding System:

2.1 NLP Processing for Query Analysis :

When a user submitted the query to the system, NLP is applied and sense of the query is detected. Natural Language Processing (NLP) allows chatbot to understand user's messages and respond appropriately.

2.2 Search Questions in database:

Once the query is submitted, furthermore, we have to search the exact answer of the query in our pre-trained database. As the query description can change from person to person. The same question can be asked in different ways from multiple users. One user asks a question so simply and clearly, while another user may ask the same question with different format. So it is necessary to find out what is the correct solution of a submitted query.

2.3 Answer the Queries:

As described above, whenever user submits a complaint, then it is checked that is there such question registered in database. If the answer is matched, then that is sent to that User. If the answer of submitted query is not found in the database, then such questions are answered by admin person. Once he answered the query, the answer is sent to that user. And also it is stored in database so that whenever such questions will be asked they get answer of that query directly from the database. So admin doesn't need to answer the same query again and again. For this we are using powerful Naive Bayes' Algorithm[9].

3. Architecture Diagram of Chabot System:

Figure 1: shows the architecture of the chatbot system for college website.

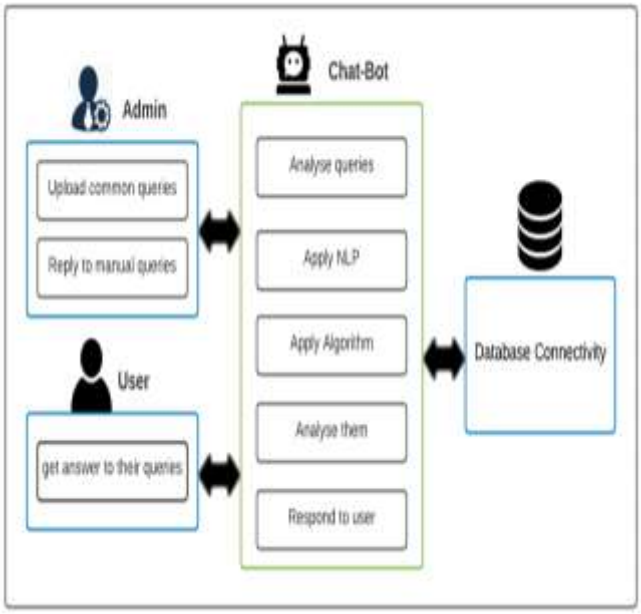


Figure 1: Architecture Diagram of Chatbot System

4. Flowchart of Chatbot System:

The flowchart of the system displays how the chatbot performs. Initially, the user message is pre-processed and connectivity to the database is obtained. Then, based upon conditions satisfied, the chatbot process flows and provide response to the user. If the user cannot find the answer for a query then in such condition chatbot will provide admin's contact details to the user. Admin can view and answer the corresponding query.

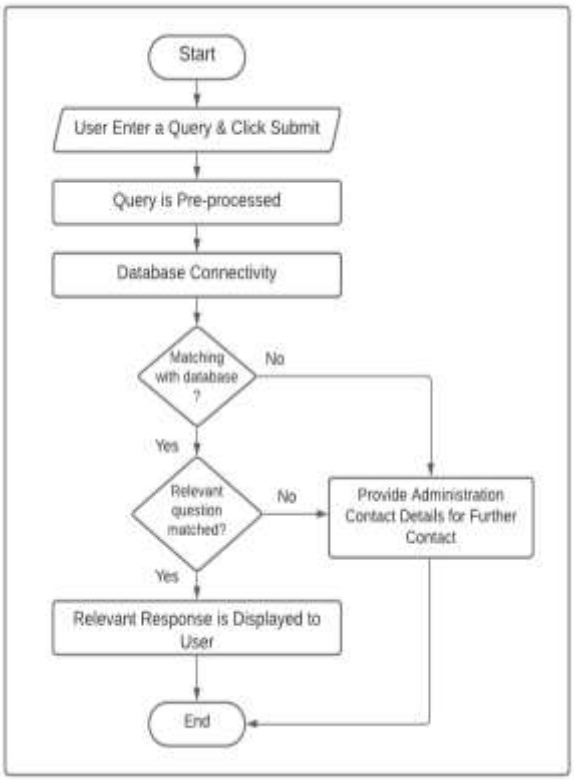


Figure 2: Flowchart of Chatbot System

5. Algorithm Implemented

Naive Bayes' is a powerful algorithm for text classification problems. It is a probabilistic machine learning algorithm which is based on Bayes' theorem[9]. This classifier assumes that the presence of a selected function in a category is unrelated to the presence of another function. In this algorithm a closed domain dataset containing questions/user-responses and corresponding answers is made, which every question is given a label, this will relate the question to its answer. Due to multiple questions could have the same response, there can be multiple questions having the same answer.

The Formula for Naive Bayes' Algorithm is as follows:

$$P(A \mid B) = \frac{(P(B \mid A) * P(A))}{P(B)} \quad (1)$$

Where,

P (A | B) = Probability of 'A' occurring given evidence of 'B' has already occurred.

P (B | A) = Probability of 'B' occurring given evidence of 'A' has already occurred.

P (A) = Probability of 'A' occurring.

P (B) = Probability of 'B' occurring.

6. Different Algorithms Used for Chatbot System:

Some of the most popular algorithms for chatbots are Porter Stemmer[1], Naive Bayes'[9], Support Vector Machines, K-means and atural language processing (NLP). Chatbots mainly use classification algorithms to recognize intent in phrases. Every algorithm has its own advantages and disadvantages according to its working method. Table 1 shows comparison of different algorithms WRT to Naive Bayes' algorithm.

Table 1: Comparison of Different Algorithms

Sr. No.	Porter-Stemmer Algorithm	K-Means Clustering Algorithm	Naive Bayes' Algorithm
1.	The stems generated are not always real words.	Different partitions can result into different final clusters.	Naive Bayes' classifier performs better than other models with less training data if the assumption of independence of features holds.
2.	It has five steps and sixty rules. And takes more time.	Difficult to predict K-Value, hence it is time-consuming.	The algorithm works very fast and can easily predict the class of a test dataset.

3.	This System is limited to English words only.	Doesn't support categorical data.	The algorithm performs well with categorical variables in comparison to numerical variables.

IV. RESULTS AND DISCUSSION

The Chatbot is to carry out a conversation between both human and machine. Figure 3: shows the simple and attractive pop-up Interface for College Chatbot System. Firstly, chatbot will print a welcome message then a user can type and submit the query and bot will provide an appropriate answer to the user's query. In figure 4 'What is college full name?' this question is asked by the user and correct answer is given by the chatbot. The User can query any college related activities through the chatbot system.

Figure 3: Chatbot Interface-1

Figure 4: Chatbot Interface-2

V. CONCLUSION AND FUTURE SCOPE

The aim of the system is to provide a user-friendly and efficient chatbot system for College. The chatbot will be very useful in guiding students to get correct and up-to-date information source. This system will be fruitful for students, teachers as well as parents. They can get information at any time without having to visit the college administration office every time. In the future scope of this project, the system will include voice-based requests and responses. Users are required to provide voice input and the system will output in the form of text. Chatbot will be able to provide a voice output as well, with the help of text to speech or speech to text conversion.

REFERENCES

[1] K. Bala ,M. Kumar, S.Hulawale, and S. Pandita, *"Chat-Bot For College Management System Using A.I"*, International Research Journal of Engineering and Technology (IRJET), Vol. **04**, Issue.**11**, pp.**2395-0072**, **2017**.

[2] P. Nikhila, G. Jyothi, K. Mounika, Mr.K. K. Reddy and Dr. B.V. Ramana Murthy, "*AI and Web-Based Human-Like Interactive University Chatbot (UNIBOT)",*In theProceedings of the Third International Conference on Electronics Communication and Aerospace Technology[ICECA 2019], pp.**1-12,2019**.

[3] B.Setiaji, F. W. Wibowo, "*Chatbot Using A Knowledge in Database- Human-to-Machine Conversation Modeling*", In the Proceedings of the 2016 International Conference on Intelligent Systems, Modelling and Simulation, pp.**2166-0670**, **2016**.

[4] K. Shivam, K. Saud, M. Sharma, S. Vashishth, and S. Patil, "*Chatbot for College Website*", International Journal of Computing and Technology, Vol.**5**, Issue.**6**, pp.**2348-6090, 2018**.

[5] E. Haller and T. Rebedea, "*Designing a Chat-bot that Simulates a Historical Figure*", Faculty of Automatic Control and Computers university of Bucharest, IEEE 978-0-7695-4980-4/13, **2013**.

[6] S. B. Sonawane, A. S.Badwar, R. H. Dalvi, G. N. More and S. A. Talekar, "*Design of Chatbot System for Student Counselling*", International Journal of New Innovations in Engineering and Technology, Vol.**13**, Issue.**3**, pp.**2319-6319**, **2020**.

[7] P. Jain,"*College Enquiry Chatbot Using Iterative Model*", International Journal of Scientific Engineering and Research (IJSER),Vol.**7**, Issue.**1**, pp.**2347-3878,2019**.

[8] N. Hatwar, A. Patil and D. Gondane, "*AI BASED CHATBOT*", International Journal of Emerging Trends in Engineering and Basic Sciences (IJEEBS) ISSN (Online) 2349-6967 Volume **3**, Issue **2** (March-April **2016).**

[9] A. Prajapati, P. Naik, S. Singh and A. Kale, "*Android Based Chatbot For College*",International Journal of Scientific & Engineering Research, Vol.**9**, Issue.**4**, pp.**2229-5518**, **2018**.

[10] A. Ohm, K. Bhavani,"*Chatbot for Career Guidance Using AI*", International Journal of Computer Sciences and Engineering (IJCSE), Vol.**7**, Issue.**6** ,pp.**2347-2693**, **2019**.

[11] K. G. Yohandi, M. I. Sani, "*Designing a Mobile Chatbot For Elementary School Vocabulary*", International Journal of Computer Sciences and Engineering(IJCSE), Vol.**8**, Issue.**7**, pp.**2347-2693**, **2020**.

AUTHORS PROFILE

Shivani Pravin Rashinkar, Bachelor of Engineering in Computer Science, MVP's KBT College of Engineering, Nashik, India

Neha Dilip Wanjol, Bachelor of Engineering in Computer Science, MVP's KBT College of Engineering, Nashik, India

Shivani Nandkumar Rane, Bachelor of Engineering in Computer Science, MVP's KBT College of Engineering, Nashik, India

Mr. Pushkar P. Shinde, pursed Bachelor of Engineering in Computer Science and Master of Engineering in Computer Science. Completed Diploma in Advance Computing, CDAC. He is currently working as Assistant Professor in Department of Computer Engineering, MVPS's KBTCOE, Nashik, India. He has published & presented more than 06 research papers. His research interest is Machine Learning. He has 16 years of teaching experience.

Mr. Sopan A. Talekar, pursued Bachelor of Engineering from North Maharashtra University Jalgaon, Maharashtra in 2004 and Master of Technology from Visvesvaraya Technological University, Belagavi, Karnataka in year 2010. He is currently pursuing Ph.D and currently working as Assistant Professor in Department of Computer Engineering, MVPS's KBTCOE, SPPU, Pune, since 2005. He is a life member of IE & ISTE. He has published & presented more than 07 research papers. He received prestigious Promising Engineer Award from the Institution of Engineers (India), Local center Nasik, in 2013. His research interest is Wireless Network, Mobile Ad hoc Network, Cognitive Radio Network & Artificial Intelligence. He has 16 years of teaching experience.

IJCAT - International Journal of Computing and Technology, Volume 5, Issue 6, June 2018
ISSN (Online) : 2348-6090
www.IJCAT.org
Impact Factor: 0.835

Chatbot for College Website

[1] Kumar Shivam; [2] Khan Saud; [3] Manav Sharma; [4] Saurav Vashishth; [5] Sheetal Patil

Department of Computer Engineering, Bharati Vidyapeeth
Deemed To be University College of Engineering, Pune

Abstract : A chatbot is a software that is used to interact between a computer and a human in natural language like humans chat. Chatbots chat with the user in a conversation in place of a human and reply to the user. The goal of this report on chatbot was to resemble a human being in the way they interact, trying to make the user think he is chatting with another human being. The chat bot application helps the students to access the university related information from anywhere with internet connection. This system reduces work of college administration providing information to students and also reduces the workload on the staff to answer all the queries of the students.

Keywords : *Chatbot, Artificial Intelligence, Enquiry, AIML, Response , Query.*

1. Introduction

Chat Bot is a computer program that can talk to humans in natural language, the way we interact with each other. It can replace a human for many tasks of answering queries. A chatbot is an agent that interacts with users using natural language. It was built as an attempt to fool humans. Several applications of chatbots such as Customer Service, call centers etc. uses Artificial Intelligence Markup Language to chat with user.

One of the prime goals of chatbots is to resemble an intelligent human and make it difficult for the receiver of the conversation to understand the real working along with various architecture and capabilities for their usage has widely broadened.

These chatbots can prove sufficient to fool the user into believing they are "talking" to a human being, but are very limited in improving their knowledge base at runtime, and have usually little to no means of keeping track of all the conversation data .

Chatbots makes use of machine learning to reach artificial intelligence helping them to understand the user query and provide an appropriate response. The chatbots are developed using the Artificial Intelligence Markup Language for communicating or interacting with the user. This consist a software which will be made up using Artificial Intelligence and will help user to chat with machine.

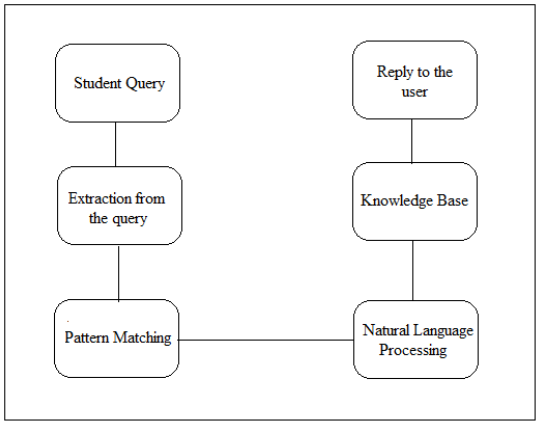


Fig.1 : Data Flow Diagram

2. Literature Review

There are numerous applications that are consolidating a human appearance and proposing to reproduce human exchange, yet in the majority of the cases the information of the conversational bot is put away in a database made by a human specialists. Be that as it may, not very many inquires about have explored making a talk bot with a fake character and identity beginning from website pages or plain content about someone in particular. This paper portrays a way to deal with recognizing the most critical realities in writings depicting the life of an authentic figure for building a conversational operator that could be utilized as a part of center school CSCL situations

This paper portrays an adaptable technique for educating initial counterfeit consciousness (AI) utilizing a novel, Python-executed, basic operator system grew particularly for the reasons for this course. Albeit various operator systems have been proposed in the huge collection of writing, none of these accessible structures ended up being sufficiently basic to be utilized by fourth-year understudies of software engineering. structure that would be reasonable for the points of the course, for the level of registering aptitudes of the planned gathering of understudies, and for the span of this gathering of understudies. The substance of the initial AI course being referred to is an arrangement of assignments that requires the understudies to utilize keen specialists and other AI systems to screen, channel, and recover important data from the World Wide Web. It speaks to, in this way, an amalgamation of the customary objectivist approach and a certifiable situated, constructivist way to deal with instructing programming to amateurs. The fundamental point of executing such an instructional method was to connect with the understudies in figuring out how which they by and by relate while achieving scholarly meticulousness. Classroom encounter shows that understudies take in more viably when the conventional objectivist approach is joined with a constructivist approach than when this standard way to deal with instructing programming to amateurs is utilized alone.

Manmade brainpower machines are made to carry on in wondrous ways, frequently adequate to stun even the most experienced onlooker. Be that as it may, once a specific program is unmasked, once its inward workings are clarified ... its enchantment disintegrates away; it stands uncovered as a simple accumulation of systems ... The eyewitness says to himself "I could have composed that". With that idea he moves the program being referred to from the rack stamped "savvy", to that held for doodads ... The protest of this paper is to cause simply such a re-assessment of the program going to be "clarified". Hardly any system at any point required it more.

3. Proposed System

This System is a web application which gives reply to the question of the user. This system simply need to question through the bot which is utilized for talking. The System utilizes worked in counterfeit consciousness to answer the inquiry. The appropriate responses are suitable what the client questions. In the event that the appropriate response found to invalid, client simply need to choose the invalid answer catch which will tell the administrator about the inaccurate answer. Administrator can see invalid answer through entrance by means of login System permits administrator to erase the invalid answer or to include a particular answer of that comparable inquiry. The User can inquiry any college related exercises through the framework. The client does not need to go to the college for enquiry always. The System examines the inquiry and after that responses to the client. The framework answers to the inquiry as though it is replied by the individual. With the assistance of computerized reasoning, the framework answers the question asked by the understudies. The framework answers utilizing a compelling Graphical UI which infers that as though a genuine individual is conversing with the client. The client can question about the school related exercises through online with the assistance of this web application. This framework causes the understudy to be refreshed about the school exercises.

The proposed system will simply take the query of the user which can be a student or a parent, and will give response according to the query. The system will match the user query with knowledge base and see for the appropriate response. The system can also reply to the general queries of the student. The algorithm of the complete system can be understood by the following flow chart.

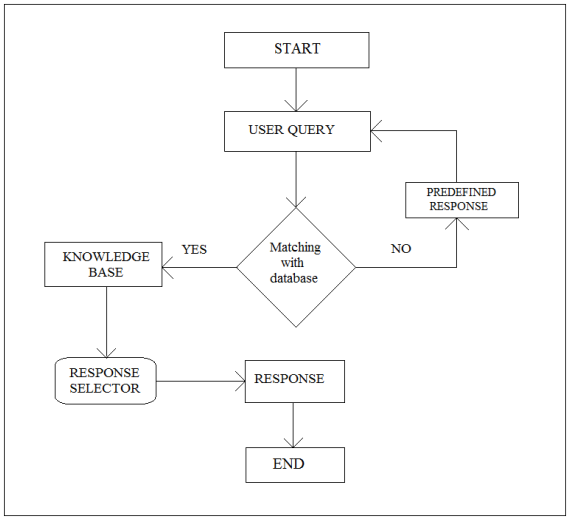


Fig 2 : Flow Chart of Proposed Model

4. Discussion of Current Scenario

- Chat bot system is unknown to people who are not updated with the technology.

- Even if there exist a chat bot system, it is not much accurate in proving the answer or solutions.
- Students need to manually visit to the college to get their queries answered by the college help desk.
- This process consumes lot of time as well as money as the customer needed to visit college if its miles away from home.
- Also, this process may lead to communication gap between student and college.
- No LIVE update platform yet for all student and parent queries
- No alternative available for student and parents beyond visiting college administration.

4.1 Objectives of this system:

- Reduce management effort.
- Provide necessary details to student and pa rent online.
- Provide information about college activities and schedule of current academic year.
- To get rid of manual efforts.
- To reach college administration easily.
- Reducing visit to college administration for every enquiry.
- 24x7 availability for all student and parent queries.
- Creating a situation of delight for parents and student with extra technical support.
- Students will get their queries resolved without any hassle to reach out the college administration.
- This application enables the students to be updated with college cultural activities.
- This application saves time for the student as well as teaching and non-teaching staffs.

5. Data and Results

We have created an application with the help of Facebook messenger API which is open source and uses artificial intelligence to interact with user and provide the desired information. This Facebook API is integrated with Python backend, webhook is used to deliver the query of the user to the server. We have used WIT.AI as a pre-trained artificial intelligence module so that we could use it's pre-trained neural networks to answer the user's query with efficiency and accuracy. We also made some custom modules/entities such as calendar, time-table etc. in the wit.ai module to make it ample to answer college related queries. This application would be available on the college website, for this purpose we have created a static web page to mimic the college website. This webpage is built using html with bootstrapping and design part is done with help of html css.

5.1 Test Cases:

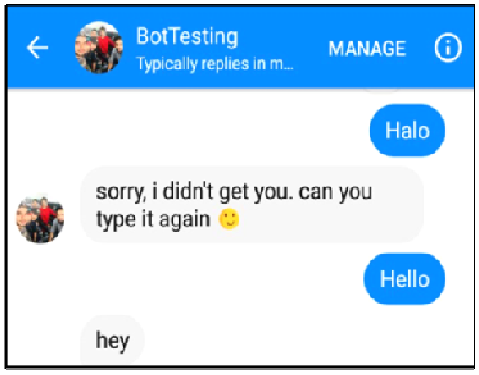


Fig 3: Test Case 1

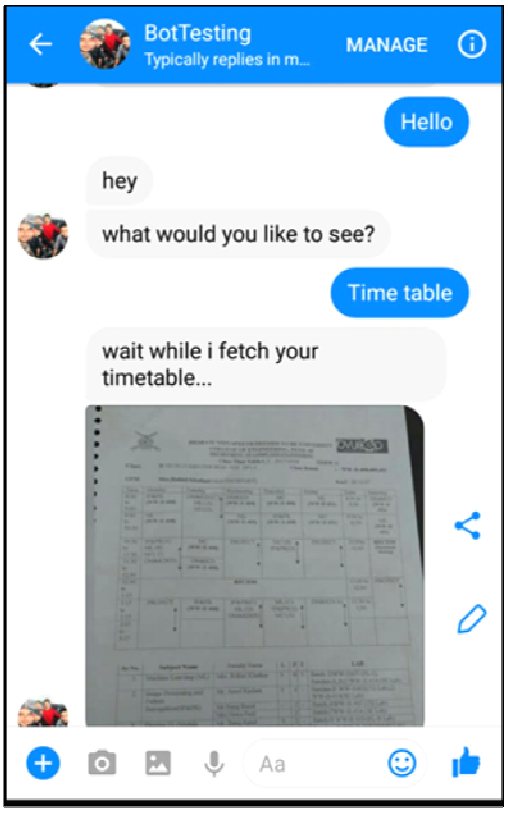


Fig 4: Test Case 2

6. Conclusion

The main objectives was to develop an algorithm that will be used to identify answers related to user submitted questions. To develop a database were all the related data will be stored and to develop a web interface. The web interface developed had two parts, one for simple users and one for the administrator. A background research took place, which included an overview of the conversation procedure and any relevant chat bots available. We created a database , which stores all the information about questions, answers, keywords, logs and feedback messages. A usable system was designed, developed and deployed to the web server.

References

[1] https://chatbotconf.com.ua/en/article/top-3-glavnih-nedostatkov-chat-botov-66332
[2] The Anatomy of A.L.I.CE : Dr.Richard S.Wallace http://www.alicebot.org/anatomy.html
[3] Artificial Intelligence Markup Language (AIML), A.L.I.C.E. AI Foundation, http://alicebot.org/TR/2001/WD-aiml/
[4] http://developers.facebook.com
[5] AIML Interpreter Overview 2004, http://www.aimlbots.com/en/aiml-interpreters.html
[6] Computing machinery and intelligence, Alan Turing [1950], http://www.abelard.org/turpap/turpap.htm
[7] Using Dialogue Corpora to Train a Chatbot (Bayan Abu Shawar, Eric Atwell) http://www.comp.leeds.ac.uk/andyr/research/papers/techreport2003_02

IMPLEMENTING A COLLEGE ENQUIRY CHATBOT

A Project

Presented to the faculty of the Department of Computer Science

California State University, Sacramento

Submitted in partial satisfaction of
the requirements for the degree of

MASTER OF SCIENCE

in

Computer Science

by

Ujaliben Kalpesh Bavishi

SPRING
2019

IMPLEMENTING A COLLEGE ENQUIRY CHATBOT

A Project

by

Ujaliben Kalpesh Bavishi

Approved by:

_____, Committee Chair
Dr. Pinar Muyan-Ozcelik

_____, Second Reader
Dr. V. Scott Gordon

_____
Date

Student: Ujaliben Kalpesh Bavishi

I certify that this student has met the requirements for format contained the University format manual, and that this project is suitable for shelving in the Library and credit is to be awarded for the project.

_____, Graduate Coordinator _____

Dr. Jinsong Ouyang Date

Department of Computer Science

Abstract
of
IMPLEMENTING A COLLEGE ENQUIRY CHATBOT
by
Ujaliben Kalpesh Bavishi

This project is focusing on creating a chatbot to be used by students to get their queries responded easily from the college website. The College Enquiry Chatbot has the capacity to make friendly conversations; respond the course and faculty details; give the link for the academic calendar; answer the frequently asked questions; calculate the fees based on the student's input; and give the timings, address, contacts, and events information of the departments like Union, Library, IPGE, and AIRC. To build the chatbot, Microsoft Azure bot service as well as Microsoft cognitive services, namely, Text Analytics, LUIS, and QnA Maker are used.

Most of the existing chatbots lack empathy and fail to accommodate anything outside of the script. In order to address these problems, the College Enquiry Chatbot extends the implementation of the current chatbots by adding sentiment analysis and active learning. Although, sentimental analysis correctly recognizes the user's query as positive, negative and neutral, the system was partially successful in adding empathy to the chatbot. It is because the system requires more rigorous training data to handle all queries which are off-script. However, for such queries, active learning helps to improve the chatbot

performance since it correctly understands the user's questions, asks clarifying question, and then retrains the system to give the response what the user intends to get.

The future work include training the chatbot with more varied data; increasing the scope of the chatbot by adding a speech recognition feature so that users can speak to get responses; and including integration with multiple channels such as phone call, SMS, and various social media platforms.

_____ Committee Chair
Dr. Pinar Muyan-Ozcelik

_____
Date

ACKNOWLEDGEMENTS

I would like to express my gratitude towards Dr. Pinar Muyan-Ozcelik who has been a precious help in my voyage of finishing this undertaking. Without her assistance, direction, and support, making this project a reality would have been unimaginable. Her opportune criticism was tremendously essential for me and helped me greatly to reach the targets set for the project tasks. Likewise, I would also like to express gratitude toward Dr. V. Scott Gordon for his commitment and supporting this project as my second reader. In conclusion, I would like to thank the personnel and staff of the Department of Computer Science Engineering at California State University, Sacramento, for their ceaseless regulatory help.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Figures Page

1 INTRODUCTION AND MOTIVATION

This project is focusing on creating a chatbot to be used by students to get their queries responded easily from the college website. A chatbot is a program which can do real conversations with textual and/or auditory methods [1]. Using Artificial Intelligence (AI), chatbots can simulate human conversations. There are two categories of chatbots. One category is command based chatbots where chatbots rely on a databank of replies and heuristics. The user must be very specific while asking the questions so that the bot can answer. Hence, these bots can answer limited set of questions and cannot perform function outside of the code. The other category is chatbots based on AI or machine learning algorithms, these bots can answer ambiguous questions which means the user do not have to be specific while asking questions. Thus, these bots create replies for the user's queries using Natural Language Processing (NLP).

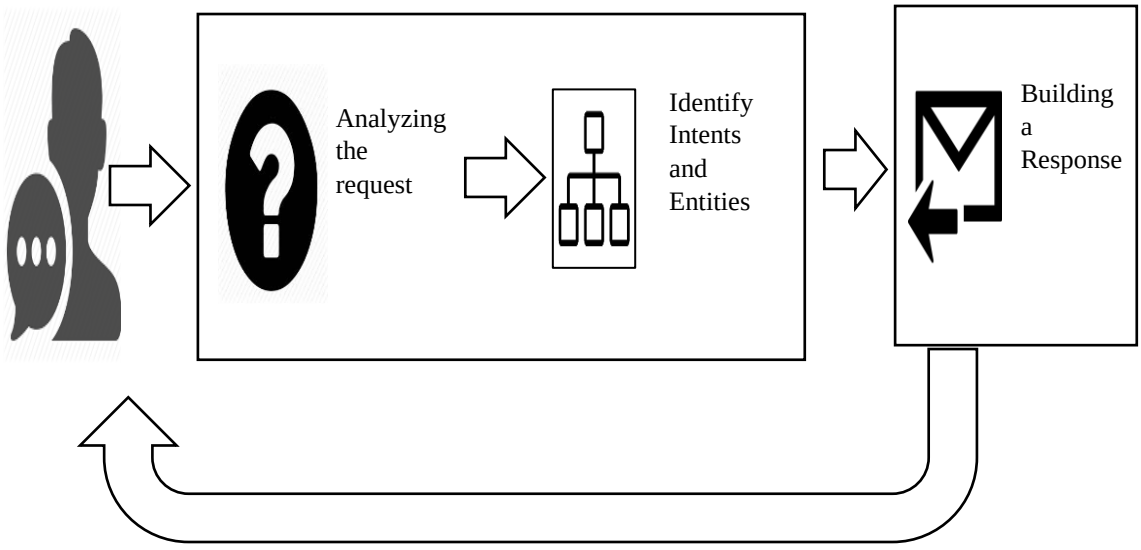


Figure 1: How a Chatbot Works [2]

Figure 1 shows how a chatbot works. Whenever a user asks any query, the bot will first analyze the request, then identifies intents and entities, builds a response and sends it back to the user. Now, intents mean intention of the query and entity means details of that query. For example, if a student wants to know the office hours of a faculty then the intent will be office hours and entity will be name of the faculty in this case.

AI-powered chatbots are motivated by the need of traditional websites to provide a chat facility where a bot is required to be able to chat with user and solve queries. When live agent can handle only two to three operations at a time, chatbots can operate without an upper limit which really scales up the operations. Also, if any school or business is receiving lots of queries, having a chatbot on a website takes off the load from support team. Having a chatbot clearly improves the response rate compared to human support team. In addition, since millennials prefer live chats over a phone call, they find a chatbot, which provide a highly interactive marketing platform, very attractive. Furthermore, a chatbot can automate the repetitive tasks. There can be some scenarios where a business or school receives same queries in a day for many times and support team must respond to each query repetitively. Lastly, the most important advantage of having a chatbot is that it is available 24/7. No matter what time it is, a user can get a query solved. All these advantages of a chatbot constitute the motivation to implement a College Enquiry Chatbot.

Before implementing College Enquiry Chatbot, various existing chatbots were reviewed such as Amazon Shopping App [3], Alexa [4], Bank of America (Erica bot) [5]

and CNN news bot [6]. In order to understand the requirement of a chatbot, consider an example of Amazon Shopping App. In this app, when a customer buys an item, he/she does not have any information about how to return the item. To get this information, the customer must call and wait to talk to customer representative for a long time. However, this whole process is tedious for a customer. Hence, Amazon created a chatbot to answer simple queries of customers.

Similarly, the College Enquiry Chatbot is designed to help students to get their queries solved on a fingertip. The most essential downside I found while utilizing the previously mentioned chatbots is absence of personality and conversational flow. As Storman [7] suggests, the CNN chatbot neglects to give sympathy and effortlessness. To be efficient, the chatbot must have the capacity to relate and associate itself with the user. For example, a discussion with the CNN chatbot is depicted in the article where at whatever point a user says anything aside from news or any current alternatives, it answers with the news comprising of those words and toward the end it says "Not sure I understand what you're looking for. Try again or pick one of the options below." and afterward it gives the choices as programmed. This implies, although it conveys the news proficiently, the CNN bot needs compassion.

Solution to this problem is described by Rahman et al. [8]. This study proposes that “there is a need to understand and consider the stability, scalability, and flexibility issues along with high level of intention of a human language”. Hence, for implementing a chatbot that handles complicated queries, the sentiment analysis is incorporated into College

Enquiry Chatbot. Sentiment analysis aims to obtain writer's feelings expressed in positive, negative or neutral comments. Based on sentiment analysis, the bot is trained to have empathy while answering to the user. For example, if a user says "I am sad today." then bot should reply to it with some empathy like "I'm sorry to hear that, how can I help you today?" and not just reply the standard message like "Sorry did not understand your question."

Another downside which was found during a research on chatbot is that bots are created in such way that they follow a specific route and mostly all of them fails to satisfy anything outside of the previously defined scripts. This means that if they are not part of a predefined scripts, a significant number of the bots will fail in understanding even the most fundamental kind of queries, which results in a repeating and horrendous experience. To resolve this issue, active learning can be introduced to the system to make probabilistic assessments and provide autonomous responses to the users [9]. Active learning is an algorithm which interactively queries user to obtain the desired output. Whenever a user asks anything which is outside of the script the chatbot will ask questions to the user by giving two to three options and based on the user's input, the bot returns the answer to that query. This whole learning process is called as active learning.

2 SYSTEM DESIGN

System design of College Enquiry Chatbot consists of integration of multiple technologies. This system makes use of NodeJS for backend and ReactJS for frontend. Microsoft Azure provides services like Cognitive Service, Bot Service [10], and Web Apps. Cognitive services consist of Text Analytics, LUIS [11], and QnA Maker [12]. The bot service is used to create the bot application on Azure. In addition, Web Apps are used to host the application on Microsoft Azure. System uses Mongo DB to store the conversations history.

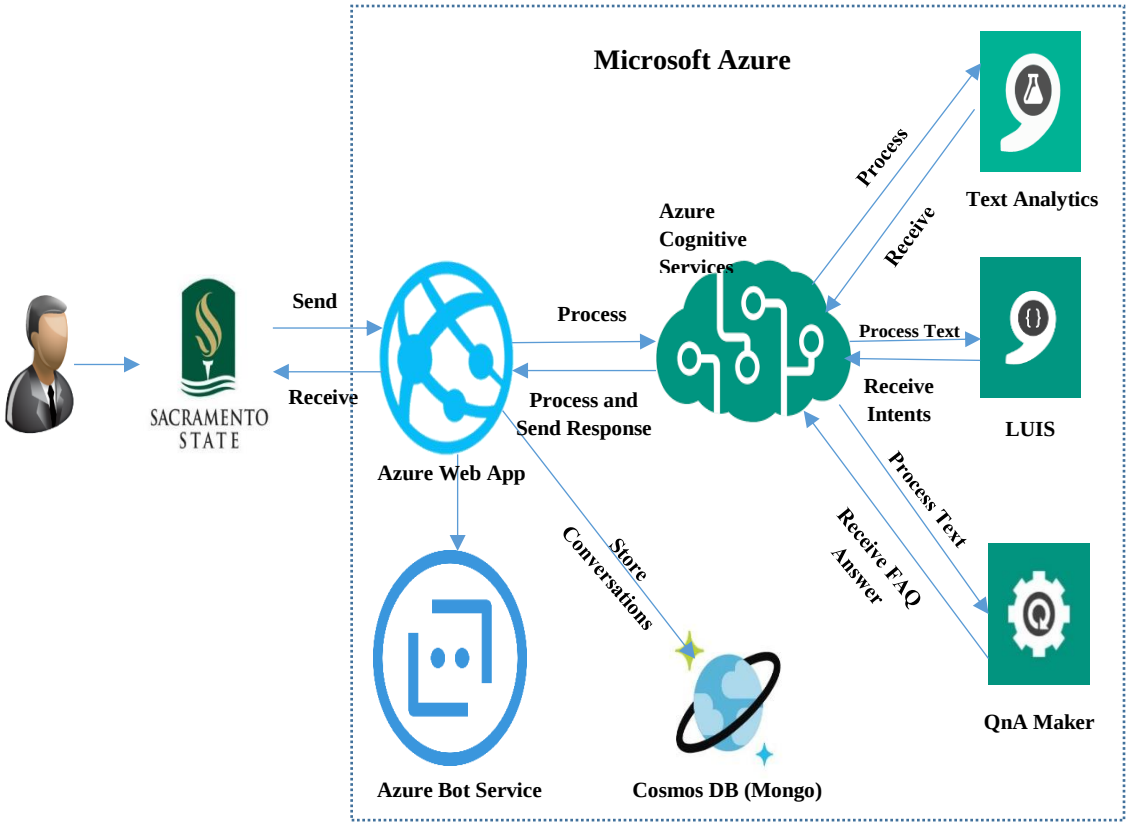


Figure 2: System Design

2.1 Microsoft Azure

Microsoft Azure provides cloud services to build, manage, and deploy applications on a cloud network which helps organization to meet their business needs using their favorite tools and frameworks. College Enquiry Chatbot uses all the services shown in Figure 2 which are provided by Microsoft Azure.

2.1.1 Azure Bot Service

Azure bot service is a service provided by Microsoft Azure which builds and artificial intelligence (AI) chatbot. It offers the ability to add intelligent agents that can do conversations with users without having to commit the resources to develop one's own AI [13].

2.1.2 Azure Web App

I have used Azure web app to host my bot application build by Azure bot service. Azure Web App are the web applications which are hosted on Microsoft Azure cloud without any programming language barrier. Since it is hosted on the cloud, infrastructure maintenance is not needed to host the applications. It also provides various features like auto scaling, automated deployment and also supports both Windows and Linux environments [14].

2.1.3 Azure Cognitive Services

Azure Cognitive services are used to add intelligence to the bots by adding features such as sentiment analysis and language understanding which help in analyzing user's queries [15]. These services can be added to an existing bot by adding service calls to the

Microsoft's SDK to get the desired results. College Enquiry Chatbot utilizes Text Analytics, LUIS, and QnA Maker services which are part of cognitive services and use natural language processing (NLP) for different purposes.

2.1.3.1 Text Analytics

Text Analytics is used to perform sentiment analysis on a user's queries. As shown in Figure 2, a text is sent to text analytics and it returns the sentiment of that text.

2.1.3.2 LUIS

"LUIS stands for Language Understanding Intelligent Services which aims at creating cloud-based machine learning language understanding (LU) models specific to an application and without machine learning expertise" [16]. To implement a College Enquiry Chatbot, I have first created all the possible intents and entities using LUIS tool. Based on these intents and entities, LUIS have built and train a LU deep learning model. Then, the created model needs to be pushed to the endpoint on the cloud. The queries from the user are passed to the endpoint to get the recognized intent and entities in JSON format. Based on various flows, response will be sent back to the user.

Active learning is also implemented using LUIS as shown in Figure 3. LUIS detects the user's query (utterances) in terms of intents and entities. It sends back the response to the service code where those utterances are examined based on the confidence score. Then, College Enquiry Chatbot labels these utterances, re-trains, and publishes the LU model [17]. Another method of active learning is also implemented which involves using user's response to re-train the model. The endpoint provided by LUIS responds with intents,

entities, and a corresponding confidence score. A threshold is maintained based on the confidence score and correspondingly prompts are provided to the user. When the users respond to it, the system keeps track of the responses and uses it to re-train the model.

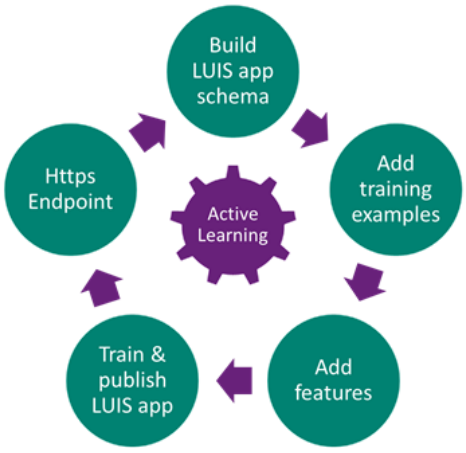


Figure 3: Active Learning Process [16]

2.1.3.3 QnA Maker

For answering simpler queries, instead of detecting intents and entities, College Enquiry Chatbot uses QnA Maker for structuring answers from a semi-structured document like FAQs. This web-based service is useful in incorporating multiple FAQs from an URL, structured documents, product manuals, or editorial contents; and building the LU model for natural language processing.

2.1.4 Cosmos DB (Mongo)

Cosmos DB is a database service provided by Microsoft Azure. College Enquiry Chatbot uses this database service to store my conversational history.

3 IMPLEMENTATION DETAILS

In this section of the report, step by step demonstration of how to setup the bot, LUIS (for NLP), and QnA Maker as well as explanation of service code which includes the implementation of the Active Learning are provided. In addition, setting up databases to store the conversation history is explained. All these steps serve as a template to get a basic bot deployed and published on Azure cloud.

The followings are couple of the prerequisites that are needed in order to get started:

- Microsoft Azure Student Account
- Installation of Git (https://git-scm.com/download/win) and NodeJS (https://nodejs.org/en/download/) on Windows.

3.1 Azure Bot Setup

Open https://portal.azure.com as shown in Figure 4 and follow the steps explained in the following subsections.

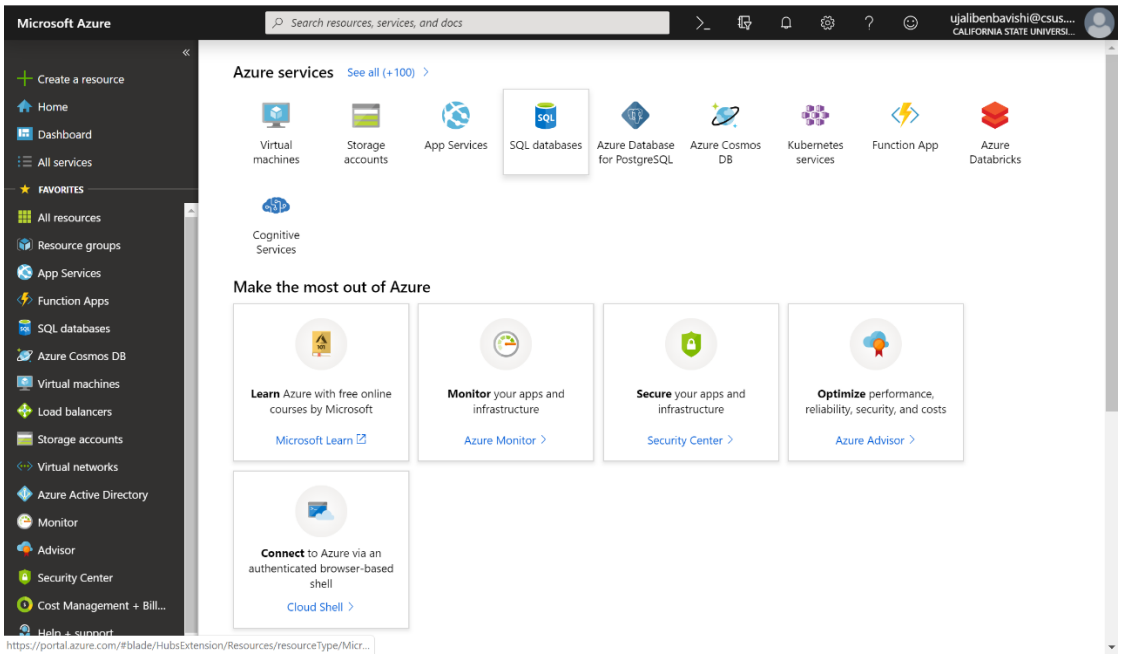


Figure 4: Azure Portal Home Page

3.1.1 Bot Creation

In the left side pane, click on “Create a Resource” and search for “Web App Bot”. As a next step click on “Create” to start creating a bot. Fill up all the required fields as shown in Figure 5.

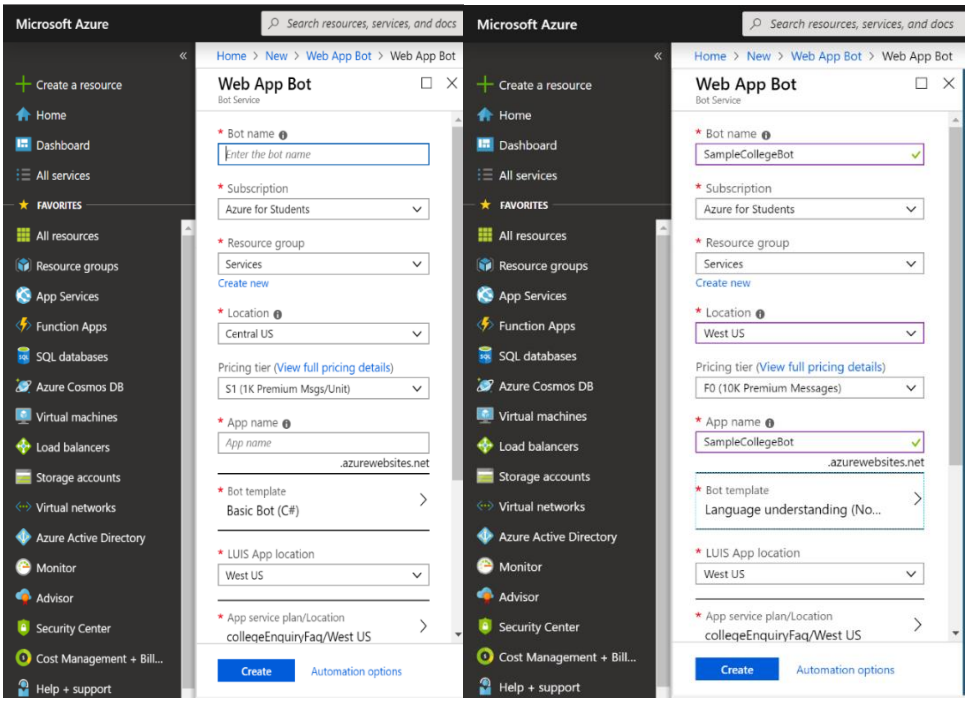


Figure 5: Web App Bot Creation

3.1.2 Adding the DirectLine Channel

After the bot has been created, several channels can be configured to access the bot such as Skype, Webchat, Slack, Facebook, etc. In this project, DirectLine channel is used to communicate with the bot. Figures 6 and 7 show how to add a channel to a bot. After the channel is added, copy the secret key to be used in the service and click on “Done”.

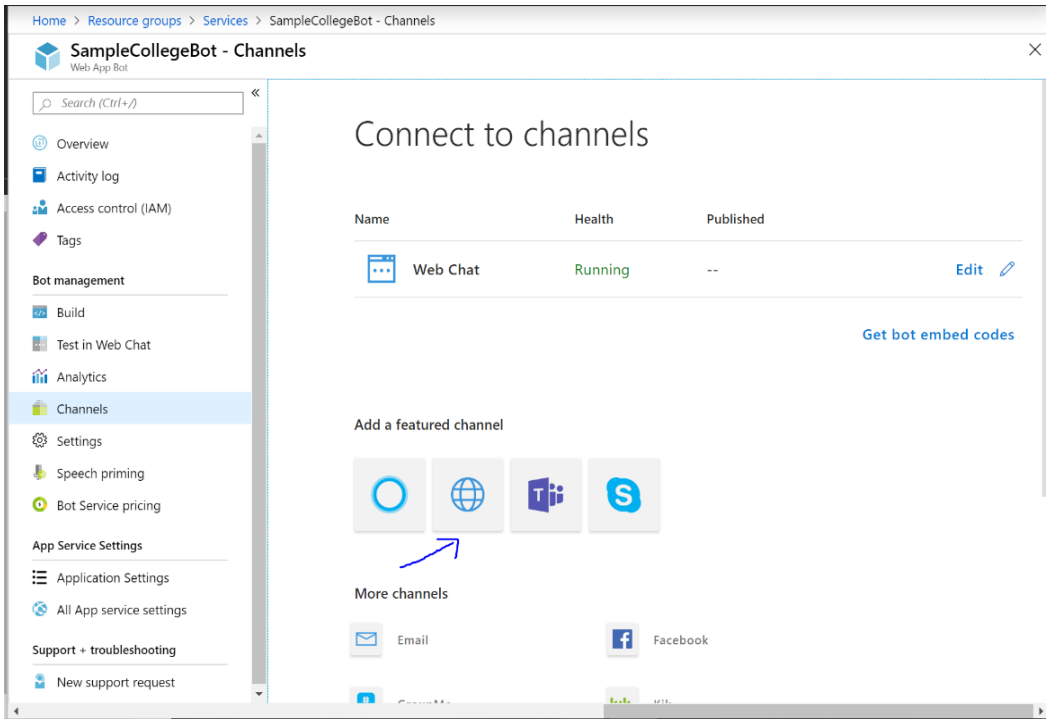


Figure 6: Channel Configuration Page of the Bot

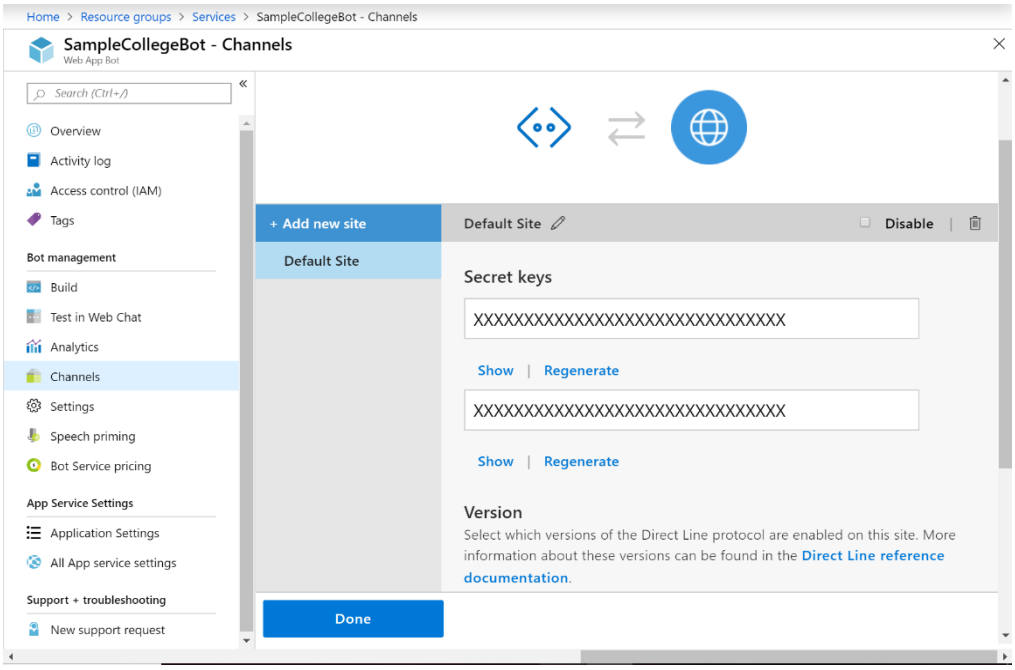


Figure 7: Configure DirectLine Channel for the Bot

3.1.3 Testing the Echo Bot

After the channel is configured, we can test the new bot with some preconfigured bot template where we can echo back whatever the user queries as shown in Figure 8.

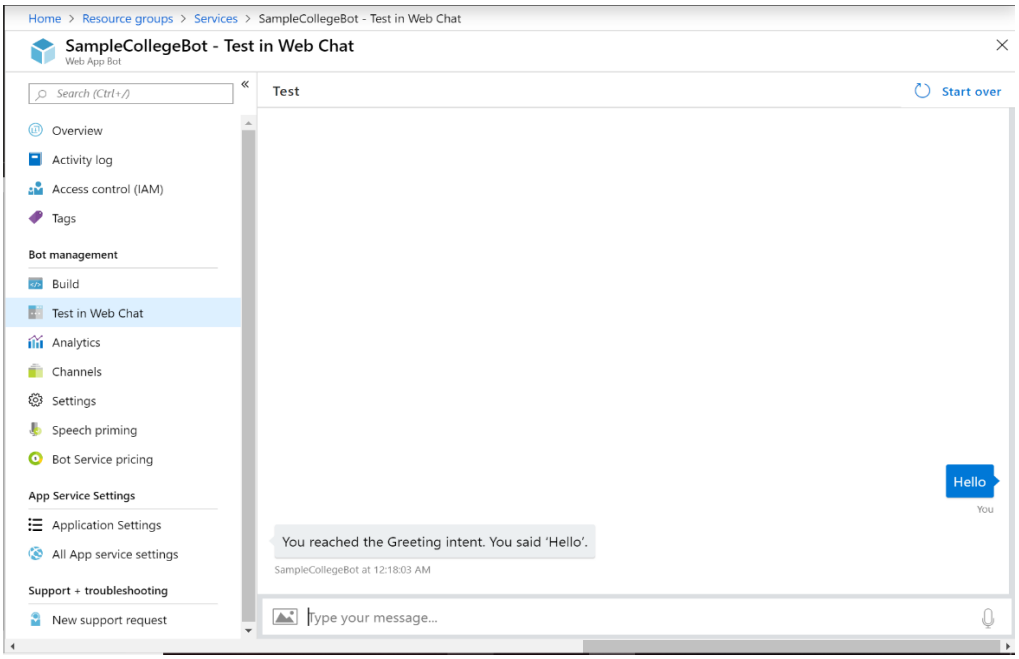


Figure 8: Web Chat Testing of the Created Bot

3.2 LUIS Setup

Go to https://www.luis.ai to setup the intents and entities to create a model and train the natural language processor.

3.2.1 LUIS App Creation

After visiting the website, sign in with the same Azure credentials, scroll down and click on the button “Create a LUIS app now” as shown in Figure 9. Please note that if a Web

App Bot is created with LUIS bot template, then a LUIS App will be created automatically with preconfigured intents and entities to work with.

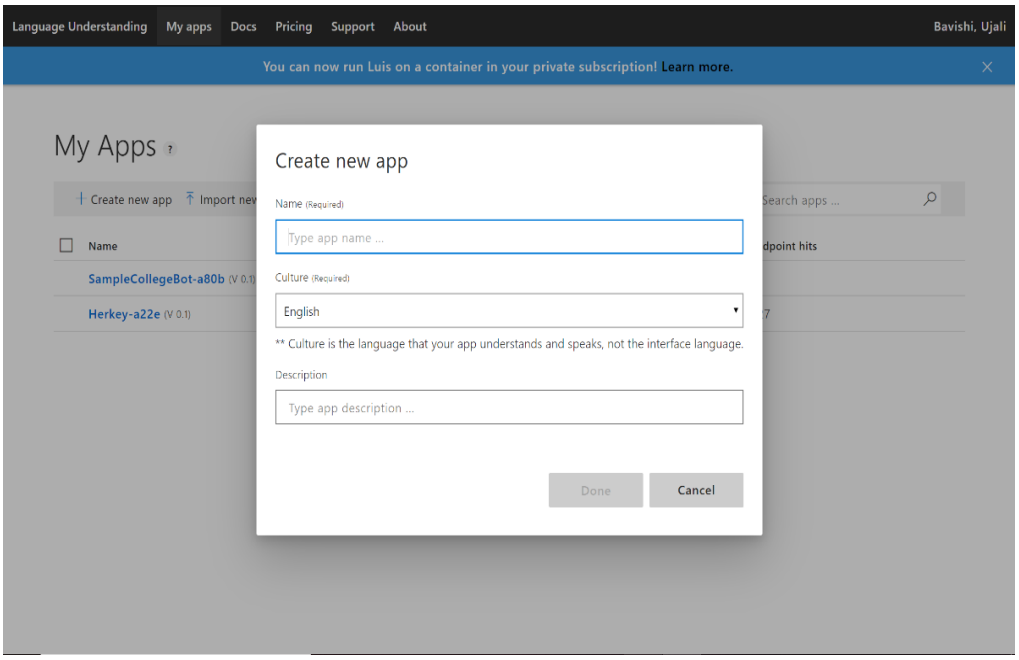


Figure 9: LUIS App Creation

3.2.2 Intents Creation

Go to created LUIS app and click on “Create new intent” and provide a specific intent name as shown in Figure 10. The next step would be to add training data to train the intent with all different types of utterances as shown in Figure 11.

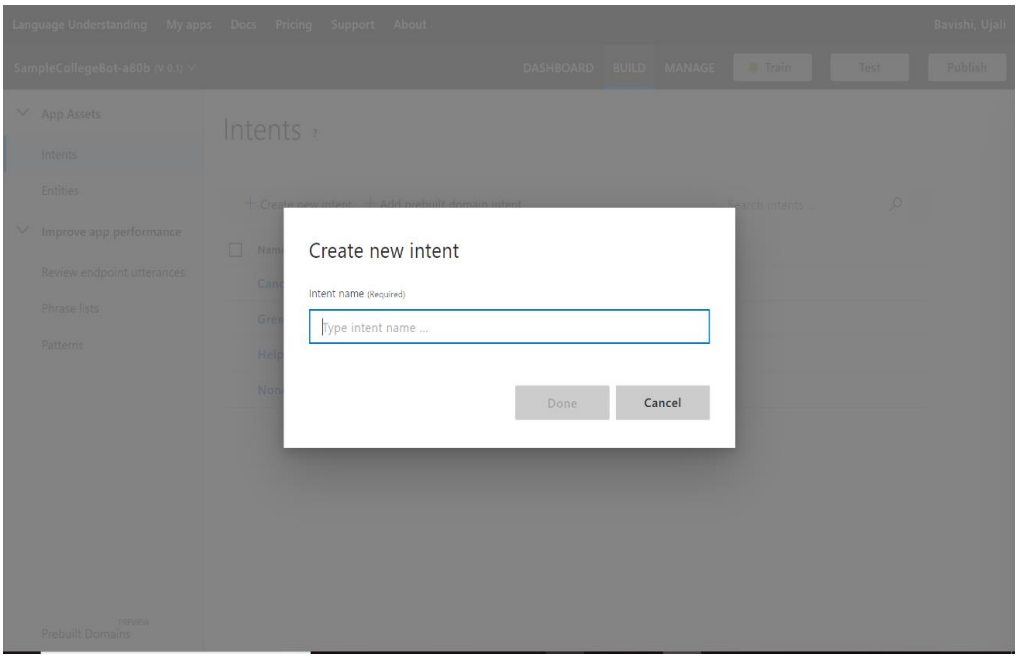


Figure 10: Intent Creation in LUIS

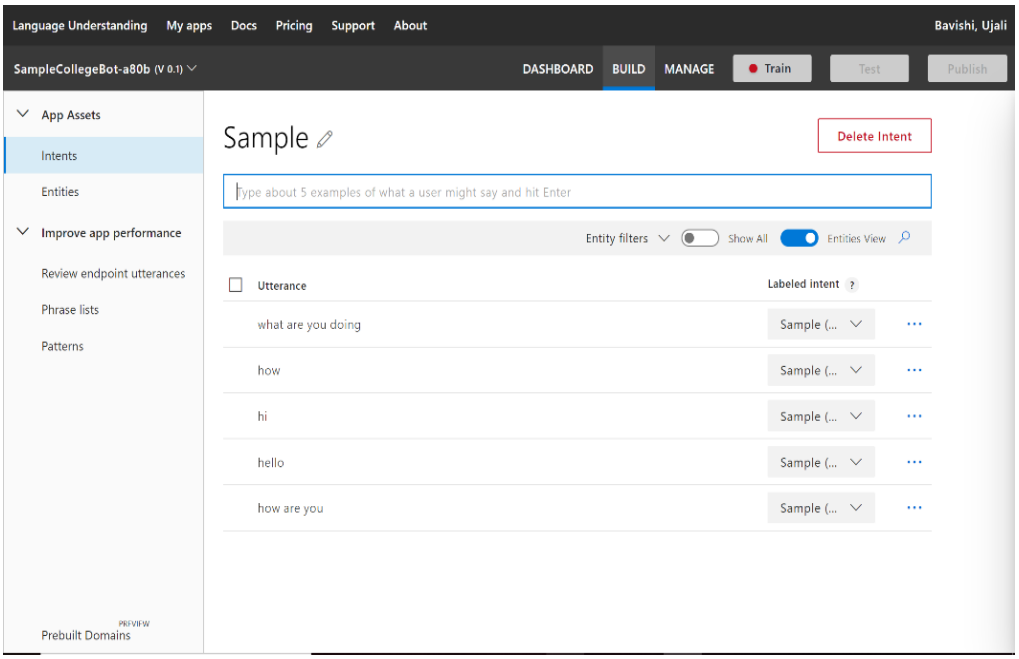


Figure 11: Adding Training Data to Created Intent

3.2.3 Entity Creation

In the left side pain, click on “entities” and then select “Create an Entity” and provide a name for the entity. In this project, a list type of entity is used since the professor and course values are stored in the entity sub list as shown in figures 12 and 13.

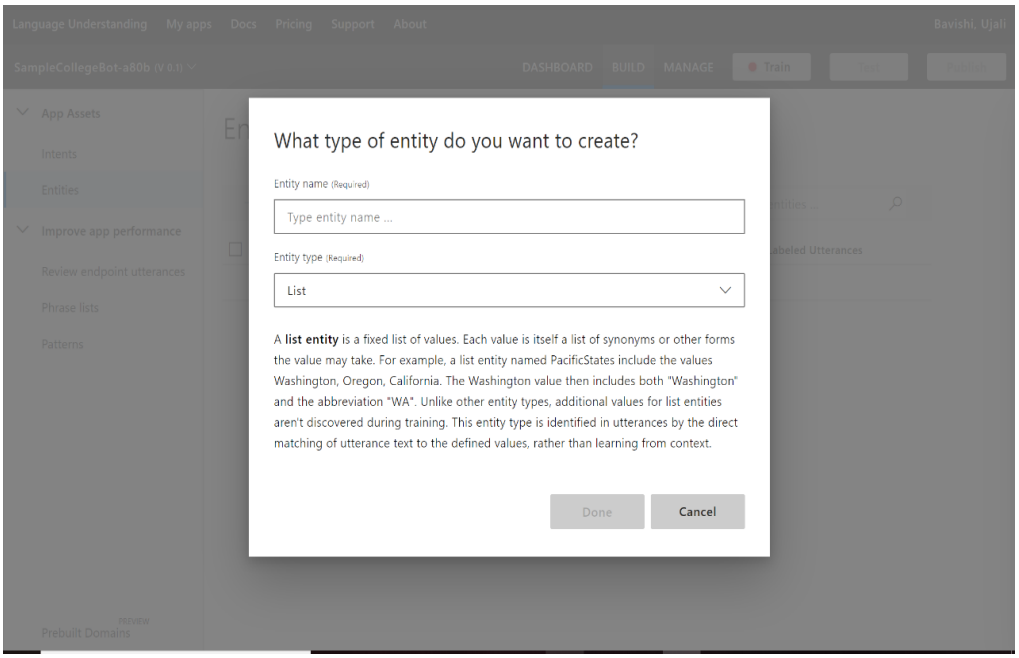


Figure 12: Entity Creation in LUIS - 1

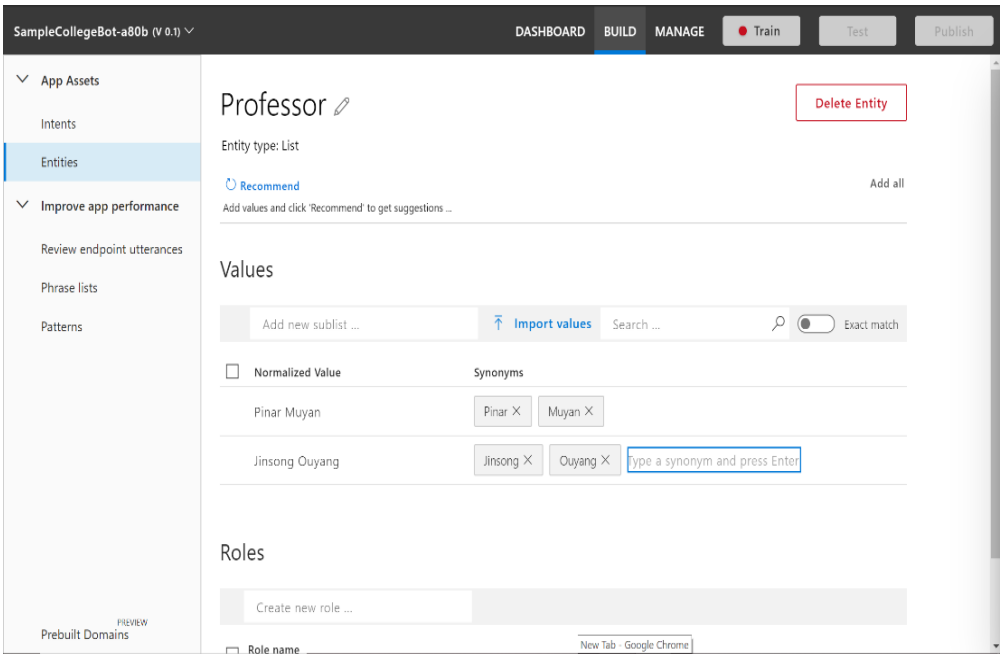


Figure 13: Entity Creation in LUIS - 2

3.2.4 Train and Publish the Model

After adding the intents and entities, the model is trained and published. After the model is published, REST API can be accessed from the service to access the LUIS endpoint to get the intent and entity from the user's query.

3.3 QnA Maker Setup

Go to https://www.qnamaker.ai to setup the questions and answers from the FAQs listed in any website and also to include some of the responses to be displayed depending on user's emotions based on some metadata.

3.3.1 QnA Maker App Creation

After visiting the website, sign in with the same Azure credentials, click on “Create a knowledge base” from the top navigation bar as shown in Figure 14. Then, follow the steps as shown in figures 15 and 16.

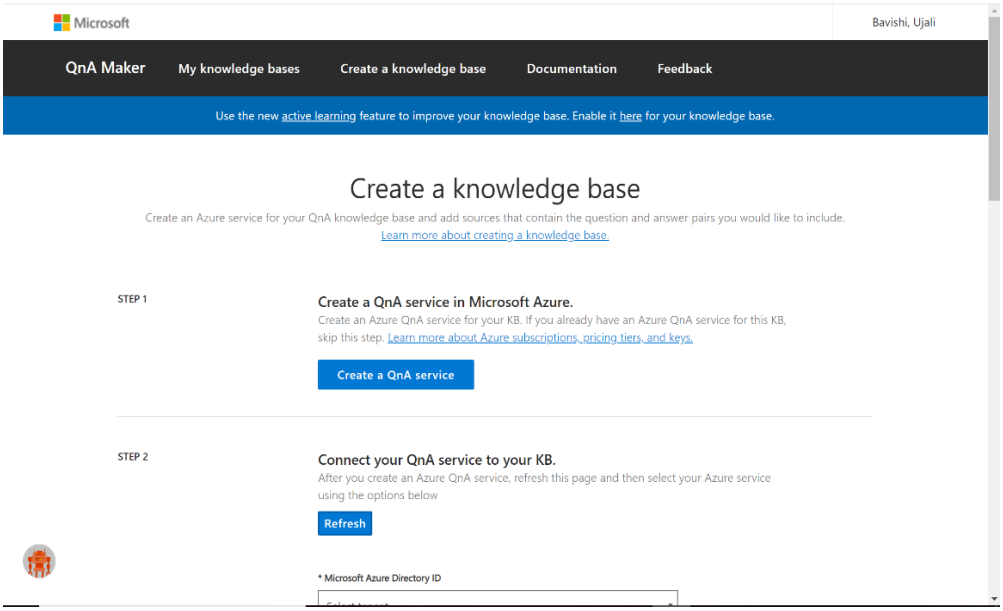


Figure 14: QnA Maker App Creation - 1

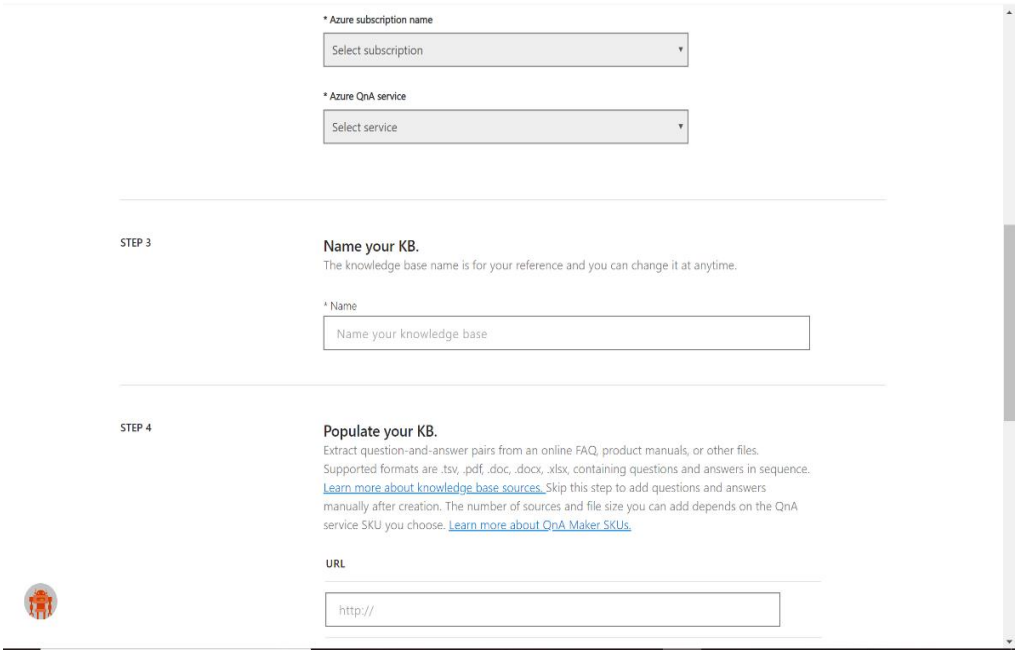


Figure 15: QnA Maker App Creation - 2

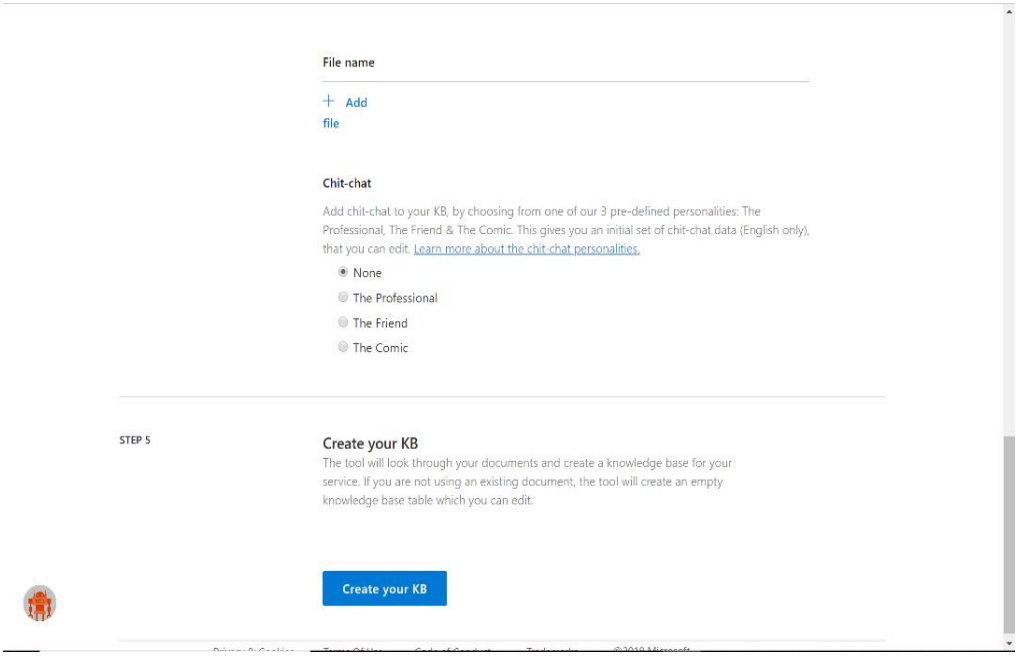


Figure 16: QnA Maker App Creation – 3

3.3.2 Add/Manage Knowledge base

Click on “Settings” from the top navigation and add FAQs URL from the CSUS website which are needed to be included in the knowledge base as shown in Figure 17

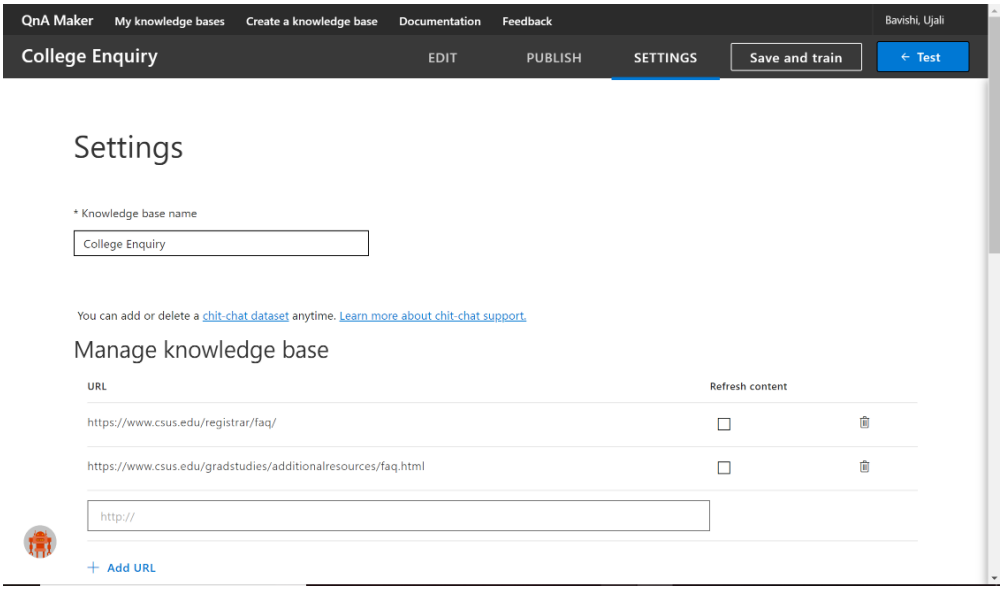


Figure 17: Knowledge Base Configuration Page

3.3.3 Train and Publish the Knowledge Base

After the knowledge bases are added, the next step is to train the app and publish to get a published endpoint to use the QnA Maker service.

3.4 Service Code Setup

I am using Azure DevOps to setup continuous deployment with a source control repository where the code will reside.

3.4.1 Create Project, Repository, and Branches

Sign in to Azure DevOps using the Azure account, create a new project and setup a repository based on the needs as shown in figures 18-20.

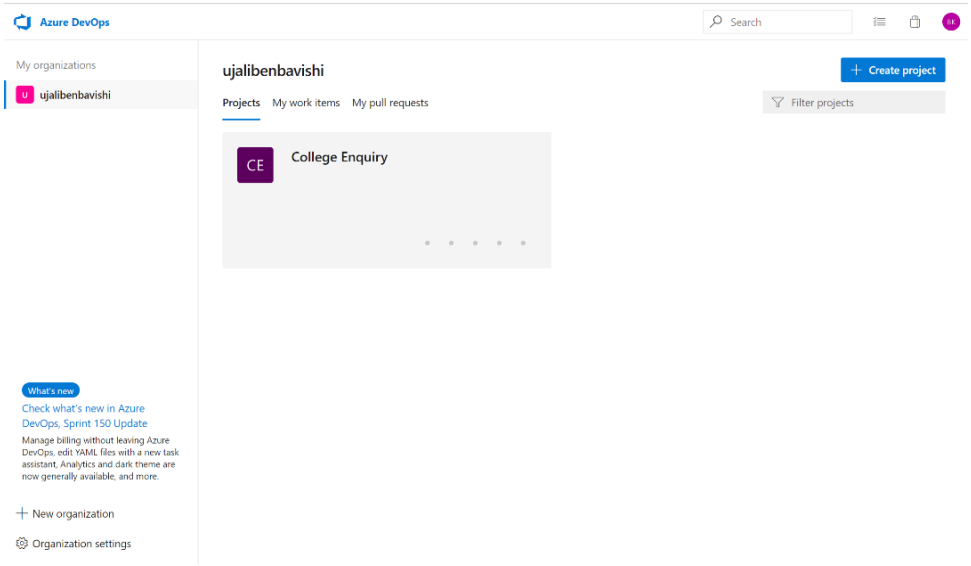


Figure 18: Azure DevOps Source Control Repo

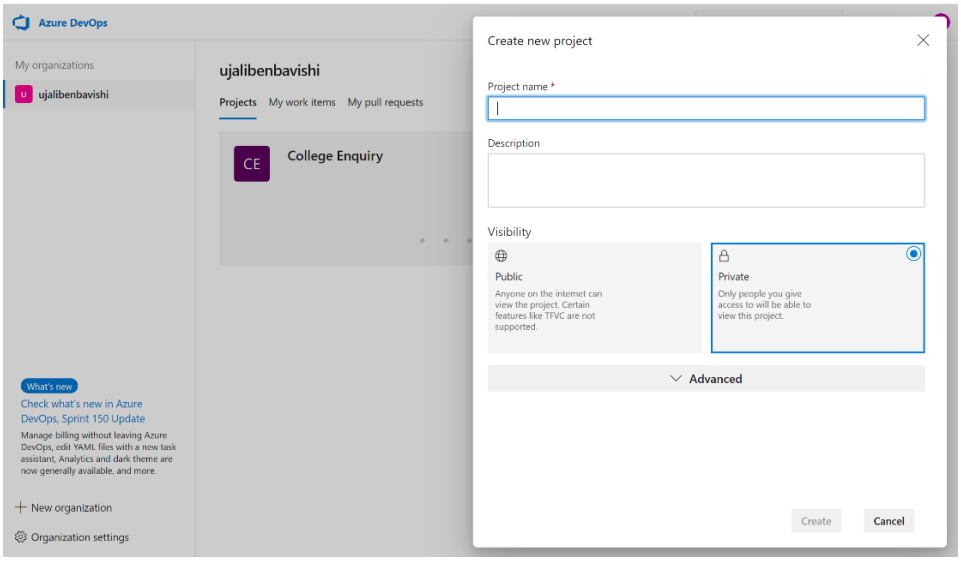


Figure 19: Project Creation on Azure DevOps

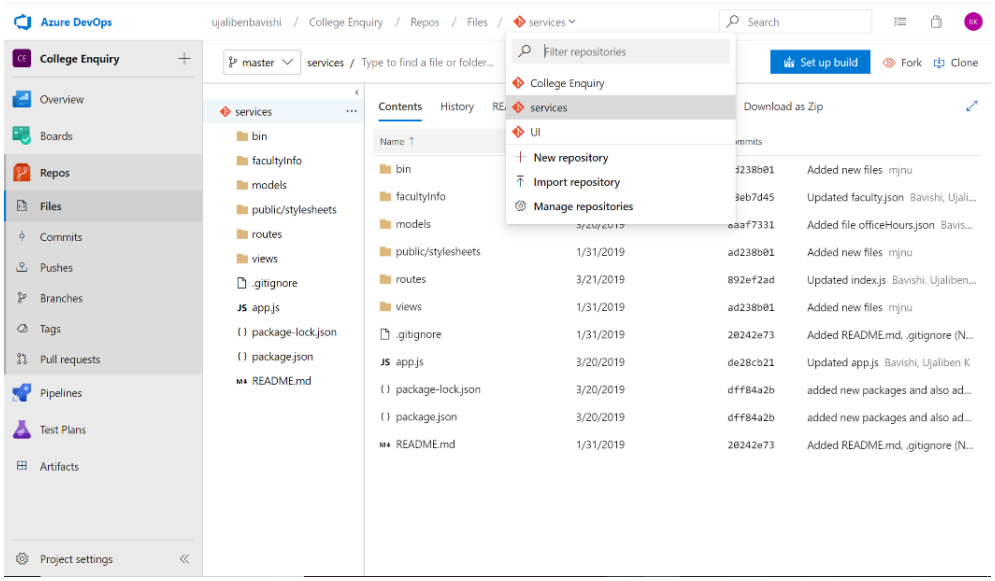


Figure 20: Repository and branch configuration in Azure DevOps Project

3.4.2 Sync Service Code with Azure DevOps

After the repository is setup, use Git clone functionality to clone the repo onto the local system and add all the files related to the service into the project. Finally, when a Git push is made from the same folder, all the files will be copied over to Azure DevOps.

3.5 Setup Continuous Deployment

As a last step, we need to configure the project created on Azure DevOps with the application created on Azure portal as shown in figures 21-24.

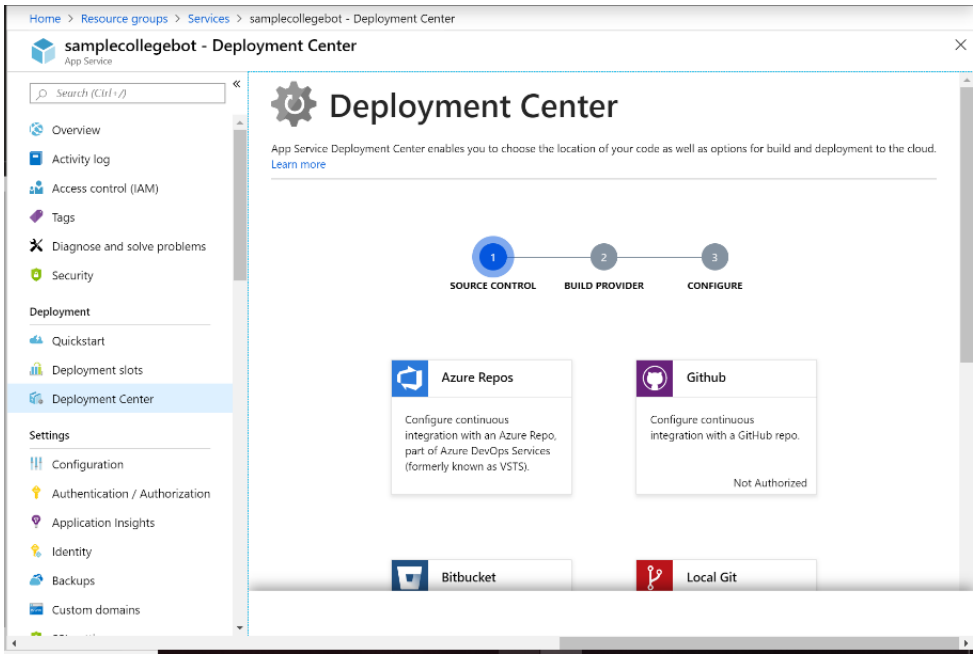


Figure 21: Continuous Deployment Configuration - 1

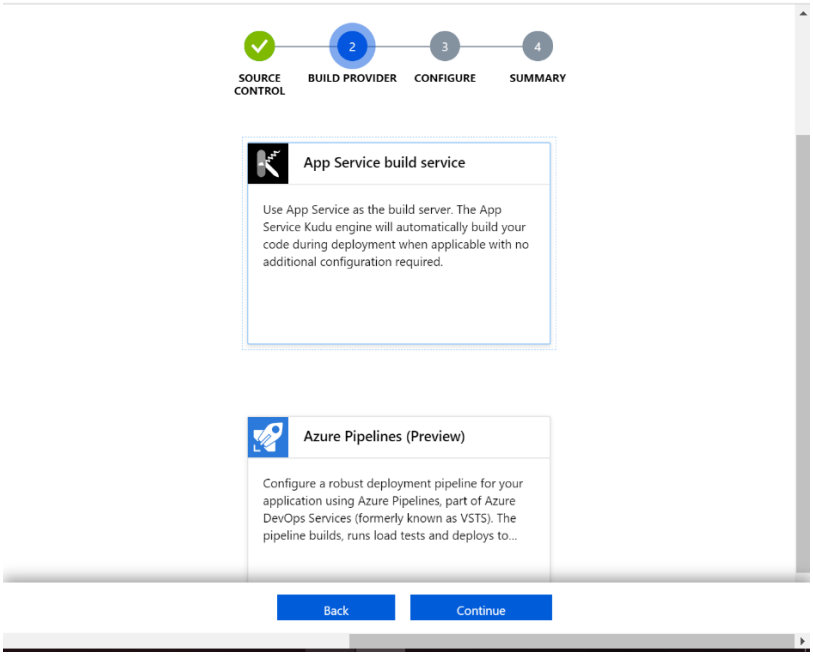


Figure 22: Continuous Deployment Configuration - 2

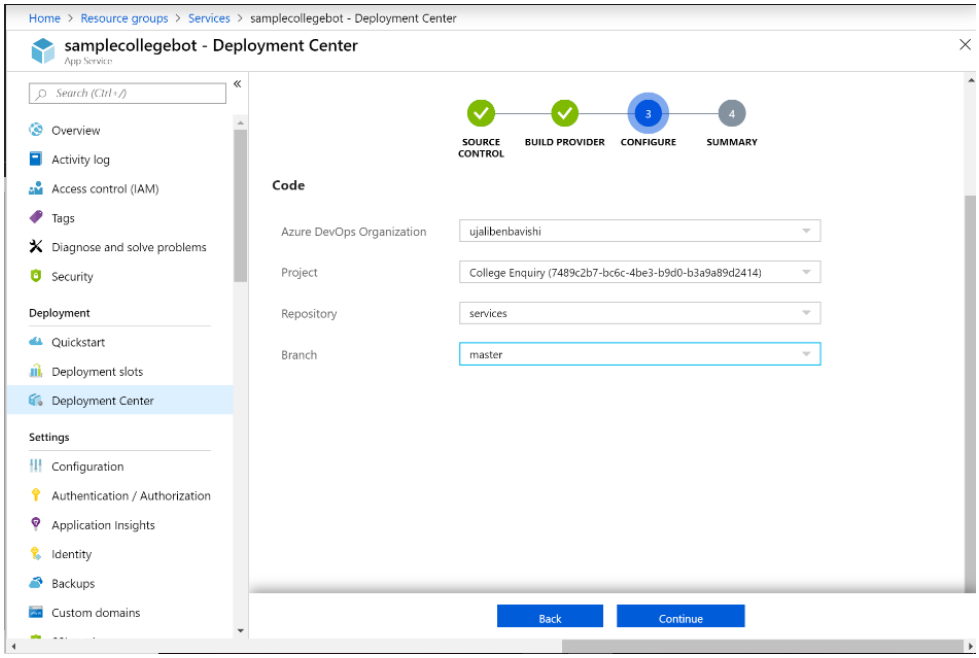


Figure 23: Continuous Deployment Configuration - 3

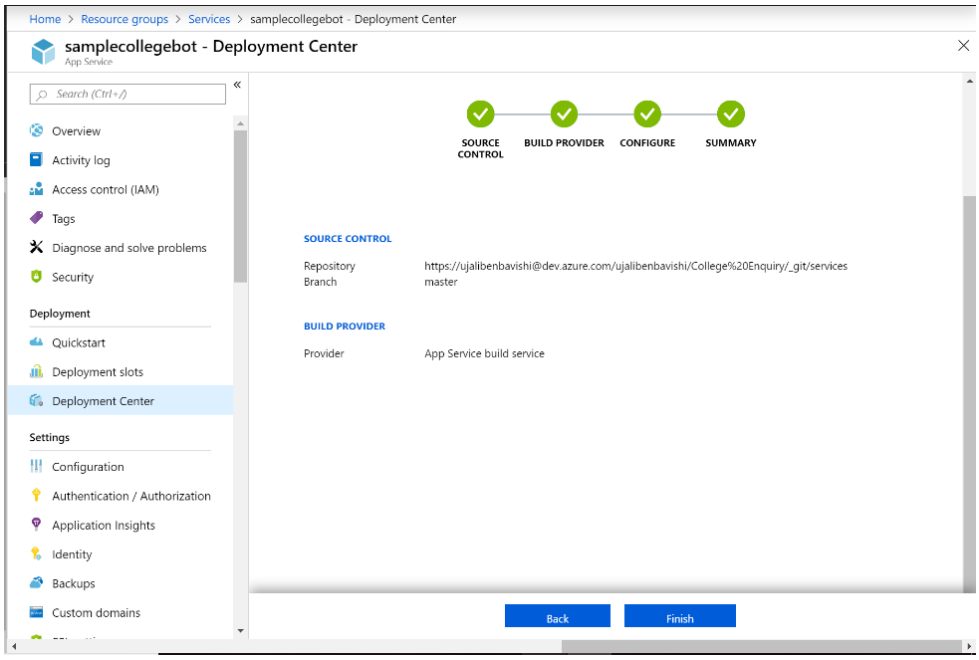


Figure 24: Continuous Deployment Configuration – 4

3.6 Service Code Snippets

In order to provide overview of the service code and some of the utility functions that are used by the bot, this section provides some screenshots. Figure 25 shows the overall structure of the code which utilizes NodeJS and ExpressJS frameworks.

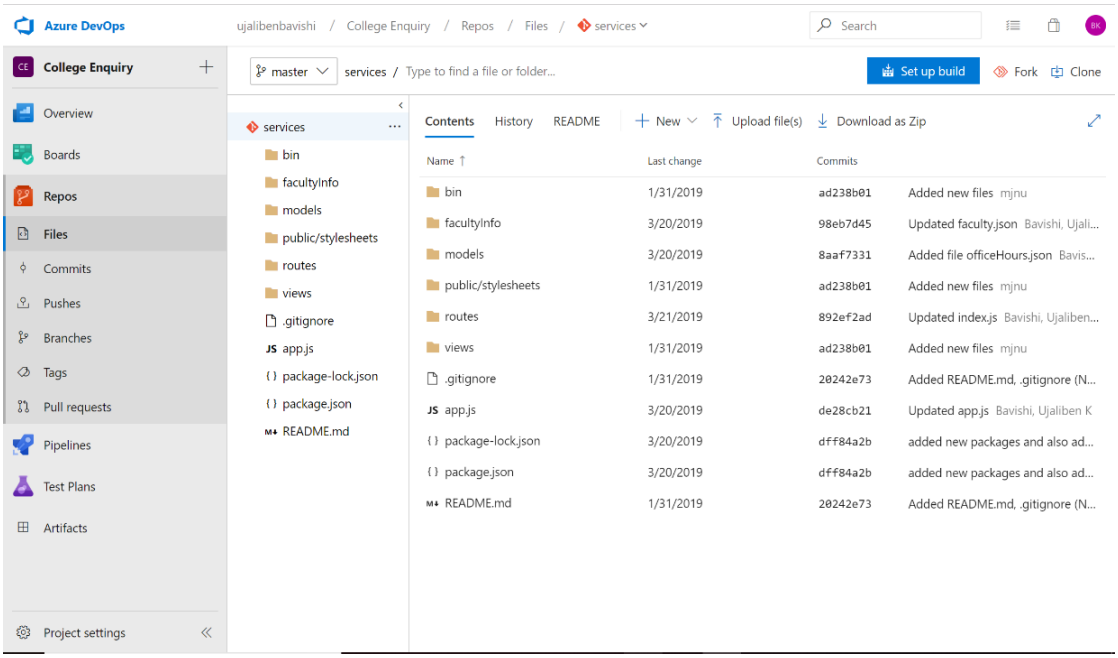


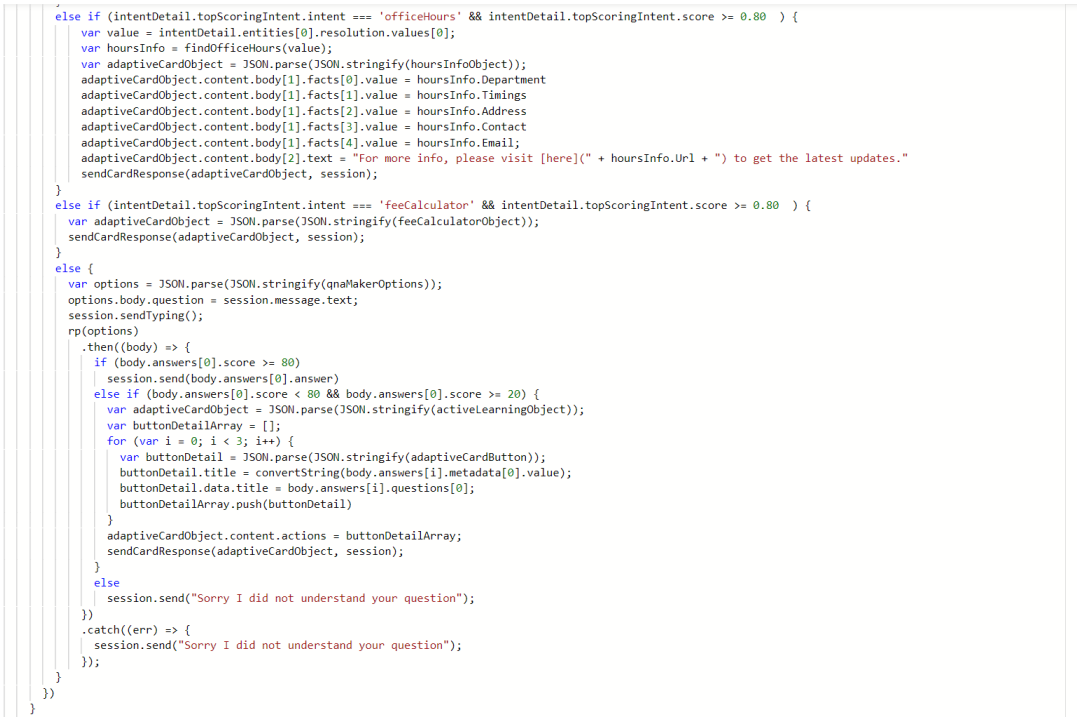
Figure 25: Code Structure Overview

Figures 26 and 27 show various functionalities including calling LUIS REST API, intercepting bot messages to save the chat history to the database and presenting the logic to show feedback card used in active learning.

```
bot.use({
  botbuilder: function (session, next) {
    if (session.message && session.message.value) {
      var studentType = session.message.value.studentType;
      var credits = session.message.value.credits;
      var registrationFee = 0;
      var totalFee = 0;
      if (studentType === "UnderGraduate") {
        if (parseInt(credits) > 6)
          totalFee = 3655 + parseInt(credits) * 396;
        else
          totalFee = 2449 + parseInt(credits) * 396;
      }
      else if (studentType === "Graduate") {
        if (parseInt(credits) > 6)
          totalFee = 4372 + parseInt(credits) * 396;
        else
          totalFee = 2866 + parseInt(credits) * 396;
      }
      var resp = "Based on your input, the total fees would be approximately $" + totalFee + ".00";
      session.send(resp);
    }
    else if (session.message && session.message.text !== "") {
      getIntentOptions.url = LuisModelUrl.concat(session.message.text);
      rp(getIntentOptions).then((result) => {
        var intentDetail = JSON.parse(result);
        console.log(result);
        var timeStamp = moment(new Date()).tz("America/Los_Angeles").format("MM/DD/YYYY hh:mm:ss a");
        var updates = {};
        if (intentDetail.sentimentAnalysis) {
          if (Math.round(intentDetail.sentimentAnalysis.score) === 1) {
            updates = { $setOnInsert: { startTimeStamp: timeStamp }, $set: { endTimeStamp: timeStamp }, $inc: { positive: 1 }, $push: { conversations: { $each: [{ "use
            }
            else {
              updates = { $setOnInsert: { startTimeStamp: timeStamp }, $set: { endTimeStamp: timeStamp }, $inc: { negative: 1 }, $push: { conversations: { $each: [{ "use
            }
          }
          else {
            updates = { $setOnInsert: { startTimeStamp: timeStamp }, $set: { endTimeStamp: timeStamp }, $push: { conversations: { $each: [{ "user": session.message.text,
          }
          insertIntoDb(session.message.address.conversation.id, updates)

          if (intentDetail.topScoringIntent.intent === 'courseInfo' && intentDetail.topScoringIntent.score >= 0.80  ) {
```

Figure 26: LUIS and Active Learning Logic - 1



```
    else if (intentDetail.topScoringIntent.intent === 'officeHours' && intentDetail.topScoringIntent.score >= 0.80  ) {
      var value = intentDetail.entities[0].resolution.values[0];
      var hoursInfo = findOfficeHours(value);
      var adaptiveCardObject = JSON.parse(JSON.stringify(hoursInfoObject));
      adaptiveCardObject.content.body[1].facts[0].value = hoursInfo.Department
      adaptiveCardObject.content.body[1].facts[1].value = hoursInfo.Timings
      adaptiveCardObject.content.body[1].facts[2].value = hoursInfo.Address
      adaptiveCardObject.content.body[1].facts[3].value = hoursInfo.Contact
      adaptiveCardObject.content.body[1].facts[4].value = hoursInfo.Email;
      adaptiveCardObject.content.body[2].text = "For more info, please visit [here](" + hoursInfo.Url + ") to get the latest updates."
      sendCardResponse(adaptiveCardObject, session);
    }
    else if (intentDetail.topScoringIntent.intent === 'feeCalculator' && intentDetail.topScoringIntent.score >= 0.80  ) {
      var adaptiveCardObject = JSON.parse(JSON.stringify(feeCalculatorObject));
      sendCardResponse(adaptiveCardObject, session);
    }
    else {
      var options = JSON.parse(JSON.stringify(qnaMakerOptions));
      options.body.question = session.message.text;
      session.sendTyping();
      rp(options)
        .then((body) => {
          if (body.answers[0].score >= 80)
            session.send(body.answers[0].answer)
          else if (body.answers[0].score < 80 && body.answers[0].score >= 20) {
            var adaptiveCardObject = JSON.parse(JSON.stringify(activeLearningObject));
            var buttonDetailArray = [];
            for (var i = 0; i < 3; i++) {
              var buttonDetail = JSON.parse(JSON.stringify(adaptiveCardButton));
              buttonDetail.title = convertString(body.answers[i].metadata[0].value);
              buttonDetail.data.title = body.answers[i].questions[0];
              buttonDetailArray.push(buttonDetail)
            }
            adaptiveCardObject.content.actions = buttonDetailArray;
            sendCardResponse(adaptiveCardObject, session);
          }
          else
            session.send("Sorry I did not understand your question");
        })
        .catch((err) => {
          session.send("Sorry I did not understand your question");
        });
    }
  })
}
```

Figure 27: LUIS and Active Learning Logic - 2

Figure 28 shows a sample file which contains basic information related to the course like course id, name, information, prerequisites, professor's name and URL of the course.

```
[
    {
        "Course Id": "CSC201",
        "Course Name": "Programming Language Principles",
        "Course Information": "Notations for the specification of programming language syntax and semantics; attribute, translational, operational, axiomatic, algebra
        "Prerequisites of Course": "Fully classified graduate status in Computer Science or Software Engineering",
        "Units": "3",
        "Professor for the course": "Dr. Cui Zhang",
        "Course Url": "https://www.ecs.csus.edu/wcm/csc/pdfs/advising/csc201.pdf"
    },
    {
        "Course Id": "CSC204",
        "Course Name": "Data Models for Database Management Systems",
        "Course Information": "Database management system (DBMS) architecture. Database file organizations and access methods. The relational model and relational alg
        "Prerequisites of Course": "Fully classified graduate status in Computer Science or Software Engineering",
        "Units": "3",
        "Professor for the course": "Bill Mitchell",
        "Course Url": "https://www.ecs.csus.edu/wcm/csc/pdfs/advising/csc204.pdf"
    },
    {
        "Course Id": "CSC205",
        "Course Name": "Computer Systems Structure",
        "Course Information": "Overview of computer systems structure, covering hierarchical structure from software and hardware points of view. Concepts of relocati
        "Prerequisites of Course": "Fully classified graduate status in Computer Science or Software Engineering",
        "Units": "3",
        "Professor for the course": "Dr. Weide Chang",
        "Course Url": "https://www.ecs.csus.edu/wcm/csc/pdfs/advising/csc205.pdf"
    },
    {
        "Course Id": "CSC206",
        "Course Name": "Algorithms and Paradigms",
        "Course Information": "Design and analysis of algorithms. Classical design paradigms including greedy, divide-and-conquer, dynamic programming, and backtracki
        "Prerequisites of Course": "Fully classified graduate status in Computer Science or Software Engineering",
        "Units": "3",
        "Professor for the course": "Dr. Chung-E Wang",
        "Course Url": "https://www.ecs.csus.edu/wcm/csc/pdfs/advising/csc206.pdf"
    },
```


Figure 28: Sample Course Details

Figure 29 shows a sample file which contains basic information related to the professors like name, office hours, location, and email.

```
[
{
    "Name":"Dr. Nik Faroughi, Department Chair (CSC)",
    "Location":"RVR 3018G",
    "Email":"faroughi@ecs.csus.edu",
    "Phone":"916-278-7628 ",
    "Office Hours": "M W F 11:00am-12:00pm",
    "Webpage":"https://athena.ecs.csus.edu/~faroughi/"
},
{
    "Name":"Dr. Scott Gordon, Assoc. Chair (CSC)",
    "Location":"RVR 3018F",
    "Email":"gordonvs@ecs.csus.edu",
    "Phone":"Use Email",
    "Office Hours": "W 2:00pm - 4:00pm",
    "Webpage":"https://athena.ecs.csus.edu/~gordonvs/"
},
{
    "Name":"Dr. Jinsong Ouyang, Graduate Coordinator (CSC)",
    "Location":"RVR 3018I",
    "Email":"jouyang@csus.edu",
    "Phone":"916-278-5769",
    "Office Hours": "Tu R 1:30pm - 3:00pm",
    "Webpage":"https://www.csus.edu/faculty/o/jouyang/"
},
{
    "Name":"Dr. Pinar Muyan-Ozcelik",
    "Location":"RVR 5008",
    "Email":"pmuyan@ecs.csus.edu",
    "Phone":"916-278-6713",
    "Office Hours": "Tu 3:00pm - 3:50pm, R 1:40pm - 3:50pm",
    "Webpage":"https://athena.ecs.csus.edu/~pmuyan/"
},
{
    "Name":"Dr. Haiquan Chen",
    "Location":"RVR 5018",
    "Email":"chenh@ecs.csus.edu",
    "Phone":"916-278-6087",
    "Office Hours": "M W 3:00pm - 4:00pm",
    "Webpage":"https://athena.ecs.csus.edu/~chenh/"
},
```


Figure 29: Sample Faculty Details

4 EXECUTION RESULTS

In this section, execution results of College Enquiry Chatbots are provided by listing several functionalities included in this project and explaining these functionalities with the help of the screenshots.

4.1 How to chat

To start chat with the chatbot, a student will have to go to the university website www.csus.edu and click on the question mark icon as shown in Figure 30.

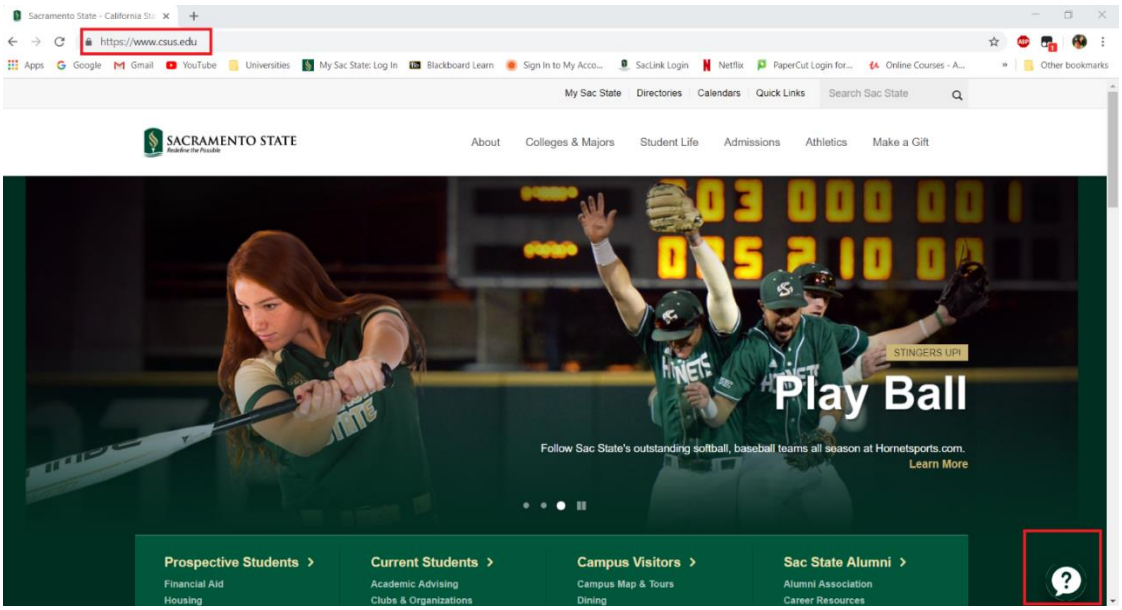


Figure 30: How to Chat

To have the chatbot on a live website, access to the codebase of a website is required. However, for now a software called Tampermonkey is used to inject the Java script into the browser.

4.2 Sentiment Analysis

Figures 31 and 32 show how a bot is answering the irrelevant questions of a user for which normal bots are not showing any kind of empathy. To make the bot to be able to respond these kinds of responses, the bot is trained with the chitchat integration which is provided by QnA Maker in which there are lots of questions and answers pairs are available.

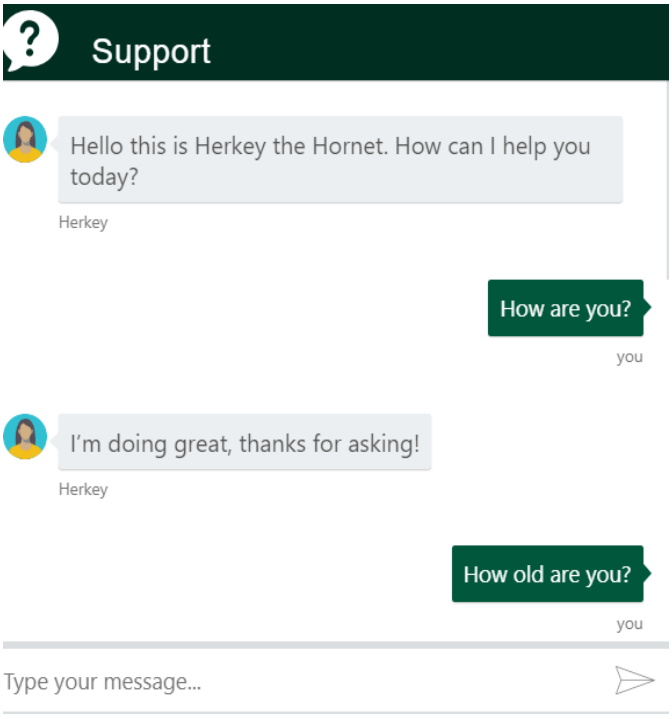


Figure 31: Friendly Conversations - 1

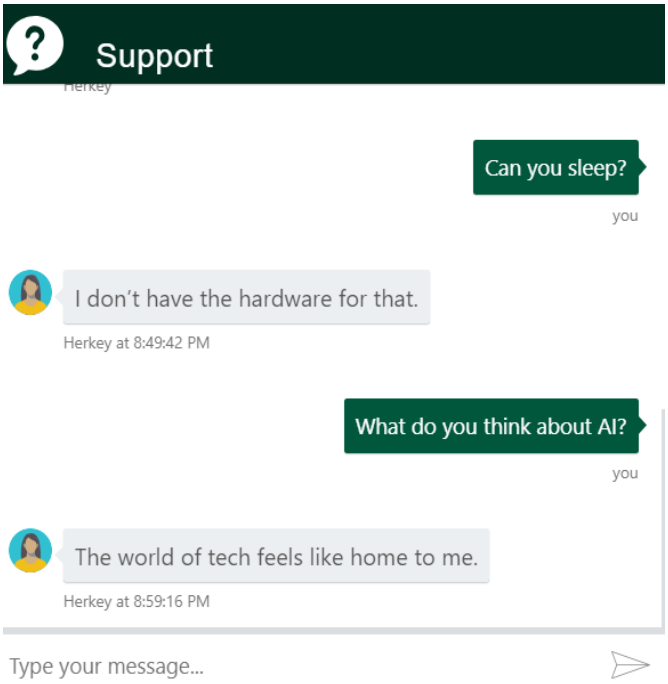
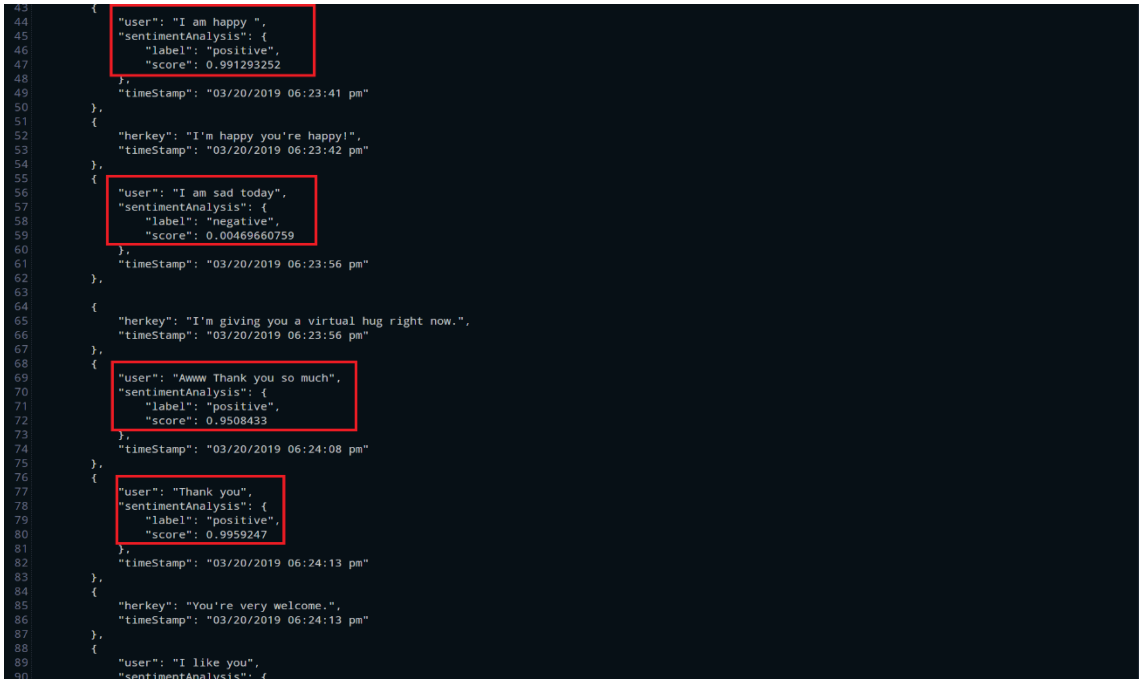


Figure 32: Friendly Conversations – 2

Figures 33 and 34 shows how conversations are being stored in the database and how LUIS performs sentiment analysis on user queries based on which it gives the response back to the user. A bot interceptor is used in the service where conversations are tracked and user's query is passed onto the LUIS endpoint which has an in-built sentiment analyzer to track whether the query is positive, negative or neutral. These results are then stored into the database for any further analysis.

```
{
    "_id": {
        "$oid": "5c92e666fca6e5e29a2628fd"
    },
    "sessionId": "GUOv9o6K91K6B03TidJJtH-g",
    "conversations": [
        {
            "herkey": "Hello this is Herkey the Hornet. How can I help you today?",
            "timeStamp": "03/20/2019 06:18:30 pm"
        },
        {
            "user": "What can you do?",
            "sentimentAnalysis": {
                "label": "positive",
                "score": 0.843127966
            },
            "timeStamp": "03/20/2019 06:20:17 pm"
        },
        {
            "herkey": "I'm here to chat and to try to help out.",
            "timeStamp": "03/20/2019 06:20:18 pm"
        },
        {
            "user": "professor info",
            "sentimentAnalysis": {
                "label": "neutral",
                "score": 0.5
            },
            "timeStamp": "03/20/2019 06:22:47 pm"
        },
        {
            "herkey": "I can help you with that. Please provide the name of the Professor",
            "timeStamp": "03/20/2019 06:22:47 pm"
        },
        {
            "user": "Pinar Muyan",
            "sentimentAnalysis": {
                "label": "neutral",
                "score": 0.5
            },
            "timeStamp": "03/20/2019 06:23:00 pm"
        },
```


Figure 33: Sentiment Analysis on User Queries - 1



```
        {
            "user": "I am happy ",
            "sentimentAnalysis": {
                "label": "positive",
                "score": 0.991293252
            },
            "timeStamp": "03/20/2019 06:23:41 pm"
        },
        {
            "herkey": "I'm happy you're happy!",
            "timeStamp": "03/20/2019 06:23:42 pm"
        },
        {
            "user": "I am sad today",
            "sentimentAnalysis": {
                "label": "negative",
                "score": 0.00469660759
            },
            "timeStamp": "03/20/2019 06:23:56 pm"
        },

        {
            "herkey": "I'm giving you a virtual hug right now.",
            "timeStamp": "03/20/2019 06:23:56 pm"
        },
        {
            "user": "Awww Thank you so much",
            "sentimentAnalysis": {
                "label": "positive",
                "score": 0.9508433
            },
            "timeStamp": "03/20/2019 06:24:08 pm"
        },
        {
            "user": "Thank you",
            "sentimentAnalysis": {
                "label": "positive",
                "score": 0.9959247
            },
            "timeStamp": "03/20/2019 06:24:13 pm"
        },
        {
            "herkey": "You're very welcome.",
            "timeStamp": "03/20/2019 06:24:13 pm"
        },
        {
            "user": "I like you",
```


Figure 34: Sentiment Analysis on User Queries – 2

4.3 Active Learning

Whenever a user asks anything which is outside of the script, the bot asks questions back to the user to understand what exactly user wants to ask and gives the responses accordingly. Figures 35 and 36 shows how a student has asked “on campus” randomly and then bot has asked question in return with three different options to understand what the user means. Once user has selected one of the options it did frame a question based on that option and gave the response to the user.

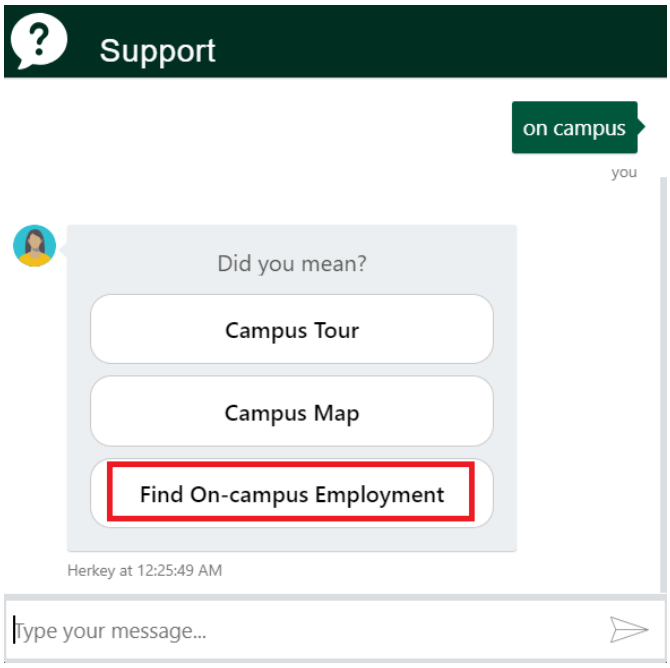


Figure 35: Active Learning - 1

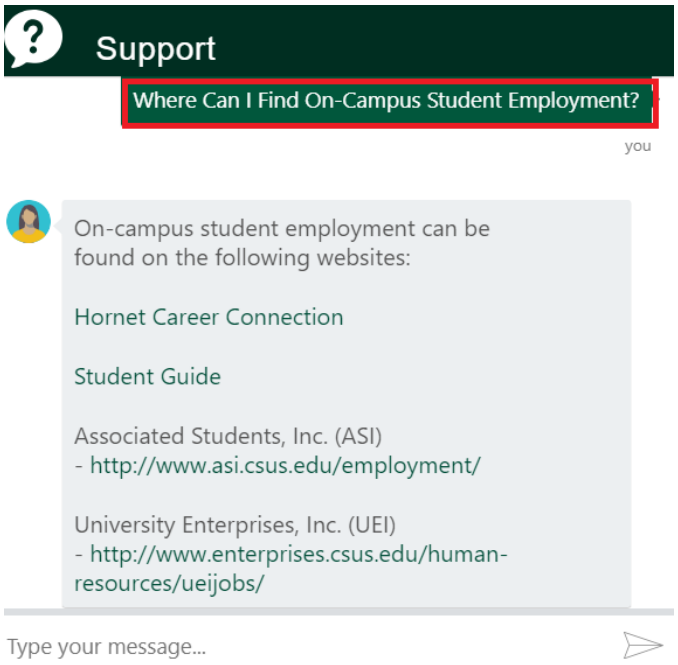


Figure 36: Active Learning – 2

4.4 Frequently Asked Questions

The system inserts FAQs related to academics, registration, records, evaluations/graduation, applications, current students, and campus information. Figures 37 and 38 show two different highlighted questions and answers from two different FAQ URLs which are added to QnA Maker.

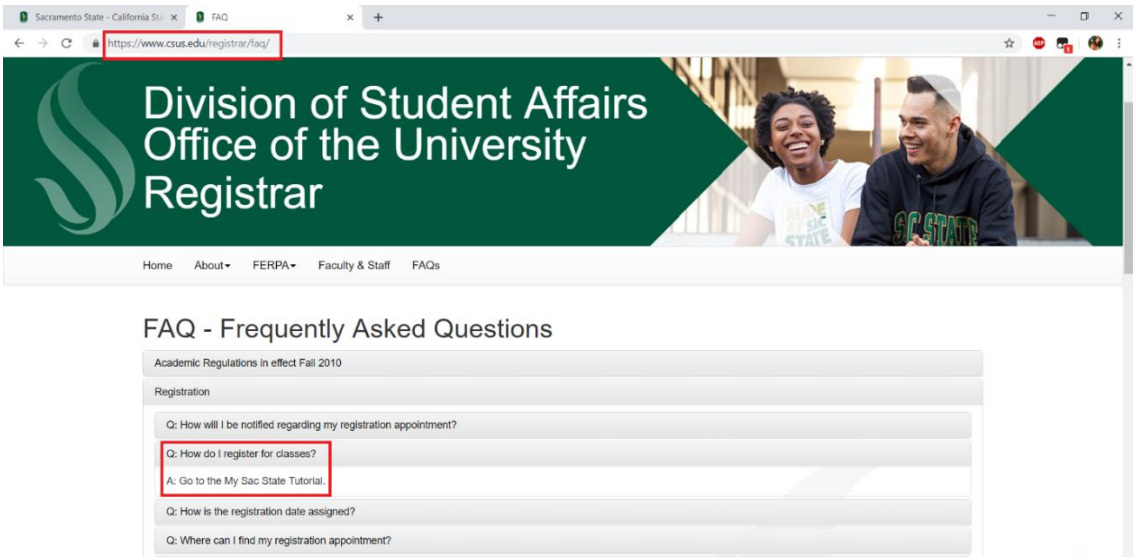


Figure 37: FAQs from Registrar

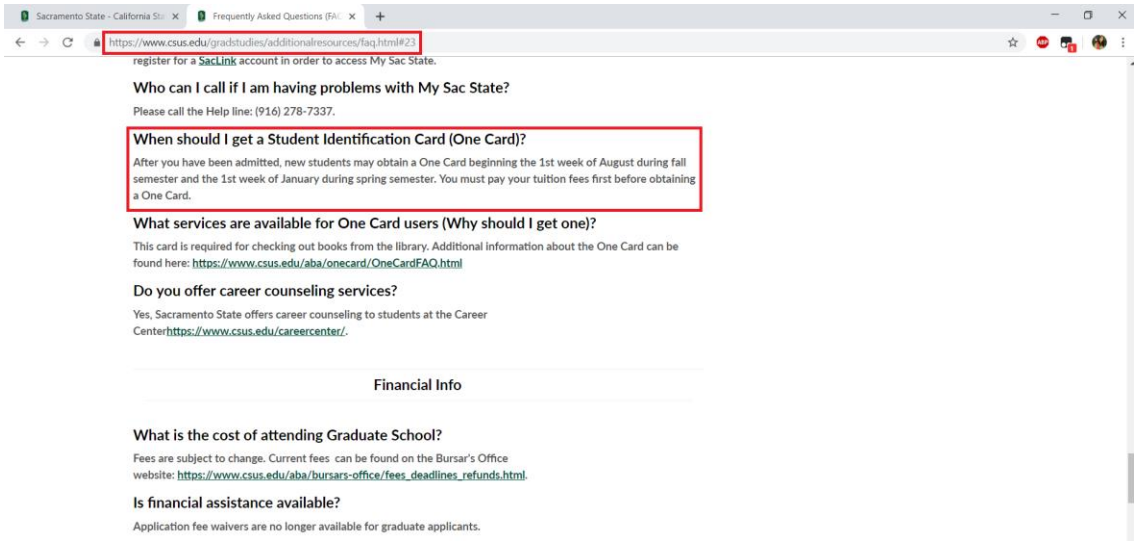


Figure 38: FAQ from Graduate Studies

Figure 39 shows how questions from FAQs are asked and responded in an exact same way. In this case, notice that a student does not have to ask an exact same question included in FAQs. This is the main advantage of an AI-powered chatbot.

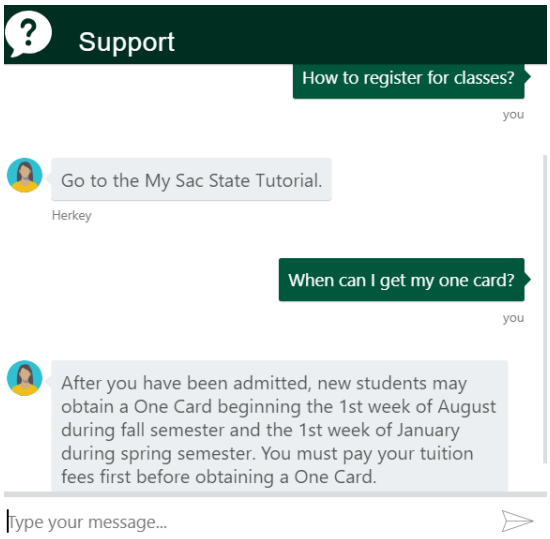


Figure 39: FAQs Asked to Chatbot

4.5 Course and Faculty Details

Figure 40 shows how chatbot is giving the information about course when asked. It shows name, overview, prerequisites, number of units, and professor of the subject. It also gives a course link to the student for more information.

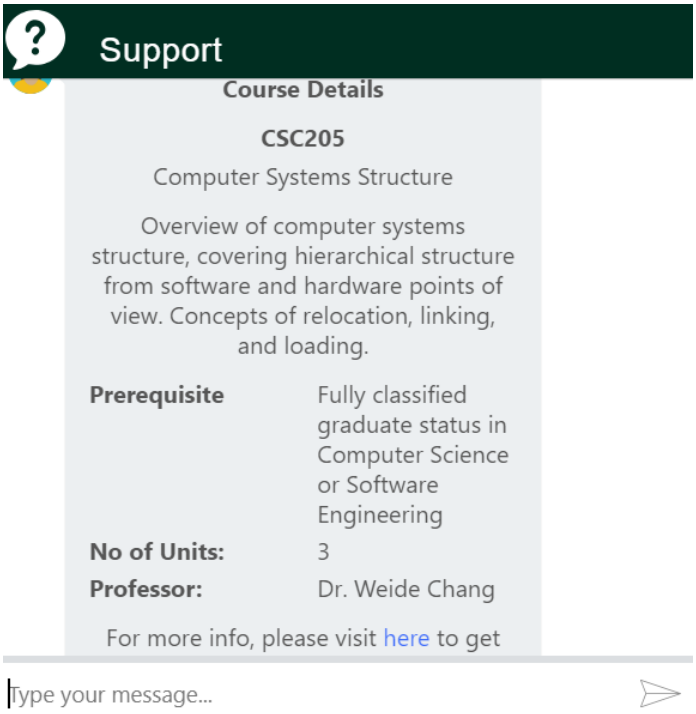


Figure 40: Course Information

Figure 41 shows the chatbot's response about office hours of a faculty. It gives all the necessary details about the faculty such as name, location of the office, email, phone, and office hours of the faculty. It also gives a link to the faculty's website for more information.

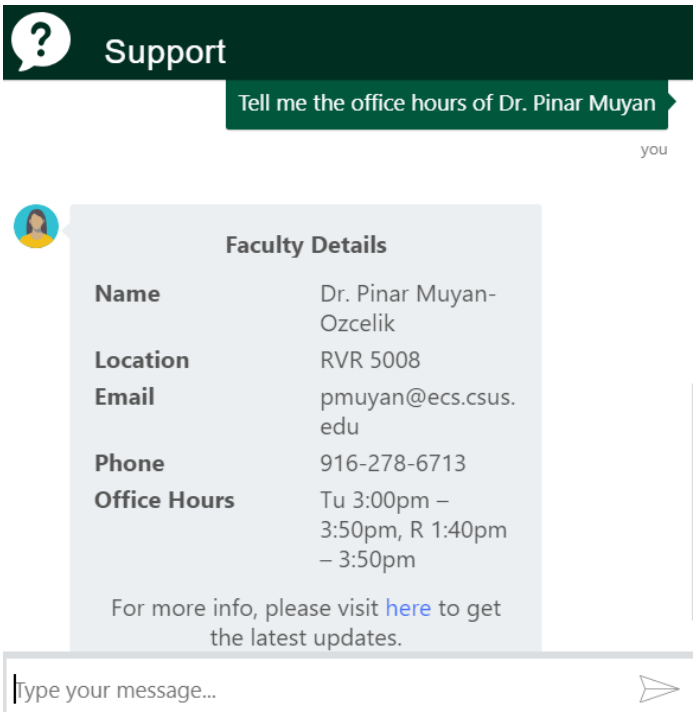


Figure 41: Faculty Information

4.6 Academic Calendar

Figure 42 shows how the bot gives a link to an academic calendar if a student is asking about all the holidays of the semester.

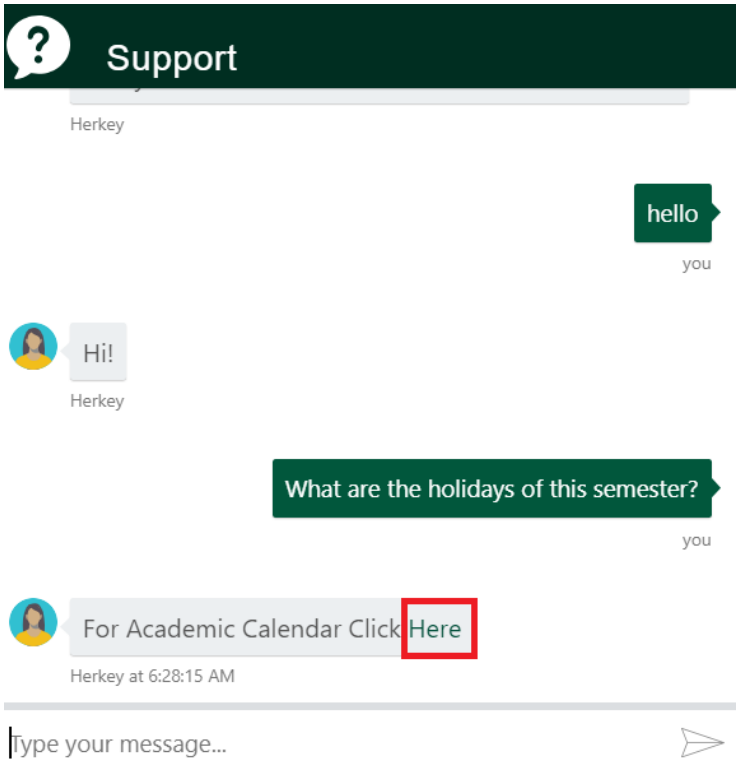


Figure 42: Academic Calendar

4.7 Fee Calculator

Figures 43 and 44 shows about the chatbot providing facility of calculating the fees by just selecting undergraduate or graduate and inserting number of units.

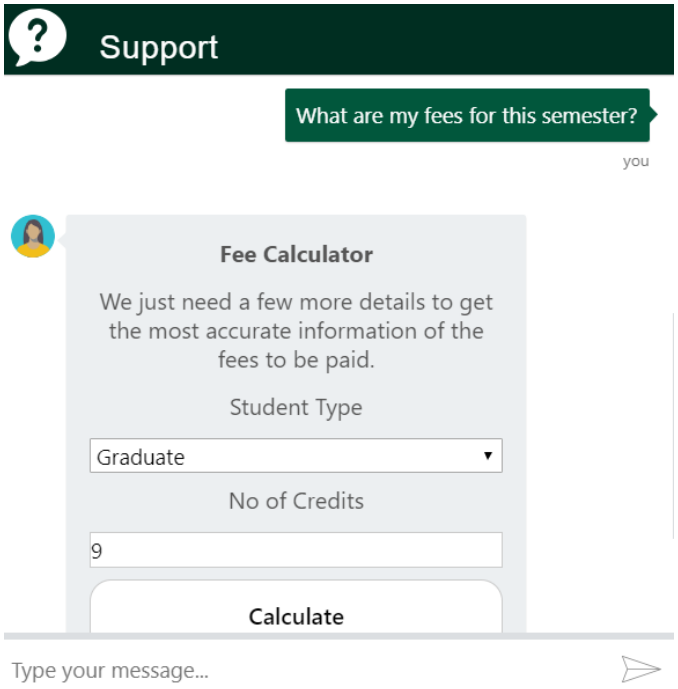


Figure 43: Fee Calculator – 1

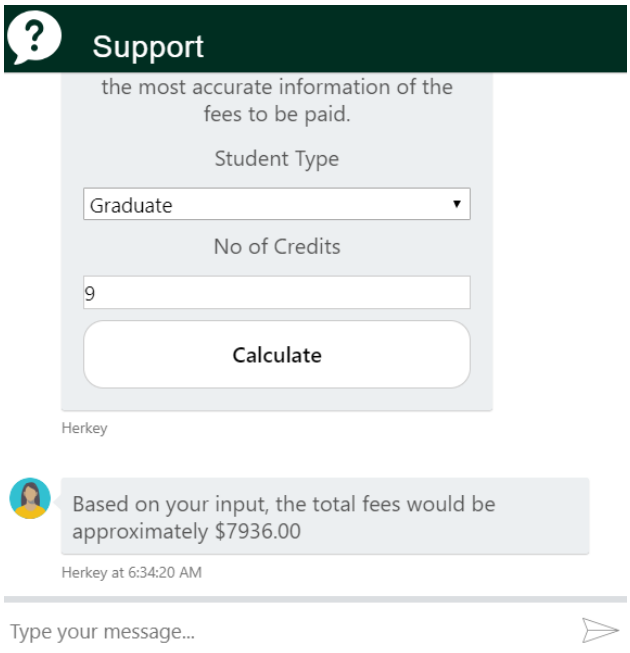


Figure 44: Fee Calculator - 2

4.8 Other Departmental Information

Figures 45 and 46 shows how the chatbot gives information about IPGE and Union office hours, address, contact info, and email. It also gives a link to the respected department for more information.

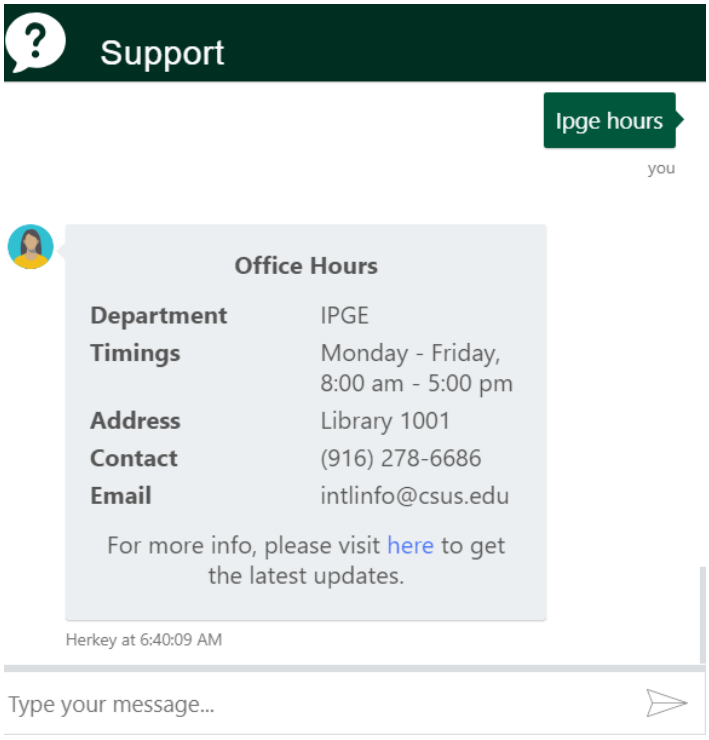


Figure 45: IPGE Hours Information

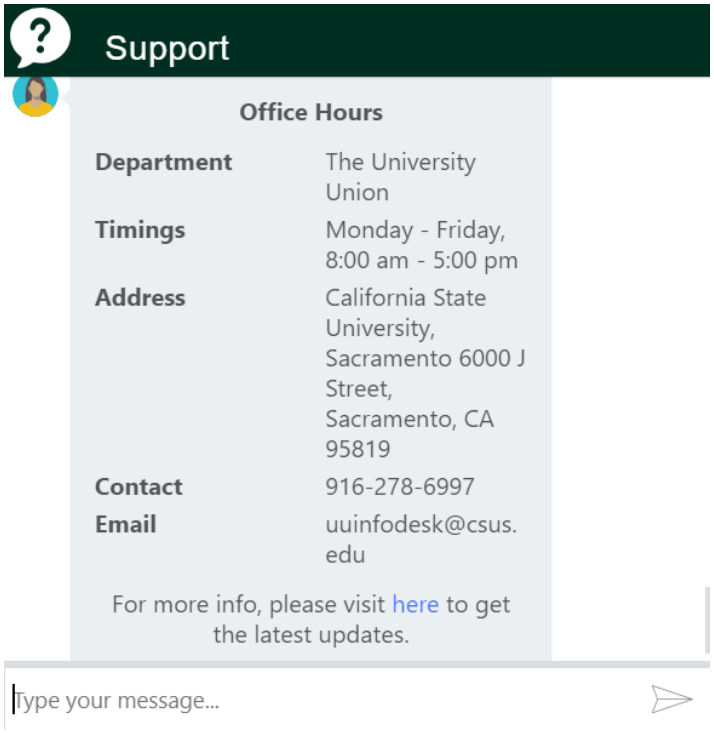


Figure 46: Union Hours Information

4.9 Results

The results of this project are measured in whether sentiment analysis and active learning is correctly implemented or not. Sentiment analysis correctly recognizes the user's query such as positive, negative, and neutral by storing all the conversations in the database (as shown in Figures 33 and 34). These results are used to add empathy to the bot. However, the system was partially successful in adding empathy to the bot. It is because, although large amount of data was added to include some common answers to the queries which are off scripts and to add empathy to the bot (so that it understands what is the current mood of the user and responds accordingly), since scope of these queries is vast, the system

requires more rigorous data to handle all the questions which are out of script. This can be illustrated in the Figure 47 where the bot understood that I am in a negative sentiment, so it answered with a proper response. But, it did not understand the meaning of 'I am excited' since bot did not have any training data related to the typed query as shown in Figure 48.

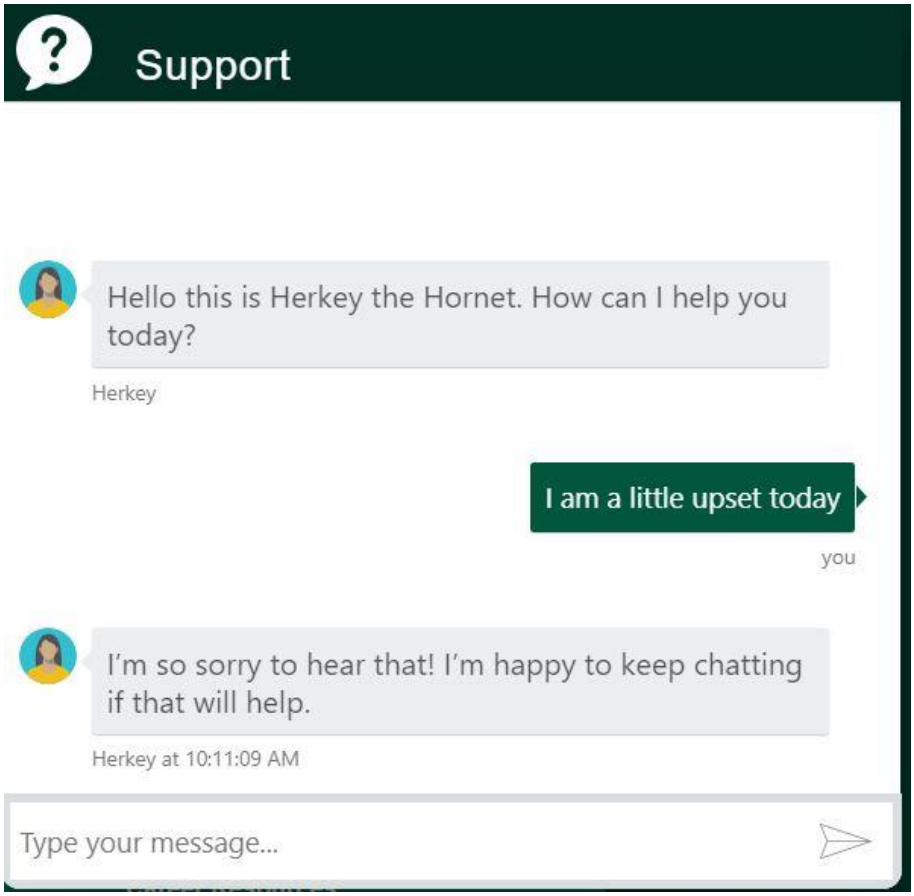


Figure 47: Bot successfully detecting the sentiment of the query

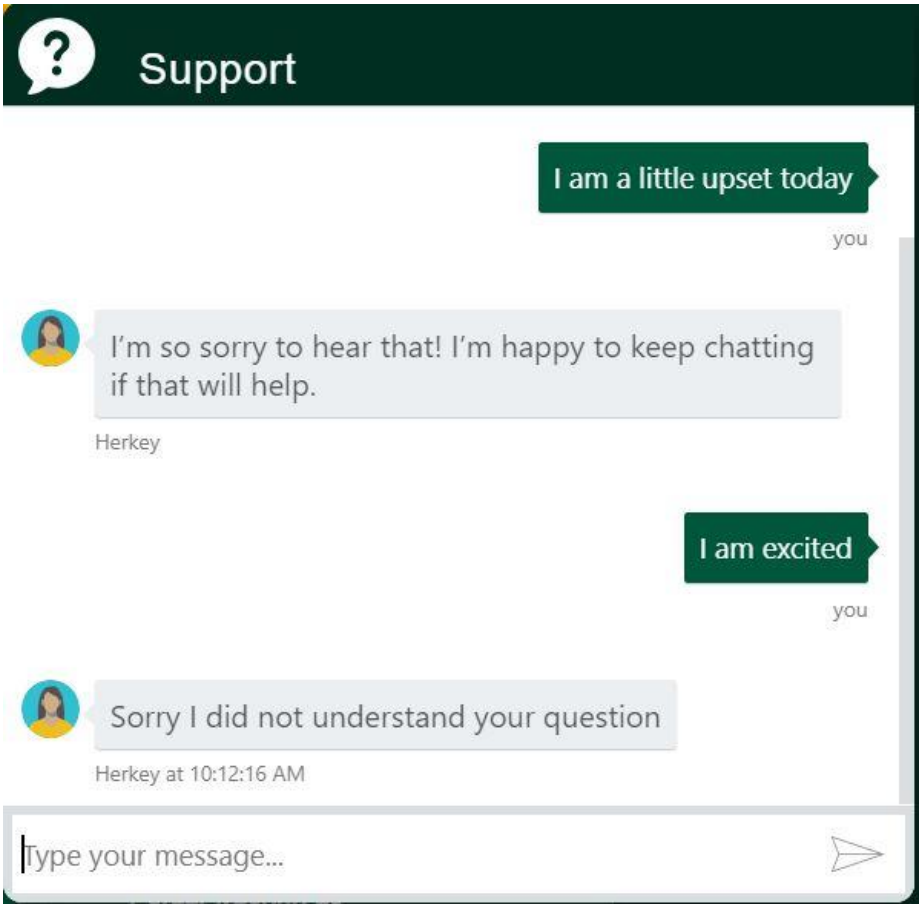


Figure 48: Bot failed to detect the sentiment of the query

On the other hand, active learning helps to improve the bot performance for handling off-script queries. It correctly understands the user's questions, asks clarifying question, and then re-trains the NLP to give response what the user is intended to get.

5 CONCLUSION AND FUTURE WORK

To conclude, College Enquiry Chatbot is helpful in guiding students with correct and most up to date sources of information. It is advantageous for international applicants for queries such as fee payment and academic matters. Students can get the information at their fingertips rather than visiting college office. It improves efficiency by taking over tasks for which humans are not essential.

Sentiment analysis implemented in College Enquiry Chatbot correctly recognizes the user's query such as positive, negative, and neutral by storing all the conversations in the database. However, the system was partially successful in adding empathy since scope of these queries is vast and the system requires more rigorous data to handle all the questions which are out of script. Nevertheless, active learning helps to improve the bot performance for handling off-script queries.

To improve the current functionalities of College Enquiry Chatbot, in the future, the scope of the chatbot can be increased by inserting data for all the departments, training the bot with varied data, testing it on live website, and based on that feedback inserting more training data to the bot. Some of the new features which can be added to the bot are 1) speech recognition feature through which students can ask their queries verbally and get the answers from the bot, 2) integration with multiple channels such as phone call, SMS, and various social media platforms like Skype, Facebook and Twitter, 3) handling context aware and interactive queries in which bot will be aware of the context of an ongoing conversation with a student, 4) integration with services such as password reset and course

enrollment, and 5) adding a capability for the bot to perform analytics based on user's sentiment based on which the bot can be re-trained on human emotions so that more empathy can be added to the bot.

References

[1] "Chatbot", en.wikipedia.org, 2010. [Online]. Available: https://en.wikipedia.org/wiki/Chatbot. [Accessed Jul. 10, 2018].

[2] "Chatbot: What is Chatbot? Why are Chatbots Important? -", Expertsystem.com, 2018. [Online]. Available: https://www.expertsystem.com/chatbot/. [Accessed Sep. 11, 2018].

[3] "Amazon.com: Online Shopping for Electronics, Apparel, Computers, Books, DVDs & more", Amazon.com, 2017. [Online]. Available: https://www.amazon.com/. [Accessed Sep. 12, 2018].

[4] "Amazon Alexa - Build for Amazon Echo Devices", Developer.amazon.com, 2017. [Online]. Available: https://developer.amazon.com/alexa. [Accessed Sep. 12, 2018].

[5] "Meet Erica, Your Financial Digital Assistant From Bank of America", Bank of America, 2018. [Online]. Available: https://promo.bankofamerica.com/erica/. [Accessed Sep. 12, 2018].

[6] C. Doug Gross, "Facebook: Get Messenger app or else - CNN", CNN, 2014. [Online]. Available: https://www.cnn.com/2014/07/29/tech/social-media/facebook-messenger/index.html. [Accessed Sep. 12, 2018].

[7] A. Storman, "5 Chatbot Challenges and How to Overcome Them", Chatbots Magazine, 2016. [Online]. Available: https://chatbotsmagazine.com/5-chatbot-challenges-and-how-to-overcome-them-caccc3a26d7c. [Accessed Sep. 11, 2018].

[8] A. M. Rahman, A. A. Mamun and A. Islam, "Programming challenges of chatbot: Current and future prospective," *in 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Dhaka, 2017.*

[9] "Problems with Chatbots Today - Attunix", Attunix, 2018. [Online]. Available: https://attunix.com/problems-chatbots-today/. [Accessed Sep. 11, 2018].

[10] "Azure Bot Service - chatbot | Microsoft Azure", Azure.microsoft.com, 2017. [Online]. Available: https://azure.microsoft.com/en-us/services/bot-service/. [Accessed Jul. 22, 2019].

[11] "LUIS (Language Understanding) – Cognitive Services – Microsoft Azure", Luis.ai, 2017. [Online]. Available: https://www.luis.ai/home. [Accessed Jul. 15, 2018].

[12] "QnA Maker", Qnamaker.ai, 2016. [Online]. Available: https://www.qnamaker.ai/. [Accessed Jul. 17, 2018].

[13] "What is Azure Bot Service? - Definition from WhatIs.com", WhatIs.com, 2018. [Online]. Available: https://whatis.techtarget.com/definition/Azure-Bot-Service. [Accessed Jan. 17, 2019].

[14] "Azure App Service Documentation - Tutorials, API Reference", Docs.microsoft.com, 2015. [Online]. Available: https://docs.microsoft.com/en-us/azure/app-service/. [Accessed Dec. 17, 2018].

[15] "Cognitive Services | Microsoft Azure", Azure.microsoft.com, 2017. [Online]. Available: https://azure.microsoft.com/en-us/services/cognitive-services/. [Accessed Dec. 22, 2018].

[16] "LUIS.AI: Automated Machine Learning for Custom Language Understanding", azure.microsoft.com, 2017. [Online]. Available: https://azure.microsoft.com/en-us/blog/luis-ai-automated-machine-learning-for-custom-language-understanding/. [Accessed Jul. 22, 2019].

[17] "Review user utterances - Language Understanding - Azure Cognitive Services", Docs.microsoft.com, 2019. [Online]. Available: https://docs.microsoft.com/en-us/azure/cognitive-services/luis/luis-how-to-review-endoint-utt. [Accessed Sep. 14, 2018].

Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 09, September-2020

College Enquiry Chat-Bot System

Harshala Gawade
Computer Department
Saraswati College of Engg Kharghar, India

Prachi Vishe
Computer Department
Saraswati College of Engg Kharghar, India

Vedika Patil
Computer Department
Saraswati College of Engg Kharghar, India

Sonali Kolpe
Computer Department
Saraswati College of Engg Kharghar, India

***Abstract*—Nowadays, many people are using smartphone with many new applications i.e. technology is growing day by day. Today Artificial Intelligence is playing a major role in a variety of fields ranging from industries in product manufacturing, to customer care in public relations. As there are many online Artificial Intelligence (AI) systems or chat bots which are in existence that help people solve their problems. So, we are going to implement a virtual assistant based on AI that can solve any college related query. This will work as a College Oriented Intelligence machine. This virtual machine will respond the queries of students on college related issues. A chat bot has information stored in its database to identify the sentences and making a decision itself as response to answer a given question. The college enquiry chat bot will be built using algorithm that analyses queries and understand user's message.**

Keywords— Artificial Intelligence, Database, Intelligence Machine.

I. INTRODUCTION

A chatbot is a software application used to conduct an on-line chat conversation via text or text-to-speech, in lieu of providing direct contact with a live human agent. Designed to convincingly simulate the way a human would behave as a conversational partner [14]. Bots can be created by using language like Artificial Intelligence Mark-up Language(AIML), a language based on XML that allow developer's write rules for the bot to follow. Another drawback is writing rules for different scenarios is very time consuming and it is impossible to write rules for every possible scenario. So these bots can handle simple queries but fail to manage complex queries is stated in paper [7]. In paper [2] the chat-bot system is been proposed and designed using chat fuel platform and integrated in Facebook page. The chatbot has been designed to provide students feel like talking to the staff from college and their queries are addressed through the conversational text. Responses can be provided to the user in text format, pictures and with many more features provided by the chat fuel. The setup AI feature makes the bot smart and answers the queries of user [2].

The purpose of developing this project is based on an intellectual chat-bot system which will deal with the academic activities like admission enquiry, fees structure, scholarship details, time-table of every department, details of the documents required to attach etc. With this chat-bot system it will be easy for the student to directly clear their queries in lesser time.

II. LITERATURE SURVEY

A literature survey is a comprehensive summary of previous research on a topic. The literature review surveys scholarly articles, books, and other sources relevant to a particular area of research. It should give a theoretical base for the research and help you (the author) determine the nature of your research. [14]

Prof. Ram Manoj Sharma [2] proposed a college enquiry chatbot system which has been built by using Artificial Intelligence algorithms. The bot analyses user's query and understands user messages. The system has modules like Online chatbot, Online Noticeboards etc[2].

P.Nikhila, G.Jyothi, K.Mounika, Mr. C Kishor Kumar Reddy and Dr. B V Ramana Murthy [3], they have designed using AIML (Artificial Intelligence Mark-up Language) to make response to queries. AIML is employed to make or customize alicebot that could be a chat-bot application supported ALICE free code [3].

Harsh Pawar , Pranav Prabhu, Ajay Yadav, Vincent Mendonca , Joyce Lemos [6], a chatbot is designed by them using knowledge in database. The proposed system has Online Enquiry and Online Chatbot System. The development is done using various programming languages by creating a user friendly graphical interface to send and receive response. The main purpose is it uses SQL (Structured Query Language) for pattern matching which is been stored in program [6].

Nitesh Thakur, Akshay Hiwrale, Sourabh Selote, Abhijeet Shinde and Prof. Namrata Mahakalkar [10], proposed an artificial chatbot using NLP (Natural Language Processing) which can be done in two ways the first via written text and the second is via verbal or voice communication. Written communication is much easier than the verbal communication. This paper introduces an interest in some emerging capabilities for evolving speed understanding and processing in virtual human dialogue system [10].

III. PROPOSED METHODOLOGY

The proposed methodology makes use of both qualitative and quantitative perspectives, and includes a broad array of approaches such as literature reviews, expert opinions, focus groups, and content validation [14].

The proposed system will have the following modules:

A] Online Enquiry:

Students can enquire about facilities and query related to exams, academics, fee structure, etc.

Students can also ask questions related to placement activities.

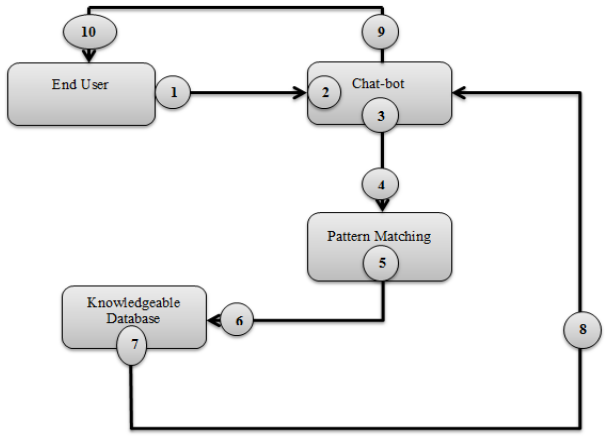
International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 09, September-2020

B] Online Chatbot:

The result can be showed in the form of images and card format or in text format. The query will be answered on the basis of questions asked and the language model built and also the response media created.

Users that want to enquire about the college at the time of admission or any competition held in the college can query to the chat-bot.

Given below is the system architecture of this chat-bot:



The basic algorithm that will be implemented for working of this proposed system is as follows:

Step 1: Start.

Step 2: Get the input query from the user.

Step 3: The query is pre-processed. E.g. suppose there is this query “what are the project domains for CSE fourth year major projects.” So, we are going to remove these stop words like “are”, “the” using pre-processing technique.

Step 4: Fetch the remaining keywords from the query.

Step 5: Match the fetched keywords with the keywords in Knowledge base, and provide an appropriate response.

Step 6: Further the Database module is used to call proper services using entity information to find proper data.

Step 7: The keywords will be matched with the help of keyword matching algorithm.

Step 8: It returns the query response to the bot.

Step 9: Chat-bot packages the data into proper response for display by the client.

Step 10: Exit

IV. CONCLUSION

The goal of the system is to help the students to stay updated with their college activities. Artificial Intelligent is the fastest growing technology everywhere in the world, with the help of Artificial Intelligent and Knowledgeable database. We can make the transformation in the pattern matching and virtual assistance. This system is developing chat bot based on android system so with the combination of Artificial Intelligent Knowledgeable database and virtual assistance. We can develop such chat bot which will make a conversion between human and machine and will satisfy the question raised by user. The main motive of the project is to reduce the work load on the college's office staff and reduce the response time to a user's query.

V. REFERENCES

[1] Ms.Ch.Lavanya Susanna, R.Pratyusha, P.Swathi, P.Rishi Krishna, V.Sai Pradeep, “College Enquiry Chatbot”, International Research Journal of Engineering and Technology (IRJET), e-ISSN: 2395-0056, p-ISSN: 2395-0072, Volume: 07 Issue: 3 Mar 2020 pp 784-788.

[2] Assistant Prof Ram Manoj Sharma, “Chatbot based College Information System”, RESEARCH REVIEW International Journal of Multidisciplinary, ISSN: 2455-3085 (Online), Volume-04, Issue-03, March-2019, pp 109-112.

[3] P.Nikhila, G.Jyothi, K.Mounika, Mr. C Kishor Kumar Reddy and Dr. B V Ramana Murthy on , “Chatbots Using Artificial Intelligence”, International Journal of Research and Development,Volume VIII, Issue I, January/2019, ISSN NO:2236-6124, pp 1-12.

[4] Payal Jain, “College Enquiry ChatBot Using Iterative Model”, International Journal of Scientific Engineering and Research (IJSER),ISSN (Online): 2347-3878, Volume 7 Issue 1, January 2019, pp 80-83.

[5] Sagar Pawar, Omkar Rane, Ojas Wankhade, Pradnya Mehta, “A Web Based College Enquiry Chatbot with Results”, International Journal of Innovative Research in Science, Engineering and Technology, ISSN(Online): 2319-8753, ISSN (Print): 2347-6710, Vol. 7, Issue 4, April 2018, pp 3874-3880.

[6] Harsh Pawar , Pranav Prabhu, Ajay Yadav, Vincent Mendonca , Joyce Lemos, “College Enquiry Chatbot Using Knowledge in Database”, International Journal for Research in Applied Science & Engineering Technology (IJRASET), ISSN: 2321-9653; IC Value: 45.98, SJ Impact Factor: 6.887, Volume 6, Issue IV, April 2018, pp 2494- 2496.

[7] Jincy Susan Thomas, Seena Thomas, “Chatbot Using Gated End-to-End Memory Networks”, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056, p-ISSN: 2395-0072, Volume: 05 Issue: 03 Mar 2018, pp 3730- 3735.

[8] Prof. Suprita Das, Prof. Ela Kumar, “Determining Accuracy of Chatbot by applying Algorithm Design and Defined process”, 4th International Conference on Computing Communication and Automation (ICCCA), 2018, 978-1-5386-6947-1/18/2018 IEEE, pp 1-6.

[9] Prof.K.Bala, Mukesh Kumar ,Sayali Hulawale, Sahil Pandita, “Chatbot For College Management System Using A.I”, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056, p-ISSN: 2395-0072, Volume: 04 Issue: 11 | Nov -2017, pp 2030-2033.

[10] Nitesh Thakur, Akshay Hiwrale, Sourabh Selote, Abhijeet Shinde and Prof. Namrata Mahakalkar, “Artificially Intelligent Chatbot”, Universal Research Reports , ISSN : 2348 – 5612, Volume : 04 , Issue : 06, July - September 2017, pp 43-47.

[11] Amey Tiwari, Rahul Talekar, Prof.S.M.Patil, “College Information Chat Bot System”, International Journal of Engineering Research and General Science, ISSN 2091-2730, Volume 5, Issue 2, March-April, 2017, pp 131-137.

[12] Malusare Sonali Anil, Kolpe Monika Dilip, Bathe Pooja Prashant, “Online Chatting System for College Enquiry using Knowledgeable Database”, Savitribai Phule Pune University, Preliminary Project Report, pp 1-53,2017.

[13] Balbir Singh Bani, Ajay Pratap Singh, “College Enquiry Chatbot Using A.L.I.C.E (Artificial Linguistic Internet Computer Entity)”, International Journal of New Technology and Research (IJNTR), ISSN:2454-4116, Volume-3, Issue-1, January 2017 Pages 64-65. https://www.google.com/

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:03/March-2023 Impact Factor- 7.868 www.irjmets.com

ARTIFICIAL INTELLIGENCE BASED COLLEGE ENQUIRY CHATBOT

Mansi Vaidya[*1], Pratika Takitkar[*2], Ravina Potpose[*3],
Prof. Mamta Balbudhe[*4]

[*1,2,3]Department Of Computer Engineering, SRPCE College Of Engineering,
Nagpur, Maharashtra, India.

[*4]Prof., Dept. Of Computer Engineering, SRPCE College Of Engineering,
Nagpur, Maharashtra, India.

DOI : https://www.doi.org/10.56726/IRJMETS34820

ABSTRACT

A chat bot is a computer program that may initiate conversations between people and other computers. With chatbot technology, it is text-based and may be utilized securely by a bigger audience. Chatbots for university research are developed using synthetic algorithms that interpret user messages and assess user demands. The goal of the chatbot's responses is to match the user's input while avoiding making oneself physically available to the institution in response to queries. The system uses its intelligence to answer to the pupils' questions.

Keywords: Artificial Intelligence, College Enquiry, Deep Learning, Cloud Computing, Machine Learning.

I. INTRODUCTION

The objective of AI (Artificial Intelligence) is to establish natural communication between humans and robots. The fastest-growing field of AI today is dialogue systems, sometimes known as interactive conversation systems. A chatbot is a programmer that uses artificial intelligence techniques like Natural Language Processing (NLP), video and image processing, and audio analysis to simulate human conversations in their natural format, including text. The artificial intelligence algorithms that will be used to create this project will study user demands. The system assists with university-related tasks including anniversary, sports day, admission, and other cultural events activities by responding to the user through an efficient graphical user interface (GUI). It will make it easier for users and students to stay informed about university events.

II. LITERATURE SURVEY

A chatbot system for college inquiries that has been created using artificial intelligence algorithms was developed by Prof. Ram Manoj Sharma [2]. The bot interprets user communications and analyses user queries. The system includes modules such as online chatbots and noticeboards, among others [2].

AIML (Artificial Intelligence Mark-up Language) has been created by P. Nikhila, G. Jyothi, K. Mounika, Mr. C. Kishor Kumar Reddy, and Dr. B. V. Ramana Murthy [3] to make responses to enquiries. To create or modify Alice Bot, which might be a chat-bot program supported by ALICE free code, AIML is used [3].

The suggested system by Payal Jain will be built on the Alice and AIML algorithm and be used to find solutions to user-submitted queries. It is necessary to create a database where all relevant data will be kept, as well as a user interface. One section of the Android interface will be designed for basic users, while the other will be for the administrator. A background investigation was conducted, which covered an overview of the conversation process and any existing chatbots that were pertinent. We'll create a database to house data about queries, responses, keywords, logs, and feedback messages [4].

A chatbot was created by Harsh Pawar, Pranav Prabhu, Ajay Yadav, Vincent Mendonca, and Joyce Lemos [6] utilizing knowledge from a database. The suggested solution includes an online inquiry and chatbot technology. A user-friendly graphical interface is developed utilizing a variety of computer languages to transmit and receive responses. The primary objective is to match patterns contained in the application using SQL (Structured Query Language) [6].

Using NLP (Natural Language Processing), which may be done in two ways—the first via written text, and the second by verbal or vocal communication—Nitesh Thakur, Akshay Hiwrale, Sourabh Selote, Abhijeet Shinde, and Prof. Namrata Mahakalkar presented an artificial chatbot. Verbal communication is far more difficult than

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:03/March-2023 Impact Factor- 7.868 www.irjmets.com

written communication. The interest in certain newly developed understanding and processing speed capabilities for growing speed in virtual human dialogue systems is introduced in this research [10].

III. METHODOLOGY

This system is an online program that responds to user questions. It uses a chatbot with artificial intelligence that acts as a substitute for awareness to respond to inquiries about colleges. Technologies for natural language processing are used to filter, stem, and tokenize the content of complaints. The system analyses the question and, following the client's responses, provides an answer that appears to have been provided by a real person. The system can also respond to the student's general questions. The following flow chart can be used to understand the algorithm of the entire system.

IV. MODELING AND ANALYSIS

Login Module

In this portal, student will able to login into system by providing its username and password.

Sign Up Module

The sing up module in this block student can able to register a new account into system by entering details like Name, Branch Semester, Year, Photo, Roll No., etc.

Chat Bot Interface Module

It is providing an effective Graphical user interface which implies that as if a real person is talking to the user and this module can be accessible through anywhere the student just has to enter the query in chat section which the user can chat by ask queries related to college and department related activities.

Chat Bot Engine Module

This section is responsible to answer the student query in this module we are going to use machine learning power and will create one ML model which will responsible to answer the student query and it uses the Artificial Intelligence and NLP to understand the user intention and answers their query.

Google web scrapper

We will create our own scrapper which will responsible to fetch result of subject related query from google.

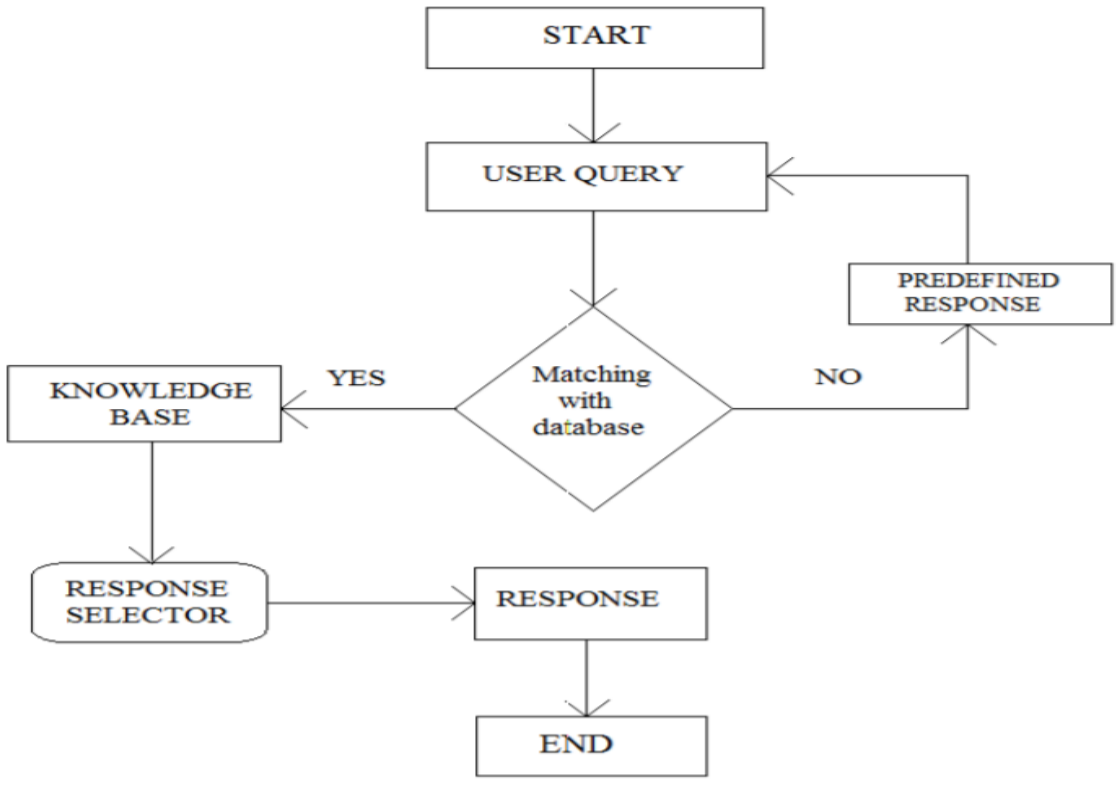


Figure 1: DFD

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:03/March-2023 Impact Factor- 7.868 www.irjmets.com

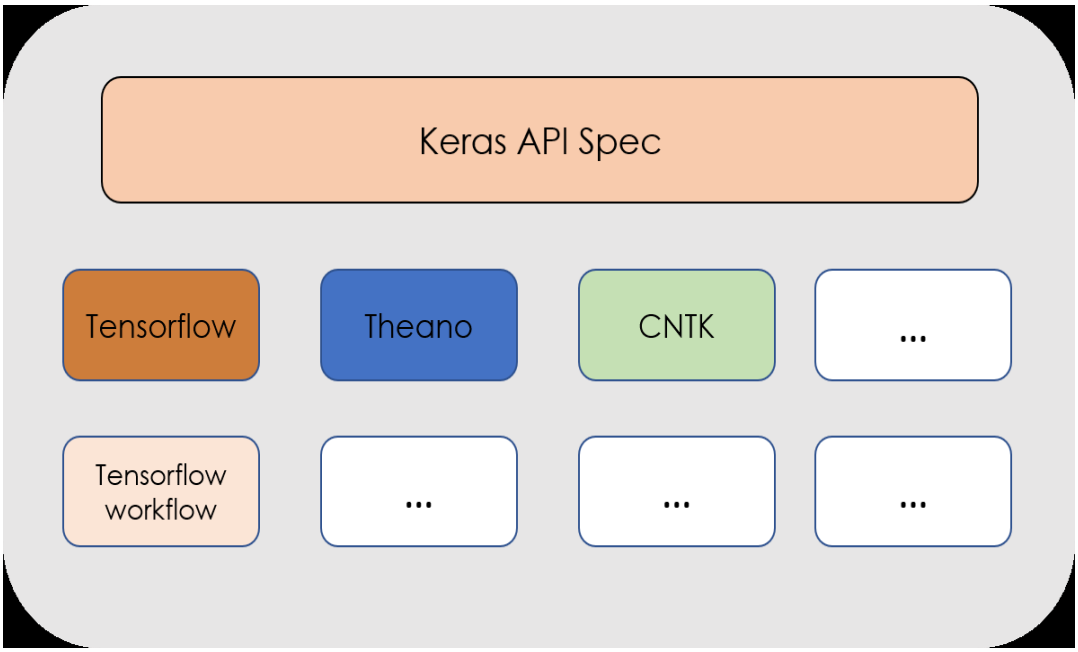


Figure 2: Algorithm architecture

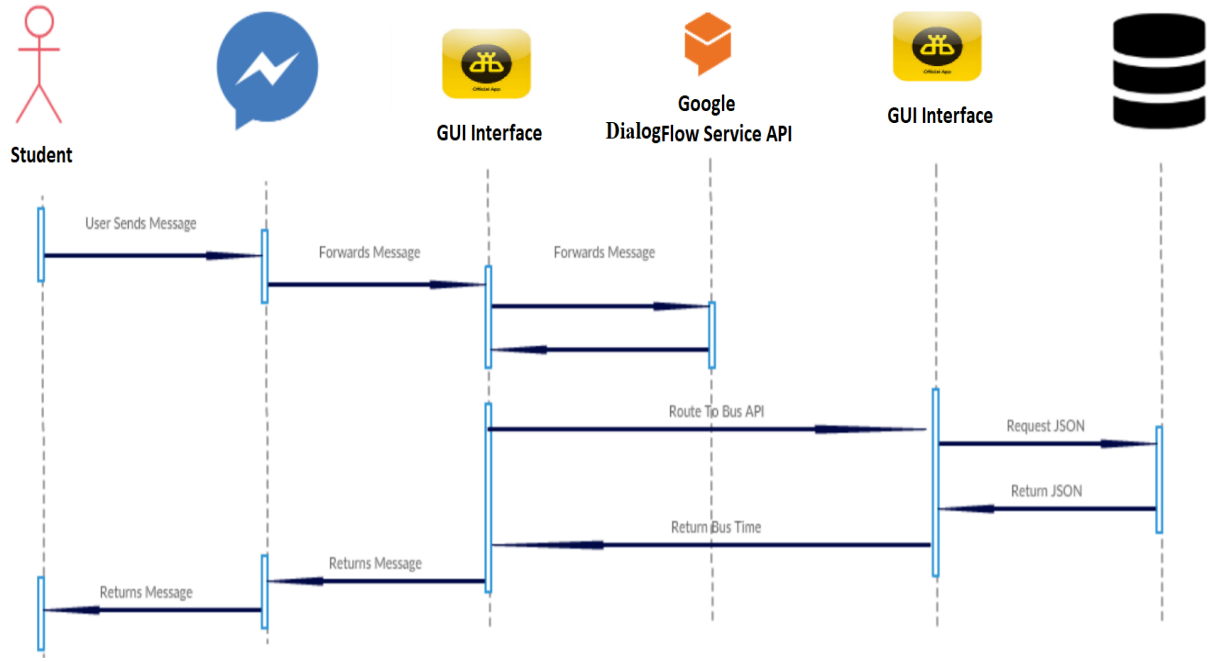


Figure 3: Architecture of System

An open source, high level library for creating neural network models is called Keras. It was created by Google Deep Learning researcher François Chollet. Its fundamental idea is to simplify the creation of neural networks, their training, and use in prediction while still enabling developers to completely customise the ANN's settings. In essence, Keras is essentially an interface that may be used with a variety of Deep Learning frameworks, such as CNTK, Tensorflow, or Theano. Regardless of the back-end that is employed, it functions the same. The Keras API's layered architecture.

In a Neural Network, each node in a specific layer takes the weighted sum of the outputs from the previous layer, applies a mathematical function to them, and then passes that result to the next layer. With Keras, we can create a block representing each layer, where these mathematical operations and the number of nodes in the layer can be easily defined. As can be seen, it can run on top of different frameworks without any issues. One logical line of code may be used to build these several levels.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:03/March-2023 Impact Factor- 7.868 www.irjmets.com

V. PROPOSED SYSTEM ARCHITECTURE

- Using NLP techniques, the data is preprocessed, and the user's input sentence is divided into a number of questions and answers using information from the conversation file.
- The words are then lowered, stemmed back to their original form, and the contractions are eliminated.
- It incorporates the words into a vocabulary dictionary that contains certain integers, converting the input text into an array of integers that the encoder can comprehend.
- Also, we'll develop an inverted vocabulary that will translate the numbers back into the words that a given key is associated with.
- The creation of a set of tokens, including padding, start of token (SOS), end of token (EOS), and so on, will come next.
- By adding the tokens and padding the sentence's length to a predetermined length, the encoder and decoder inputs will then be formed.
- Due to the encoder outputs already including the start of the string token, we will raise the decoder input by one time step.
- The encoder and decoder LSTM layers will be present. The word embedding will then be produced using this LSTM recurrent neural network.
- The decoder receives the states of the encoder and passes them on, using the decoder's output to anticipate the next word and send back to the encoder.
- This decoder layer will continue to loop until it anticipates the end of the statement token or reaches the response's maximum size.

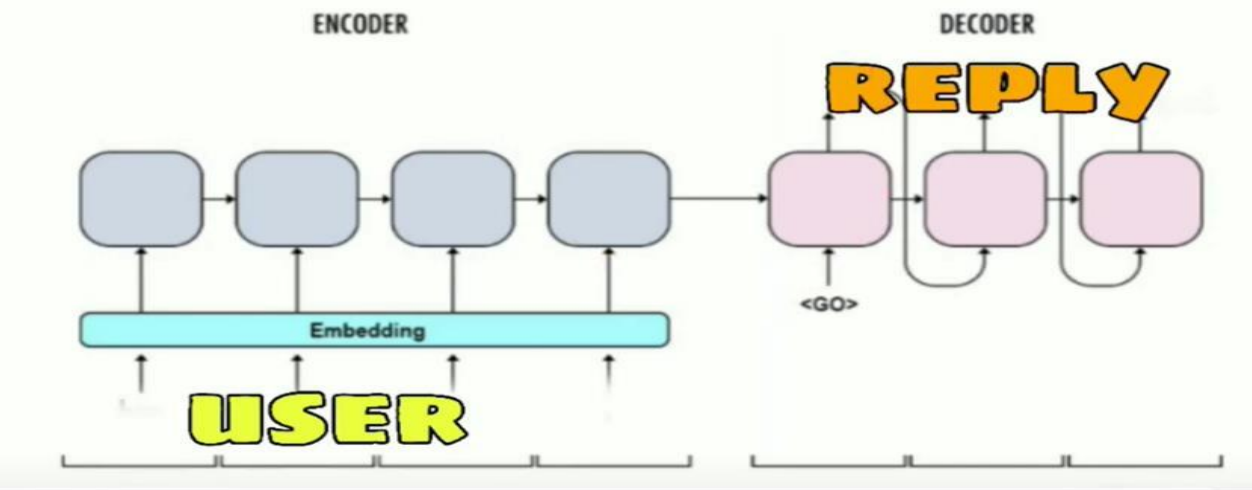


Figure 4: Communication Encoder & Decoder

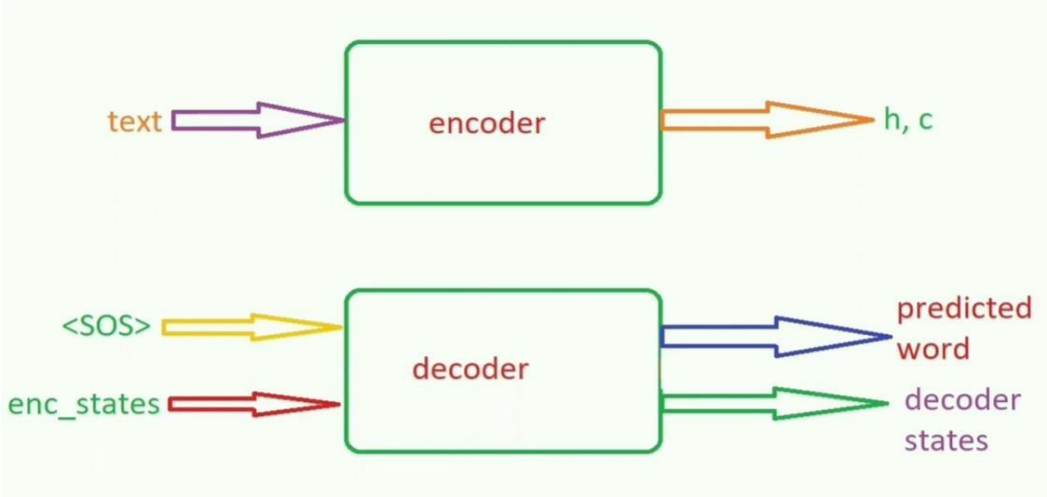


Figure 5: Communication Encoder & Decoder

VI. RESULTS AND DISCUSSION

It should be easy to interact with and use the chatbot. Maintaining the conversation's flow while taking into account the context from earlier exchanges is the chatbot industry's most common problem. The poor processing of natural language and the lack of data provide additional challenges. The accuracy of the model is inversely correlated with the level of granularity of the dataset. The most common issue in the realm of chatbots is maintaining the flow of conversation while taking the context from multiple conversations in the past. The model could have trouble comprehending or transmitting sarcasm and other figures of speech. The poor processing of natural language and the lack of data provide another difficulty. The accuracy of the chatbot's model depends on the level of granularity of the dataset. Sarcasm and other figures of speech could be difficult for the model to understand or express.

VII. CONCLUSION

Each college will utilize this software technology to enable students to freely post their questions. The complaint's content is parsed, tokenized, stemmed, and filtered using natural language processing tools. The chatbot receives the output and calculates the sentence's strength using it. Calculating the strength of denial aids in automatically prioritizing the issue for the service provider to address. The suggested approach will enable numerous enterprises, including any college, to guarantee customer satisfaction and high-quality service with less resources.

VIII. REFERENCES

[1] Ms.Ch.Lavanya Susanna, R.Pratyusha, P.Swathi, P.Rishi Krishna, V.Sai Pradeep, "College Enquiry Chatbot", International Research Journal of Engineering and Technology (IRJET), e-ISSN: 2395-0056, p-ISSN: 2395-0072, Volume: 07 Issue: 3 Mar 2020 pp 784-788.

[2] Assistant Prof Ram Manoj Sharma, "Chatbot based College Information System", RESEARCH REVIEW International Journal of Multidisciplinary, ISSN: 2455-3085 (Online), Volume-04, Issue-03, March-2019, pp 109-112..

[3] P.Nikhila, G.Jyothi, K.Mounika, Mr. C Kishor Kumar Reddy and Dr. B V Ramana Murthy on , "Chatbots Using Artificial Intelligence", International Journal of Research and Development, Volume VIII, Issue I, January/2019, ISSN NO:2236-6124, pp 1-12.

[4] Payal Jain, "College Enquiry ChatBot Using Iterative Model", International Journal of Scientific Engineering and Research (IJSER),ISSN (Online): 2347-3878, Volume 7 Issue 1, January 2019, pp 80-83.

[5] Sagar Pawar, Omkar Rane, Ojas Wankhade, Pradnya Mehta, "A Web Based College Enquiry Chatbot with Results", International Journal of Innovative Research in Science, Engineering and Technology, ISSN(Online): 2319-8753, ISSN (Print): 2347-6710, Vol. 7, Issue 4, April 2018, pp 3874-3880.

[6] Harsh Pawar , Pranav Prabhu, Ajay Yadav, Vincent Mendonca, Joyce Lemos, "College Enquiry Chatbot Using Knowledge in Database", International Journal for Research in Applied Science & Engineering Technology (IJRASET), ISSN: 2321-9653; IC Value: 45.98, SJ Impact Factor: 6.887, Volume 6, Issue IV, April 2018, pp 2494- 2496

[7] Jincy Susan Thomas, Seena Thomas, "Chatbot Using Gated End-to-End Memory Networks", International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056, p-ISSN: 2395-0072, Volume: 05 Issue: 03 Mar 2018, pp 3730- 3735

[8] Prof. Suprita Das, Prof. Ela Kumar, "Determining Accuracy of Chatbot by applying Algorithm Design and Defined process", 4th International Conference on Computing Communication and Automation (ICCCA), 2018, 978-1-5386-6947-1/18/2018 IEEE, pp 1-6.

[9] Prof.K.Bala, Mukesh Kumar ,Sayali Hulawale, Sahil Pandita, "Chatbot For College Management System Using A.I", International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056, p-ISSN: 2395-0072, Volume: 04 Issue: 11 | Nov -2017, pp 2030-2033.

[10] Nitesh Thakur, Akshay Hiwrale, Sourabh Selote, Abhijeet Shinde and Prof. Namrata Mahakalkar, "Artificially Intelligent Chatbot", Universal Research Reports , ISSN : 2348 – 5612, Volume : 04 , Issue : 06, July - September 2017, pp 43-47.

IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

CHATBOT FOR COLLEGE ENQUIRY

[1]Emil Babu, [2]Geethu Wilson

[1]Msc Scholar, [2]Assistant Professor

[1]Computer Science Department,

[1]St Joseph's College Autonomous, Irinjalakuda, India

Abstract: Frequently we tend to tend to pay our time interrelate with numerous chatterboxes on the net, mostly targeted at such functions or just amusement. The chatbots have embedded information that helps them acknowledge the user's question and provide an answer to it. The college enquiry chatbot project is meant exploitation algorithms that interpret user queries and understand user's message. The college enquiry chatbot project is developed exploitation algorithms that analyze user queries and understand user message. This method is a web application that provides answers to the student's question. Students would really like simply question through the bot. The program analyzes the user's query and answers. Then the bot responds to the query, as if the real person were asking it. The program responds to the students' question with the help of algorithms. The system can have an internet board which can browse any text through the links, this will ease the user get the relevant notifications changed. The user won't waste heaps of your time checking out the suitable notices.

Key Words **- Chatbot, Database, Python, MySQL.**

I. INTRODUCTION

Chatbot was a computer application which can speak to kinsmen naturally, the manner we interact with one anothe . It can put back a person's for multiple tasks of answering queries. A chatbot is a proxy that interacts with users using painless language. It had been built as an endeavor to hoax humans. Various appeal of chatbots like Customer Service, call centers etc. uses AI terminology to talk with user. One among the prime goals of chatbots is to favor an intelligent human and make it difficult for the receiver of the conversation to know the important working alongside various architectonics and capabilities for his or her usage has widely fill out. The chatbots can prove decent to fool the user into basic cognitive process they're "talking" to a person's being, but are very limited in improving their cognitive content at runtime, and normally have a very little to no means of keeping track of all the colloquial data. Chatbots makes use of machine learning to triumph in AI helping them to know the user query and supply proper response. The chatbots are flourished using the synthetic Intelligence parlance for interacting with the user. This consist a software which can be made up using Python, PHP framework and can help user to talk with machine.

II. MOTIVATION

As students, we have a tendency to gain more of knowledge concerning our school, college and university throughout our course. Generally obtaining this details is very cumbersome and drawn-out. Like obtaining facts concerning our fees structure or the due fees remaining may be a terribly drawn-out method that we have to travel to administration building and notice the right window so explore for a no dues form then fill it with correct information so submit it to the acceptable person so that person can tell us our due fees. So why have this long and worthless process to get this minor information. We as a computer science student are always looking forward to solving the problems around us using the technology that we learn and how to implement them to achieve ease of usage in real life. This is where we thought of using an intelligent bot delivering this information. Think about an application, where all you have to do is just ask. You would like to grasp the fee structure of a student, then ask the bot about is it clear or not it will tell you. There is no need of doing a prolonged and feverish procedure. There is no need of doing a prolonged and feverish procedure. If you want to know how to fill the exam form, no problem our bot will help you about the process and the steps. It can also solve the perplexity once a student is about to join the college. He/she might want to enquire about the fee structure of various colleges and understand their admission procedure. Currently within the current system, it will be a protracted method. You would have to go to various college websites and then look over it. Then our bot can do it for you in seconds all you may have to do is to ask it. Isn't it simple and convenient?

III. EXISTING SYSTEM

In the earlier days students had to visit the college to enquire about details like courses ,fee structure ,admission process and other information's about the college ,which is a tiresome process as well as long process for both parents as well as students. Now a days there are many changes occurred in the Education system with help of advanced technolog. Everything is happening over the internet without any trouble. In those days for enquiring about courses we have to visit the college, but as the days are passing away its completing changing. Collecting the course details, fee structure manually will be hectic procedure and it also needs a manpower. For reducing that manpower and avoid such difficulties and time consuming many devices or systems were emerged day by day.

IV. PROPOSED SYSTEM

A Student Chabot project could be a retrieval-based chatbot that uses AI concepts to possess conversations with humans. Once ever a user asks any question, the bot can first analyze the request, builds a response and send it back to the utilization. The chatbot can break down the user sentence into 2 things: intent and an entity. A retrieval-based chatbot is one that functions are predefined input patterns and set responses. Once the question is entered, the chatbot use a heuristic approach to deliver the suitable response. The retrieval-based model is extensively used to design goal destined chatbots with bespoken options just like the flow and tone of the bot to reinforce the client expertise. ChatBots use pattern matching to classify the text and produce a suitable or best response for the clients. A customary structure of those patterns is "Artificial Intelligence Markup Language" (AIML). The planned System could be a net application that has answers to the queries provided by the scholar or the user. Users can just question through the chatbot that is used for chatting. Students can chat by any format there isn't any specific format the user must follow. The answers are applicable what the user queries. If the answers are found to be invalid or not accessible, then those queries are hold on into the unanswered table that's basically created by the admin. Later those queries will updated by the admin, simply just in case of urgency we are provides a message that "our representatives can get to bear with you shortly". This could be displayed once aggregation the desired data from the user. Admin can browse invalid answer through portal via login System, it's going to permits the admin to get rid of the invalid answer conjointly as in updating the acceptable answer for the question raised by the user. The User can raise any college connected activities through the system. The user does not have to be compelled to personally move to the college for enquiry. The System analyzes the question then answers to the user. The system answers to the query as if it's answered by the real person. The system replies with the assistance of a decent Graphical interface that suggests that as if a real person is rebuke the user. The user can question concerning the college connected activities through on-line with the help of this net application. This technique helps the scholar to be updated concerning the faculty related information.

4.1Proces Flow Chart

This section shows the basic steps that how the chatbot provide answers for the users query will be shown in the following Fow Chart:-

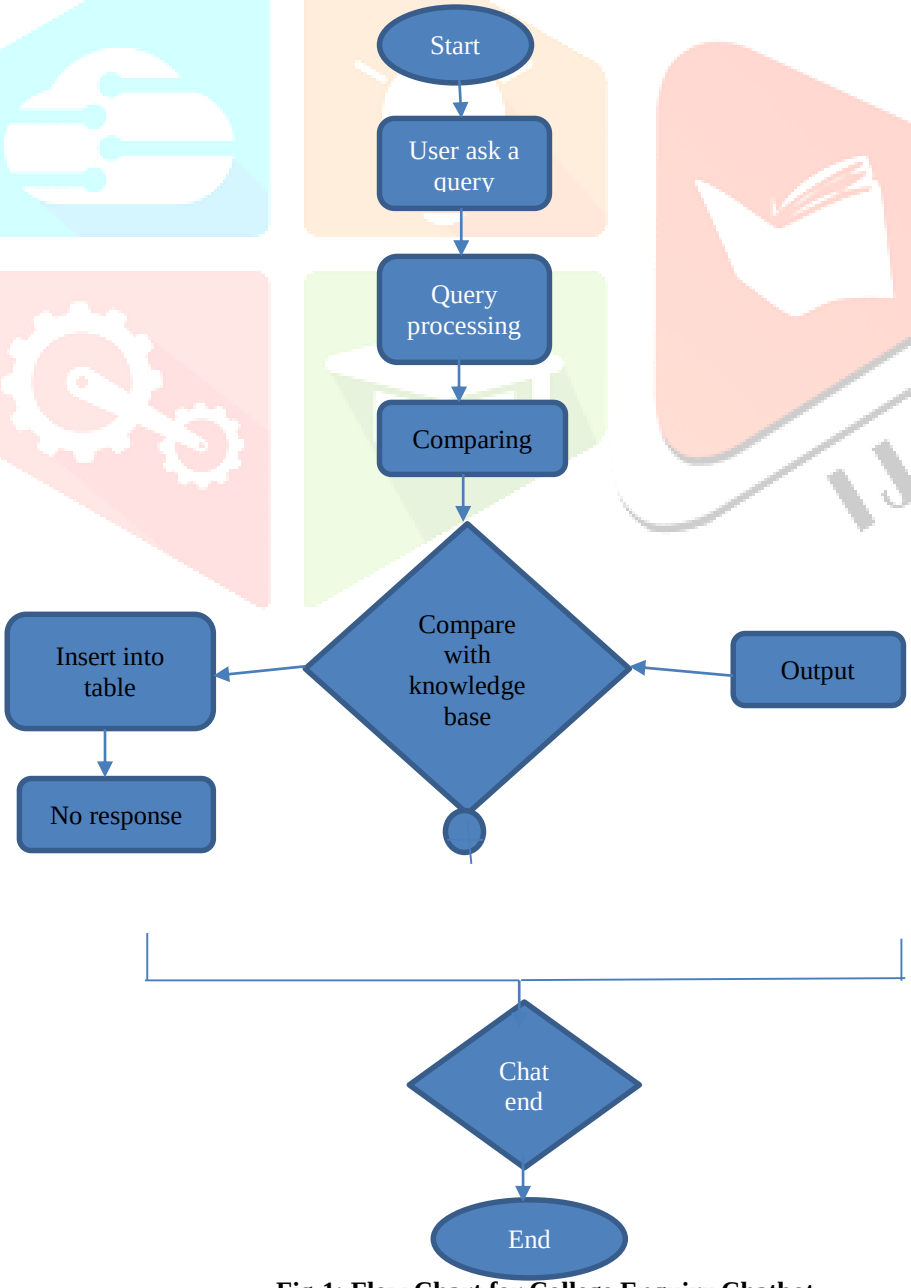


Fig-1: Flow Chart for College Enquiry Chatbot

The Flow Chart describes the entire process of the system or working of chatbot ,if the user cannot find the answer for a query then the bot will answer "sorry I can't understand", in such condition user can send feedback to the admin by using feedback form in the home page. Admin can view the feedback and the corresponding query and answer will be stored and update the data in the database.

4.2 ER Diagram

.

Fig- 2: ER Diagram of ChatBot For College enquiry

IV. IMPLEMENTATION

Block Diagram for proposed system:-

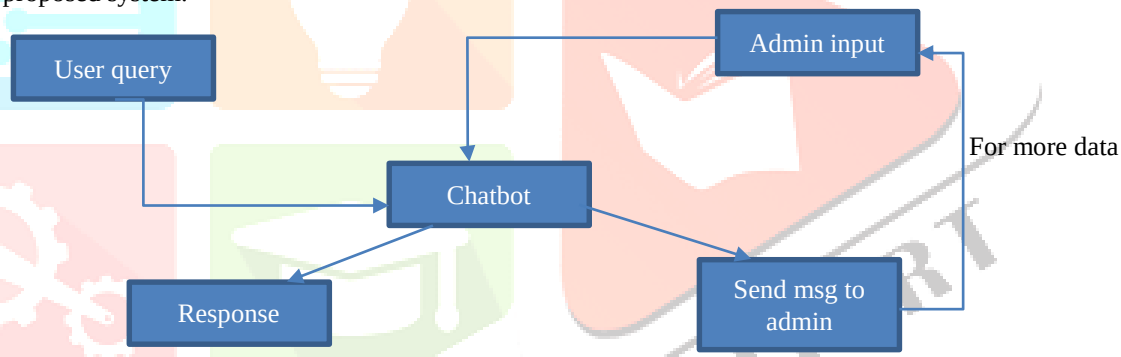


Fig:-3: Block Diagram of ChatBot For College enquiry

From the block diagram it is clearly shows that, the student or any other user ask query to the Chatbot , that query will be stored in the database ,and the answer related to the query will be displayed to the user. If that query is not in the database then the user can ask the admin to add more data to the bot. The entire background process will be done by the admin.

4.1 Creation of Table in Database

A database consisting of different tables. These different tables will store key-words, answer- sentences, mutual weight of each key-word corresponding to each answer-sentence and the lists of words to be filtered out. We are having programs or code for searching through these tables and come up with the most suitable replies or response corresponding to the user's query.

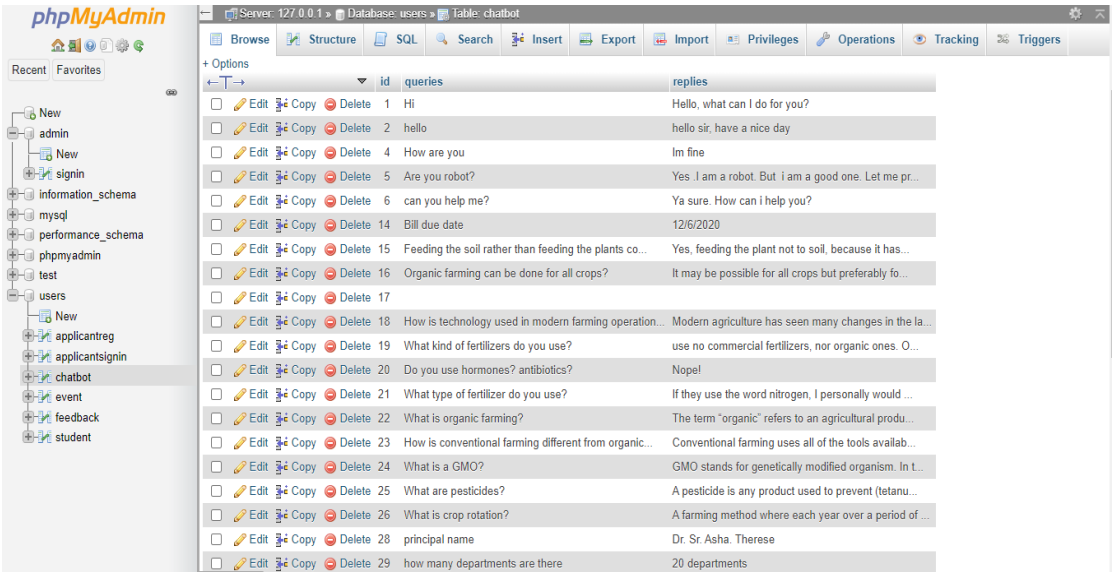


Fig:-4: Tables in Database

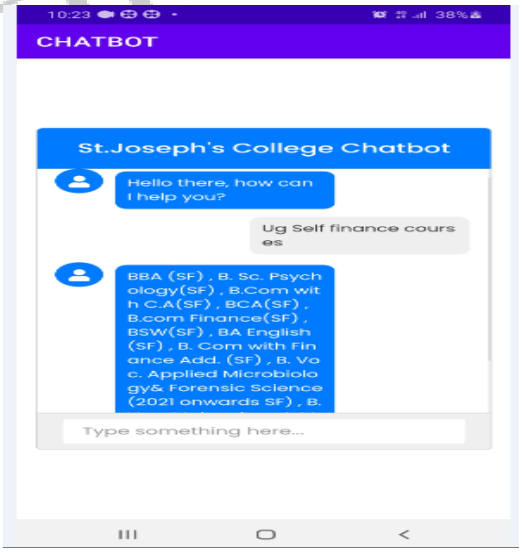
V. APPLCATIONS

- College enquiry chatbots helps the students to get the right sources of information.
- Not Only the college enquiry chatbot any bot that will provide them an instant as well as accurate response or answer.
- Enhance Artificial Intelligence Based ChatBot System,it will be used in most of the colleges and also it can be used in various firms and business-related industries.

VI. RESULTS AND DISCUSSIONS

The proposed system was successfully tested to check its effectiveness and achievability. Chatbot reduces the paperwork, manpower and time for any individual. In this paper we had developed an application which will interact with users by means of reducing the time for visiting the college to enquire about the details/information regarding admissions, college activities etc. The user can chat with the chatbot by format. The user or the student and the Administrator are interacted through a chatbot. The questions which are not answered by the chatbot will be added and updated by the Admin.

6.1 List of output Screens



VII. ACKNOWLEDGEMENT

I express my sincere gratitude to our guide, Lecturer Miss Geethu Wilson for suggestion and support during every stage of this work. And also i convey my deep sense of gratitude to Dr.Sr. Blessy, Head of Computer Science department. Finally, and special thanks to all the staff members in Computer Science department for providing full support.

VIII. CONCLUSION

The main objectives of the project were to develop an algorithm that will be identify the answers associated with user submitted queries. A database is developed to store all related data's and to develop an internet interface. The web interface developed had one parts, and that is for the administrator. A background research happened, including a summary of the conversation procedure and any relevant chat bots available. A database system was designed, that stores information regarding questions, answers, keywords, logs and feedback message.

REFERENCES

[1] Mauldin Michael (1994) ,"ChatterBots , TinyMuds, " and that tureing test : entering the loebner prize competition ", proceedings of the eleventh national conference on artificial intelligence.

[2] Bayu Setiaji, Ferri Wahu Wibawo, "Chatbot Using A Knowledge in the Database", on 2016 7th International Conference on the Intelligent Systems, Modelling and Simulation.

[3] S. J. du Preez, M. Lall and S. Sinha, "An intelligent webbased voice chat bot," EUROCON 2009, EUROCON '09. IEEE, St. - Petersburg, 2009.

[4] E. Haller and T. Rebedea, "Designing a Chat-bot that Simulates an Historical Figure," 2013 19th International Conference on Control Systems and Computer Science, Bucharest, 2013.K. Elissa, "Title of paper if known," unpublished.

[5] Zadrozny, W., et al.: Natural language dialogue for personalized interaction. Commun. ACM 43(8), 116–120 (2000).

[6] Hussain, S., Athula, G.: Extending a conventional chatbot knowledge base to external knowledge source and introducing user based sessions for diabetes education. In: 2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA) 2018. IEEE.

[7] Levesque, H.J.: Common sense, the turing test, and the quest for real AI: Reflections on Natural and Artificial Intelligence. MIT Press, Cambridge (2017) 4. Wikipedia: Chatbot. 29 December 2018 (cited 30 December 2018).

[8] M. L. Mauldin, "ChatterBots TinyMuds and the Turing Test: Entering the Loebner Prize Competition", *AAAI '94 Proceedings of the twelfth national conference on Artificial intelligence*, vol. 1, pp. 16-21, 1994.

[9] "Chatbot Market Size And Share Analysis Industry Report 2014 - 2025", [online] Available: https://www.grandviewresearch.com/industry-analysis/chatbot-market.

[10] "Gartner Top Strategic Predictions for 2018 and Beyond", [online] Available: https://www.gartner.com/smarterwithgartner/gartner-top-strategic-predictions-for-2018-and-beyond/.

[11] A. Følstad and P. B. Brandtzæg, "Chatbots and the new world of HCI", *interactions*, vol. 24, no. 4, pp. 38-42, Jun. 2017, [online] Available: http://doi.acm.org/10.1145/3085558.

[12] Masche, J., Le, N.-T.: A Review of Technologies for Conversational Systems. In: International Conference on Computer Science, Applied Mathematics and Applications. Springer, Cham (2017) .

[13] Mathur, V., Singh, A.: The Rapidly Changing Landscape of Conversational Agents (2018). arXiv preprint arXiv:1803.08419 .

[14] Weizenbaum, J.: ELIZA—a computer program for the study of natural language communication between man and machine. Commun. ACM 9(1), 36–45 (1966) .

[15] Clarke, D.: Three AI Technologies that could make chatbots intelligent. 24 March 2018 (cited 18 December 2018). https://hackernoon.com/three-ai-technologies-that-could-makechatbots-intelligent-10f8c6e8b4b0 .

[16] Wikipedia contributors. ChatScript. Wikipedia, The Free Encyclopedia. 4 September 2018, 19:19 UTC. https://en.wikipedia.org/w/index.php?title=ChatScript&oldid=858055386. Accessed 22 Jan 2019 .

JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

AI College Enquiry Chatbot System

Gokul R [1] Kalaivani S [2]

[1]Department of Computer Application, Hindusthan College of arts and science, Coimbatore, Tamilnadu, India.

[2]Assistant Professor, Department of Computer Application, Hindusthan College of arts and science, Coimbatore, Tamilnadu, India.

Abstract-- In this paper, we present a college enquiry chatbot system designed to provide quick and efficient responses to student queries. The chatbot was built using natural language processing techniques and machine learning algorithms. We evaluated the performance of the system by testing it with a dataset of commonly asked student queries. The results showed that the chatbot was able to provide accurate responses in a timely manner. The system has the potential to save time for both students and staff by providing instant answers to commonly asked questions. Our study highlights the effectiveness of chatbot systems in educational environments and suggests the potential for future research in this area..I conversational assistant is a computer program that can be used for online interaction through text or voice messages. They can make human interaction contextual which leads to actual engaging interaction. Python's Rasa X is a framework that is a powerful tool in the creation of chatbots that can serve as a college enquiry system. This research study is focused on identifying the requirements for the development of a dynamic chatbot that supports text as well as voice-based interaction with the users.

Key Words: Artificial Intelligence, Rasa X, Machine learning, College enquiry chatbot, Human-computer interaction, Natural language processing, Speech-to-text.

1. INTRODUCTION

College students often need information regarding college such as timetable, upcoming events of college, about faculties, exam timetables, new assignments and projects with their deadline many more. The traditional way such as making phone calls, browsing the school website, or sending emails is inconvenient and time-consuming since you have to wait for a long time before you get an answer from the other side. If physical support is not available then calling doesn't help in some cases. Chatbots are changing the way of communication over the past few years it has been observed that most of the websites refer to chatbot interaction over actual physical calling support. So, a solution such as a VoiceChatbot is the easiest system to access for any user and is available 24 x 7. Anyone, Anywhere, Anytime without any problem can make use of internet connection and mobile device or other laptop devices to solve doubts. The objective of this research study is to identify the appropriate software components needed for developing a chatbot support system using Machine Learning and Natural Language Processing which can take input through voice as well as text and can easily extract intent and entity of user's message as it can be easily integrated with college website to provide precise and accurate answers to college-related queries by students or parents. To minimize the load of offices of universities and increase interaction between student and college without actually involving any physical human entity that can promote student engaging system with great user interface and quick response.

2. LITERATURE SURVEY

Authors Siti Nazurah Mohd Sau Pi, Mazlina Abdul Majid [1] identified six components of the Smart Chatbot Academic Model through an extensive literature survey from the years 2017-2020. They compared existing chatbot applications for university websites and identified their purpose, type, character, and the programming languages behind them. However as this was a research work, they were unable to validate the identified components and hence could not proceed with the designing and development of the proposed product.

Authors Yurio Windiatmoo, Ridho Rahmadi, Ahmad Fathan Hidayatullah [2] implemented a chatbot based on deep learning which could be integrated with Facebook Messenger to answer university-related queries. The evaluation results of the model gave nearly perfect scores of precision, recall, and F1 with fast response time. However, the chatbot had not been used operationally on campus and thus its effectiveness and ease of use for users could not be measured yet.

Authors Olusegun Lala, Temilola Okedigba, Halleluyah Oluwatobi Aworinde [3] implemented an admission enquiry chatbot using IBM Watson for rapid response to admission related queries. When evaluated with Botium, the chatbot gave an accuracy of 95.9% with optimal and real-time feedback. While the model was successful in most aspects, it could only answer text-based queries with no support for voice input.

Authors S. Kumari, Z. Naikwadi, A. Akole, P. Darshankar [4] implemented a voice and text-based chatbot which could answer admission-related queries. In addition to the previously implemented works, this chatbot allowed the users to express their satisfaction with the provided answers by pressing the like or dislike buttons. This data was stored at the backend which served as a guideline for the Administrator to improve the answers framed. However, it could not understand the user's query if there were any glitches in input due to human spoken language, like a grammar error or a context error.

Authors Koundinya Hrushikesh, Ajay Krishna Palakurthi, Vaishnavi Putnala, Ashok Kumar [5] implemented an online chatbot system for visitors to the college website based on the AIML language which is a type of XML that enables the

user to get academic information. The chatbot utilized WordNet calculation and grammatical form labeling to distinguish the feeling of the words. The main limitation of WordNet is that it does not

provide a clear distinction criterion between atomic and non-atomic lexical units due towhich the chatbot is unable to recognize more words.

Authors Neelkumar P. Patel, Devangi R. Parikh, Darshan A. Patel, Ronak R. Patel [6] developed an interactive university chatbot with a GUI similar to a conventional messaging application that could answer text-based queries with minimal response time and very few database hits. On the downside the chatbot worked well only if the user framed the query using predefined keywords; it provided default answers when synonyms of keywords were used.

Authors Kulkarni, Pradnya, Ameya Mahabaleshwarkar, Mrunalini Kulkarni, Nachiket Sirsikar and Kunal Gadgil [7] threw light on the latest research in the field of Conversational AI along with the improvements achieved over the traditional counterparts. They explored the three main components of Conversational AI along with their accuracy, methodologies, and drawbacks.

Authors Ralston, Kennedy, Yuhao Chen, Haruna Isah and Farhana Zulkernine [8] developed a voice interactive and multilingual chatbot that could effectively respond to the users' mood, tone, and language using IBM Watson Assistant for responding to users' needs regarding exam stress. While it was a novel approach to the existing works, it could provideonly about 76.5% accuracy.

Authors Xiong, Wayne, Lingfeng Wu, Fil Alleva, Jasha Droppo, Xuedong Huang, and Andreas Stolcke [9] enhanced their conversational speech recognition system based on Microsoft for Switchboard and CallHome domains by adding the CNN-BLSTM system. The resulting system had reduced error rates than its previous iterations.

The following table contains the comparison of variousexisting chatbot frameworks that were considered while designing the proposed chatbot.

A survey conducted by Braun, Daniel, Adrian Hernandez Mendez, Florian Matthes, and Manfred Langen [10] comparesthe NLU capabilities of the above frameworks. As chart-1 shows, Rasa ranks second overall and outperforms Watson and DialogFlow. This supported our decision to use Rasa as the framework behind our proposed chatbot.

Table -1: Comparison of Existing Chatbot Frameworks

Framework	Hosting Model	Pricing	Languages Supported
DialogFlow	Cloud	Free with optional enterprise plan	Vary by channel
Amazon Lex	Cloud	Pay per use	English
IBM Watson	Cloud	Varies. 10000 free transactions per month	English, Japanese
Microsoft LUIS	Cloud	Basic: Up to 1000 transactions per second; $0.75 per 1000	Vary for prebuilt entities and prebuilt domains

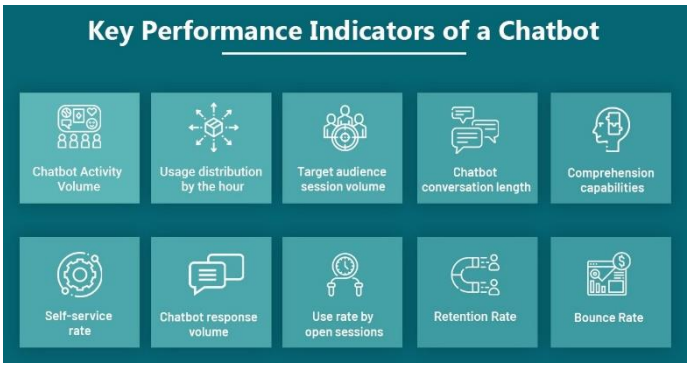


Chart -1: Performance Comparison of Chatbot

3. PROPOSED SYSTEM

The proposed methodology makes use of both qualitative and quantitative perspectives, and includes a broad array of approaches such as literature reviews, expert opinions, focus groups, and content validation. Students can enquire about facilities and query related to exams, academics, fee structure, etc.Students can also ask questions related to placement activities.

3.1 Project Scope

The chatbot is an AI-based chatbot that receives questions from users in audio or textual format, converts the audio intotext format, tries to understand the question by processingthe text using NLP, and finds an appropriate answer to the question. In natural language processing, human language isdivided into several pieces so that the grammatical structure of statements and the meaning of those pieces can be analyzed and understood in context to the whole conversation. This lets computers read and understand spoken or written text in the same way as humans. For example, when the chatbot receives the question "How manydepartments are there in the college?" it will answer "The college has 6 departments". The main objective is to reducethe burden on the college faculties by deferring the responsibility of answering the visitors' doubts regarding the college to the chatbot by creating a web-based chatbot that can be incorporated with the college website and can answer the users' text as well as audio-based queries. The goal is to provide visitors and faculty a quick and easy way tohave their doubts answered as well as offer the developers the means to incorporate new information in the chatbot's information repository.

3.2 User Classes and Characteristics

The two classes of users for this application based on the mode in which they query the chatbot are as follows:

1. Text- These users provide input in textual format by typing in the text box.
2. Audio- These users provide input in audio format which will first be converted into a textual format or the chatbot server to process.

3.3 System Architecture

Rasa is an open-source machine learning framework toautomate text-and voice-based conversations. It should provide us with all the functionality that we might need forimplementing the project. Rasa helps us build contextual assistants capable of having layered conversations with lotsof back-and-forths. Python's pyttsx3 library should be sufficient for converting audio into text.

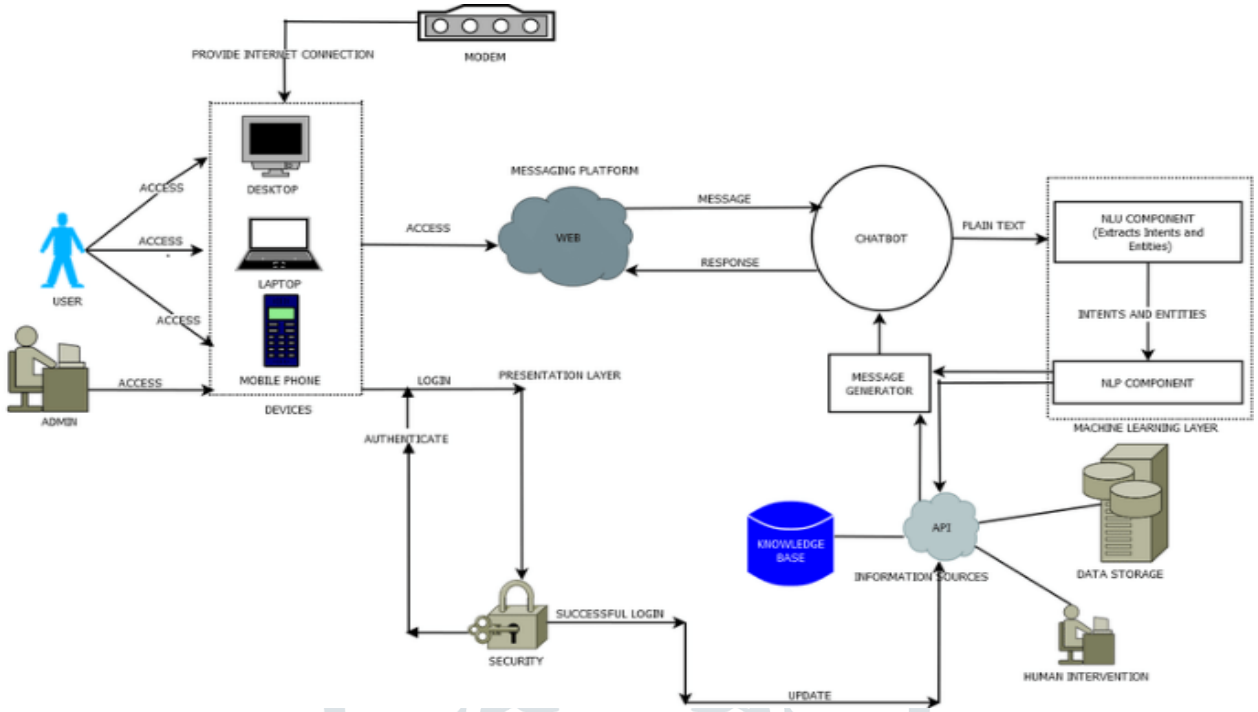


Fig -1: System Architecture

3.4 Functional Requirements

1. Users should be able to enter their query textually or inaudio format.
2. The chatbot should be able to process that query.
3. It should fetch an appropriate response.
4. It should be able to relay that response to the user.
5. In case the query is out of scope, provide a default answer.

3.4 Database Requirements

1. NLU Dataset with more than 100 examples for intent.
2. NLU Dataset with more than 100 examples for the entity.
3. Stories dataset with staring stories.
4. One dataset specifying Rules for the chatbot.

3.5 Software Requirements (Platform Choice)

1. Advanced Natural Language Processing. There are two options in the chatbot space: Click or AI.
2. Multilingual AI.
3. Easy Channel Integration.
4. Easy Backend Integration.
5. Enterprise-Grade Security.
6. Sentiment Analysis.
7. Hybrid Chat.

4. LIMITATIONS

The proposed chatbot does not support regional languages.It would be able to process and answer questions in Englishonly as it is the only language supported by Rasa NLU. spaCyis one of the default pipelines for processing user inputs in Rasa. While it works well when server capacity is low,demand for a higher configuration server to host the chatbotapplication might be a challenge.

5. CONCLUSION AND FUTURE WORKS

This research study proposes a college enquiry chatbot thatcan be integrated with a college website to interact with visitors and answer their questions about the college. The proposed chatbot will be able to accept both text-based as well as audio-based input from users. We have conducted acomprehensive literature survey to find the most appropriate chatbot framework and have referred to existingworks that could provide an insight into the domains of NLP and Conversational AI.

This research study is focused on identifying the requirements for the development of a dynamic chatbot thatsupports text as well as voice-based interaction with the users.

REFERENCES

[1] Pi, S.N.M.S. and Majid, M.A., 2020, December. "Components of Smart Chatbot Academic Model for a University Website". In 2020 Emerging Technology in Computing, Communication, and Electronics (ETCCE) (pp. 1-6). IEEE.

[2] Windiatmoko, Y., Rahmadi, R. and Hidayatullah, A.F., 2021, February. "Developing Facebook Chatbot Based on Deep Learning Using RASA Framework for UniversityEnquiries". In IOP Conference Series: Materials Scienceand Engineering (Vol. 1077, No. 1, p. 012060). IOPPublishing.

[3] Gbenga, O., Okedigba, T. and Oluwatobi, H., 2020. "An Improved Rapid Response Model for University Admission Enquiry System Using Chatbot". Int. J. Comput., 38(1), pp.123-131.

[4] Kumari, S., Naikwadi, Z., Akole, A. and Darshankar, P., 2020, July. "Enhancing College Chat Bot Assistant with the Help of Richer Human-Computer Interaction and Speech Recognition". In 2020 International Conferenceon Electronics and Sustainable Communication Systems (ICESC) (pp. 427-433). IEEE.

[5] Koundinya, H., Palakurthi, A.K., Putnala, V. and Kumar, A., 2020, July. "Smart College Chatbot using ML and Python". In 2020 International Conference on System, Computation, Automation, and Networking (ICSCAN) (pp. 1-5). IEEE.

[6] Patel, N.P., Parikh, D.R., Patel, D.A. and Patel, R.R., 2019,June. "AI and web-based human-like interactive university chatbot (UNIBOT)". In 2019 3rd InternationalConference on Electronics, Communication and Aerospace Technology (ICECA) (pp. 148-150). IEEE.

[7] Kulkarni, P., Mahabaleshwarkar, A., Kulkarni, M., Sirsikar, N. and Gadgil, K., 2019, September. "Conversational AI: An Overview of Methodologies, Applications & Future Scope". In 2019 5th International Conference On Computing, Communication, Control And Automation (ICCUBEA) (pp. 1-7). IEEE.

[8] Ralston, K., Chen, Y., Isah, H. and Zulkernine, F., 2019, December. "A voice interactive multilingual student support system using IBM Watson". In 2019 18th IEEEInternational Conference On Machine Learning And Applications (ICMLA) (pp. 1924-1929). IEEE.

[9] Xiong, W., Wu, L., Alleva, F., Droppo, J., Huang, X. and Stolcke, A., 2018, April. "The Microsoft 2017 conversational speech recognition system". In 2018 IEEE international conference on acoustics, speech andsignal processing (ICASSP) (pp. 5934-5938). IEEE.

[10]Braun, D., Mendez, A.H., Matthes, F. and Langen, M., 2017, August. "Evaluating natural language understanding services for conversational question answering systems". In Proceedings of the 18th AnnualSIGdial Meeting on Discourse and Dialogue (pp. 174-185).

Available online @ https://jjem.jnnce.ac.in
https:www.doi.org/10.37314/JJEM.2021.050204
Indexed in International Scientific Indexing (ISI)
Impact factor: 1.395 for 2021-22
Published on: 31January 2022

Institute Enquiry using Chatbot

Nuzath Khanum, Vedananda D E

Department of Computer Science and Engineering,
JNN College of Engineering, Shivamogga, Karnataka State, India,

nuzathkhanum2014@gmail.com, vedanandade@jnnce.ac.in

Abstract

Students almost always need to visit colleges or universities to obtain various data, such as information about the college, tuition costs, term schedules, and so on, during their confirmation interactions or for their daily needs. This process is arduous and time-consuming, and it also needs people to provide required data to visitors that come to the college. As a result, a chatbot can be constructed to solve the problem. The project is about user participation with the chatbot, which can be accessed from anywhere and at any time. With only a few fundamental language modifications, the chatbot may be easily/conveniently integrated with any college or university website. Chatbot provides various statistics associated with college or university, as well as additional student-related data. Any individual who has access to the college's website can use the chatbot. Artificial Intelligence and Machine Learning are also used in the venture. The advancement of Chatbots is aided by Natural Language Processing. Users can also ask college-related questions, which are subsequently applied as a contribution to the calculation, which assesses the message and displays the users' comparing reaction.

Keywords: *Chatbot, NLTK, Artificial Intelligence, Natural Language Processing, Machine Learning, HTML, Json.*

1. Introduction

A chatbot likewise alluded to as bot is a pc framework which speak by means of hear-able or perhaps text based strategies. "ChatterBot" was at first instituted by Michael Mauldin (maker of the absolute first Verbot, Julia) in 1994 to clarify these conversational applications [nine]. Chatbots are not difficult to create just as advance users collaborations. Mobile applications need being downloaded just as utilize extra space yet chatbots aren't important to be downloaded. It's anything but a message in an informing application. Bots can discover from users conduct and furthermore give significantly more customized answers. It's assessed that approximately 80% of organizations are planning to fuse chatbots by the whole year 2020. Students incline toward their issues settled rapidly, obligations to be refined promptly and information to be found quickly. The chatbot can give them a human like discussion and address the issues of the users. A few instances of chatbots are GupShup, RechargeBot, IxiBaba, Lawbot and so forth.

2. System Architecture

In this project the architecture is about the chatbot working, how does it works which is shown in Figure 1. The web application uses the python flask with the chatbot engine, as the chatbot engine contains the NLTK packages and the string processing which helps chatbot to responds to the user.
The chatbot also contains training data which is stored in .Json where the admin inserts all the queries which needs to be answer by the chatbot according to users. And also the admin part contains the Log where if the queries are not inserted in the training data and it will be

taken as the feedback from the users if not been answered on time by chatbot and that to can respond to the user next time.

According to the architecture the chatbot responds to the user with the help of the training data which is present in .Json file and also the feedback file which Log file to help the users accordingly and easy to build the queries to the admin. Chatbot consist of training data as .Json, HTML, CSS, flask as web interface and NLTK package as string processing.

The architecture also shows the chatbot responds to user to which query should it respond to .Chatbot depends and works on the training data which has been given and done by the admin. Chatbot engine includes all the data and files which is relatable to the chatbot.

The log file also shows and record the positive response which is chatbot has been answered to the queries asked by the user which is named as ’positive’ and ’response’ that is chatbot responded accurately. And on other hand the feedback queries will be recorded and shown to admin to frame the queries for the next time as the user cannot miss that next time, so which is named as ’negative’ in the Log file.

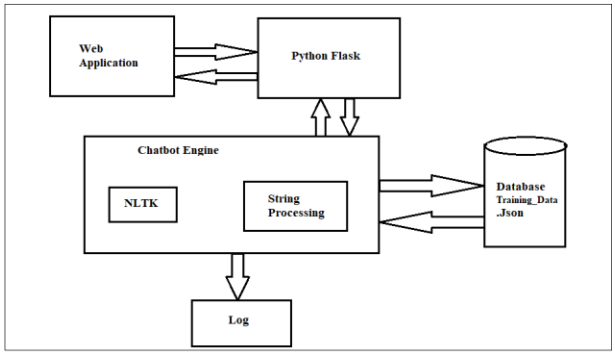


Figure 1: System Architecture

At last the chatbot answers all the queries which are present in the database which is present in .Json file and if not present taken as feedback and will answer it next time. There is no disadvantage that the chatbot do not respond and will not answer to the other question apart from the database. So this above architecture shows how each and every part works to complete this institute enquiry chatbot.

3. Methodology

The flowchart in Figure 2 shows that how the chatbot works.

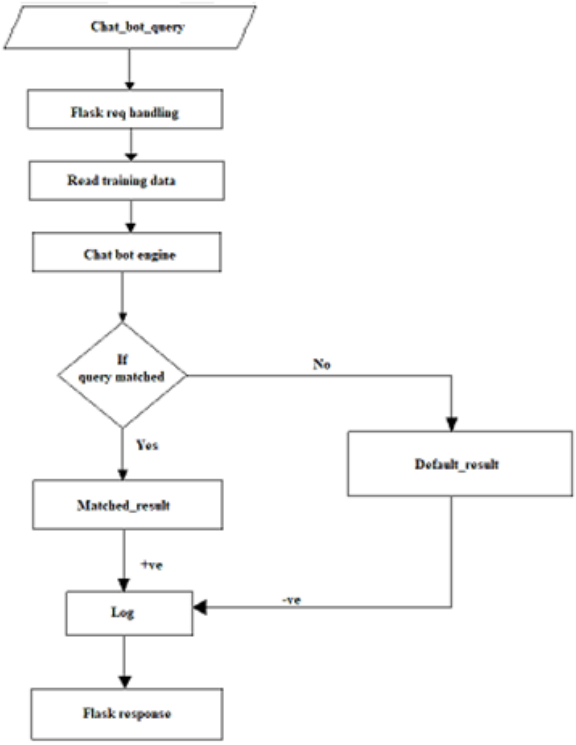


Figure 2: Flow Chart

The chatbot query will be taken as the input from the user to the chatbot, then the flask handling will taken place to the query according users after the flask request will be handled then the training data which is made up of .Json file will be read and the chatbot engine will be responded accordingly.

If yes then the query gets matched then matched result will be shown to the user by the chatbot and if not the query is matched to the user’s query then default result what has been given to the chatbot will be reflected at that time and the queries will be taken as the feedback which is stored in the Log file.

Then the Log file will have both the response which has been asked by the user and answered by the chatbot, and even the queries are not present in the training data can be taken

as the feedback so that the admin can insert those queries to be answered accordingly by the chatbot to the user. And finally, the python flask which acts as the web interface will reflect the answer by the chatbot which has been asked by the users.

In this Figure 3 is about log plotter where the graph shows that the positive response of the value 1 and the negative is of zero. The user asks the query and the chatbot responds to it correctly if the query is in the training database which has been stored in .Json file, so the response will be recorded in a log file as positive and the value shown in the graph is 1 like how many accurate answer has been responded by the chatbot to the user's query. If the training database doesn't contains the query which been asked by the user, then the chatbot responds the default answer that is "Sorry, i didn't get you" and these queries which has not been answered correctly will be taken as feedback and stored in a Log.txt file which will be shown as the Negative response. Hence it will help admin to add/insert the queries which has not been answered by the chatbot in the training data which is in .json File.

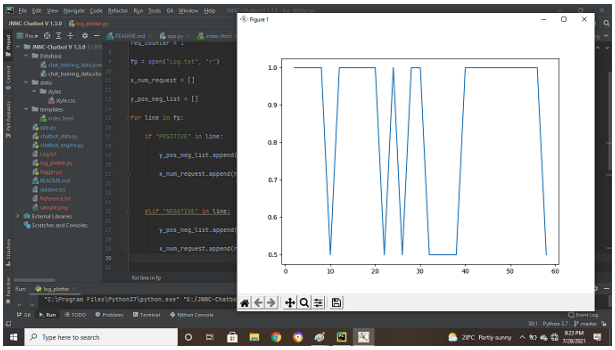


Figure 3: Log Plotter

The x and y axis are present in the graph, where the x-axis represent the num_request that is nothing but the number of request has been recorded in the Log.txt file where the chatbot has been responded to the user which correct or not correct that means positive response or negative response. In the other hand y-axis represent the pos_neg_list that is nothing but the positive negative list which has been shown in the graph that the list has like the negative is shown as 0 and the positive as 1.

4. Results and Discussion

The website snapshot is shown in Figure 4, where we can see a clear picture or snapshot of the website, as well as the blue chatbot icon on the website, which is located on the right bottom of the website page. The combination of website and chatbot is shown in this screenshot.

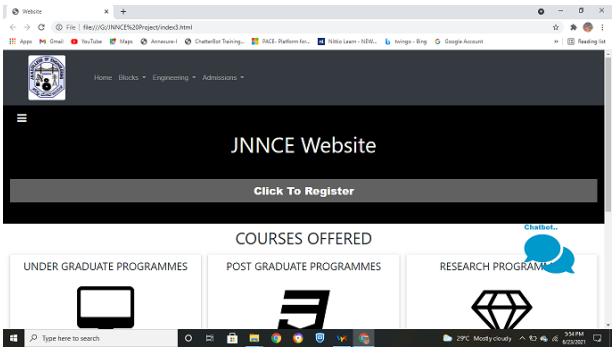


Figure 4: Website snapshot 1

In Figure 5, you can see how the chatbot actually looks. "Hello, welcome to JNNCE!" the chatbot will greet users at first. Send me a message." After that, the user can begin to ask questions about college to which the student or parent is interested.

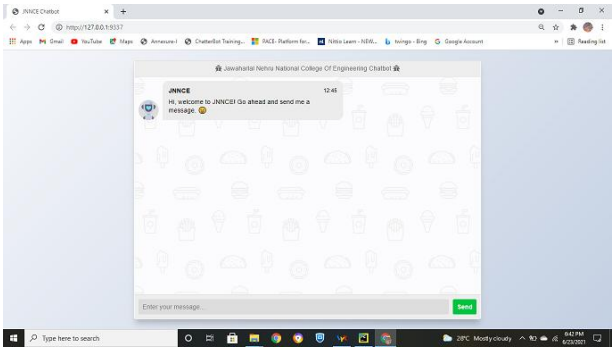


Figure 5: Chatbot Snapshot 1

Almost all users, such as students and parents, want to know where the college is located, what it looks like, and how far it is from the

users' location. The location in the snapshot is nothing more than the Url, which can be copied and entered into the user's map to locate the college. The user can access and take it from the chatbot part by using the location Urls in this chatbot. The screenshot above indicates that the chatbot can also function with Urls, as illustrated in Figure 6.

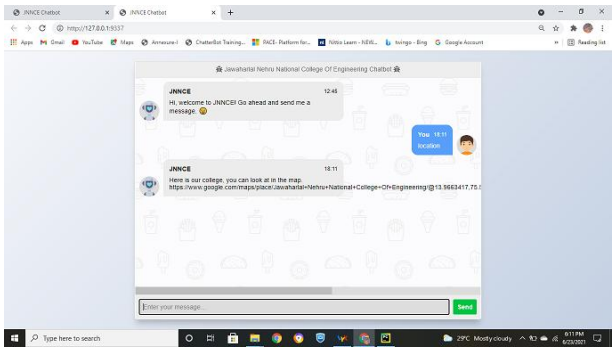


Figure 6: Chatbot Snapshot 2

The screenshot in Figure 7 shows how the chatbot responds to users for any query, with a focus on how string matching is accomplished. If a user types 'hi,' the chatbot will respond with 'Hello,' for example. ‘How may I assist you?' 'Hiiiiiiiiiiiiiiii’ many users will use and become addicted to the fact that they may use as many "i's" in the word "hi" in order for the chatbot to react by matching the string and responding with the same response as 'Hello, How may I assist you'.

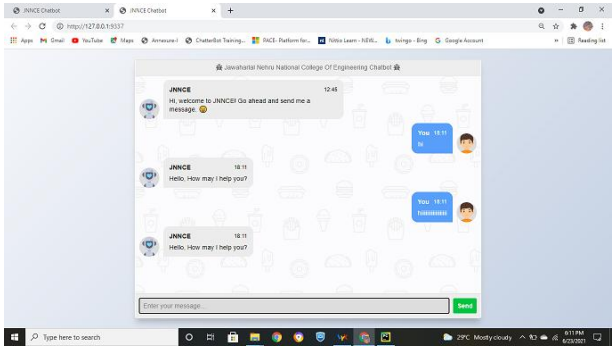


Figure 7: Chatbot Snapshot 3

The image in Figure 8 is of a log plotter, where the graph indicates a positive reaction of 1 and a negative response of 0. If the query is in the training database, which has been stored in, the user asks the question, and the chatbot responds appropriately. file json, As a result, the response will be logged as positive in a log file, and the value displayed in the graph will be 1, indicating how many accurate answers the chatbot has provided in response to the user's enquiry.

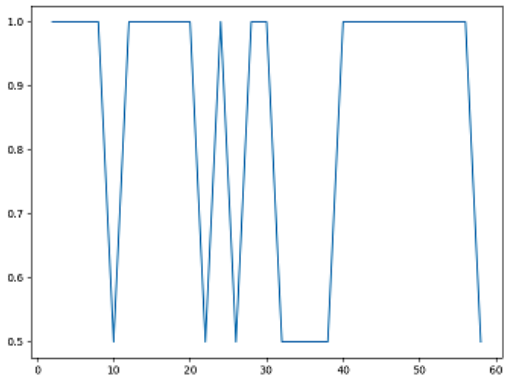


Figure 8: Snapshot Log Plotter

5. Conclusion

The users can collaborate with speedy approach by furnishing with the best instrument which is called as chatbot. Helpful to the users as it permits to enter enquires in normal language and wanted data is acquired effectively to the users. In this project, insights concerning plan, calculation utilized and execution of the chatbot is introduced. The users doesn’t have to assemble data by visiting college or universities, users can get to it straight forwardly from these chatbot anyplace or whenever that is 24×7.

6. Future Scope

The extent of the chatbot can be expanded by embeddings information for every one of the offices, preparing the bot with fluctuated information, testing it on live site and dependent on that input embeddings seriously preparing information to the bot.

A portion of the new highlights which can be added to the bot are:

1. Discourse acknowledgment include through which student can ask their

questions verbally and find the solutions from the bot.

2. Joining with numerous channels, for example, call, SMS, and different online media stages like Skype, Face book and Twitter.

References

1. K.Bala, Mukesh Kumar, Sayali Hulawale, Sahil Pandita, Chat-Bot for College Management System Using A.I, International Research Journal of Engineering and Technology (IRJET), Vol. 04, Issue 11, Nov 2017, pp. 2030-2033.

2. Chun Ho Chan, Ho Lam Lee, Wing Kwan Lo, Andrew Kwok-Fai Lui, Developing a Chatbot for College Student Programme Advisement, Proceedings of the International Symposium on Educational Technology, Osaka, Japan, 31st July -2nd Aug 2017, doi 10.1109/ISET.2018.00021

3.P.Nikhila, G.Jyothi, K.Mounika, C Kishor Kumar Reddy and B V Ramana Murthy, Chatbots Using Artificial Intelligence, International Journal of Research, Vol. 8, Issue I, January 2019, pp.1-12.

4. Nithuna S and Laseena C.A, Review on Implementation Techniques of Chatbot, Proceedings of the International Conference on Communication and Signal Processing, Chennai, India, July 28 - 30, 2020.
doi 10.1109/ICCSP48568.2020.9182168

5.Ram Manoj Sharma, Chatbot based College Information System, Research Review International Journal of Multidisciplinary, Vol. 04, Issue 3, March-2019, pp. 109-112.

6.K. V. Sreya, P. Kavya, S. Gokulakrishnan, Student Chatbot for University Administration, Aegaeum Journal, Vol.8, Issue 5, 2020, pp. 184-190.

7.Neelkumar P. Patel, Devangi R. Parikh, Darshan A. Patel, Ronak R. Patel, AI and Web-Based Human-Like Interactive University Chatbot (UNIBOT), Proceedings of the 3rd International Conference on Electronics Communication and Aerospace Technology [ICECA 2019], Coimbatore, India, 12-14 June 2019.
doi: 10.1109/ICECA.2019.8822176

8.Punith S , Chaitra B , Veeranna Kotagi, Chethana R M, Research Paper on Chatbot for Student Admission Enquiry, Journal of Advancement in Software Engineering and Testing, Vol. 3 Issue 1, 2020, pp.1-9.

9.Avni Prajapati, Pranali Naik, Sushmita Singh, Anil Kale, Android Based Chatbot For College, International Journal of Scientific & Engineering Research, Vol. 9, Issue 4, April-2018, pp.10-12.

10. Kumar Shivam, Khan Saud, Manav Sharma, Saurav Vashishth, Sheetal Patil, Chatbot for College Website, International Journal of Computing and Technology, Vol. 5, Issue 6, June 2018, pp.74-77.

Design of Chatbot System for College Website

Article *in* International Journal of Computer Sciences and Engineering · July 2021

DOI: 10.26438/ijcse/v9i7.4145

5 authors, including:

Pushkar Shinde
NDMVPS's KBT COE , Nasik
7 PUBLICATIONS **4** CITATIONS
SEE PROFILE

Sopan Talekar
Nashik District Maratha Vidya Prasarak Samaj's K.B.T. College of Engineering
16 PUBLICATIONS **24** CITATIONS
SEE PROFILE

International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol. 9, Issue.7, July 2021 E-ISSN: 2347-2693

Design of Chatbot System for College Website

Shivani Pravin Rashinkar[1*], Neha Dilip Wanjol[2], Shivani Nandkumar Rane[3], Pushkar P. Shinde[4], Sopan A. Talekar[5]

[1,2,3,4,5]Department of Computer Engineering, MVP's KBT College of Engineering, Pune University, Nashik, India

**Corresponding Author: rashinkarshivani173@gmail.com, Phone no.:+918805871670*

DOI: https://doi.org/10.26438/ijcse/v9i7.4145 | Available online at: www.ijcseonline.org

Received: 14/Jul/2021, Accepted: 20/Jul/2021, Published: 31/Jul/2021

***Abstract*—** Most of the time, Students need to visit college administration office to collect various information regarding college such as Tuition fees, Term Schedule, etc. during admission process or as per their daily needs. Hence, to overcome this problem, a chatbot can be designed and developed which can be easily integrated with any college website to provide necessary information regarding college. The goal of AI based chatbot is to make an efficient conversation between human and machine via auditory or textual methods. This project uses Natural language processing to process the user's query and generate a meaningful response. Based on the information stored in the database, bot itself determines appropriate response of a particular query fired by user. The Chatbot is based on an Artificial Intelligence algorithm, which analyses user's question and responds with a Naive Bayes' algorithm. This system will be a Web Application and can reduces work of college administration providing information to students. It also reduces the workload on the staff to answer all the queries of the students.

***Keywords*—** Chatbot, Query, Graphical User Interface, Natural Language Processing, Artificial Intelligence.

I. INTRODUCTION

Chatbot is a computer application that mimics human conversations in its natural format along with textual or voice communication. In the utilization of AI techniques together with natural language processing (NLP) [1],[3] chatbot for college website can be designed. This system will be a web application, so it can give solutions to the analysed queries of the user. User simply need to raise the query to the chatbot and the system will reply to the users through the powerful Graphical User Interface (GUI) which is similar to messaging application interface, and provides a friendly environment to the user as they are much aware of operating messaging application[1],[2]. The user can submit the question about the college-related information or activities such as admission process, contact information, address, annual day, sports day, intake and other cultural activities of colleges. Developing a chatbot solves the problems that can arises in gathering required college information. This system can be accessible from anywhere and anytime. Chatbot will deliver efficient and relevant response to the user corresponding to their entered message. Chatbot system will be beneficial for students, parents, teaching and non-teaching staff as well. Presently, there are various chatbots available for the students like UNIBOT, ALICE etc. UNIBOT is designed for the students to ask university related question. For this system a new algorithm is developed to deliver an appropriate response to the user corresponding to their entered message[2]. ALICE is a rule-based chatbot based on the Artificial Intelligence Markup Language (AIML). This System uses NLP and Pattern Matching Algorithm [3],[6],[7] to process user's query[8].

This paper is divided into several sections, where Section I contains the introduction of chatbot system, Section II contains Related Work of chatbot system, Section III explains the Methodology with architecture diagram and flow chart, Section IV contains Results and Section V describes Conclusion and Future Work.

II. RELATED WORK

K. Bala, M. Kumar, S. Hulawale, and S. Pandit et al. [1] Project on Chatbot for college management is developed with the help of AI algorithms which can analyse user's queries. This is a web application that will give answers to the analysed queries of the user. Users will simply need to select the class for queries and ask the question to the bot. In this paper, they have used Porter Stemmer algorithm to answer the user' queries. The Users should register and login to the system. Once login, user can access the various helping pages through which the user can ask queries related to college activities.

P. Nikhila, G. Jyothi, K. Mounika, Mr.K. K. Reddy et al [2] The chatbot named UNIBOT is designed for the students to ask university related question. This system uses the concept of Artificial Intelligence and Machine Learning. The System uses PHP Language for the development of UNIBOT. The query is given as an input to the algorithm, which processes it and gives the corresponding response to the user. The GUI is similar to a Messaging Application. It delivers efficient and relevant response to the user corresponding to their entered query. New algorithm is developed for UNIBOT. It is very efficient, requires less memory and has minimal database hits.

B.Setiaji and F. W. Wibowo et al. [3] Chatterbot is designed with a powerful pattern matching algorithm. This project uses Indonesian conversational pattern and MySQL database. This application is based on a knowledge base which is maintained by admin. It can be miss in defining a sentence and how to response it while connecting chat application to the database. In the pattern-matching operation, knowledge representation and implementation of SQL are important. A data that has been created which is based on the pattern of the conversation must be tested by the help of a series of scenarios. The conversation should be crosschecked to the basic pattern so that it allows you to add some knowledge to the database which is not added before. If the input record in the database does not match, then it will be remodeled.

K. Shivam, K. Saud, M. Sharma, S. Vashishth, and S. Patil et al. [4] In this paper, for designing chatbot, Facebook Messenger is used which is source and uses artificial intelligence to communicate with the user and provide the required information. This Facebook API is integrated with Python backend, webhook is used to deliver the query of the user to the server. This system has used WIT and AI as a pre-trained artificial intelligence module so that one could use its pre-trained neural networks to answer the user's query with efficiency and accuracy.

E. Haller and T. Rebedea et al. [5] This paper describes the concept of identifying vital facts in texts describing the life of a historical figure for building a conversational agent that could be used in middle-school CSCL scenario. This paper presents a way for building a chat-bot that may simulate a historical figure. The system can receive "input" as an understandable text or a web page about the historical figure and has as "output" a trained conversational agent that is in a position to answer all reasonably questions about the life expertise of that user.

S. B. Sonawane, A. S.Badwar, R. H. Dalvi, G. N. More and S. A. Talekar et al. [6] This paper describes the concept of AI based chatbot which is designed for Student Counselling for career guidance. This system uses NLP and Keyword Matching Algorithm to process user's query. This System uses a modular architecture to respond to user input. Each module contains knowledge based initialization mechanism, and logic to handle user requests. The students are able to get proper guidance for career in the field of their choice, also the college list for the same is provided as per requirement.

III. METHODOLOGY

The proposed chatbot system is a web application which gives reply to the question of the user. This system is utilized for talking. A chatbot project is built using artificial algorithms i.e. Naive Bayes' algorithm that analyse user requests and understand the user's message[9]. The system uses Natural Language Processing (NLP) and built-in artificial intelligence to answer the queries asked by the user. For the chatbot development, we have used Python programming language using Django framework and Chatterbot library. It makes it easy to generate automated responses to a user's input with the help of a machine learning algorithm to produce different types of responses. Students just have to query through the bot which is used for chatting purpose. Chatbot will reply to the query with the help of artificial intelligence.

The proposed system will reduce the administration burden and will be able to provide necessary details to students and parents online. Students do not have to visit college administration for every inquiry. Students will get their queries resolved without any hassle to reach out the college administration office. The System will be available for 24/7 to all students and parents [4].

1. User Login:

User just needs to submit his/her queries regarding the college to the bot. User can be student, teacher as well as parents.

2. Chatbot Responding System:

2.1 NLP Processing for Query Analysis :

When a user submitted the query to the system, NLP is applied and sense of the query is detected. Natural Language Processing (NLP) allows chatbot to understand user's messages and respond appropriately.

2.2 Search Questions in database:

Once the query is submitted, furthermore, we have to search the exact answer of the query in our pre-trained database. As the query description can change from person to person. The same question can be asked in different ways from multiple users. One user asks a question so simply and clearly, while another user may ask the same question with different format. So it is necessary to find out what is the correct solution of a submitted query.

2.3 Answer the Queries:

As described above, whenever user submits a complaint, then it is checked that is there such question registered in database. If the answer is matched, then that is sent to that User. If the answer of submitted query is not found in the database, then such questions are answered by admin person. Once he answered the query, the answer is sent to that user. And also it is stored in database so that whenever such questions will be asked they get answer of that query directly from the database. So admin doesn't need to answer the same query again and again. For this we are using powerful Naive Bayes' Algorithm[9].

3. Architecture Diagram of Chabot System:

Figure 1: shows the architecture of the chatbot system for college website.

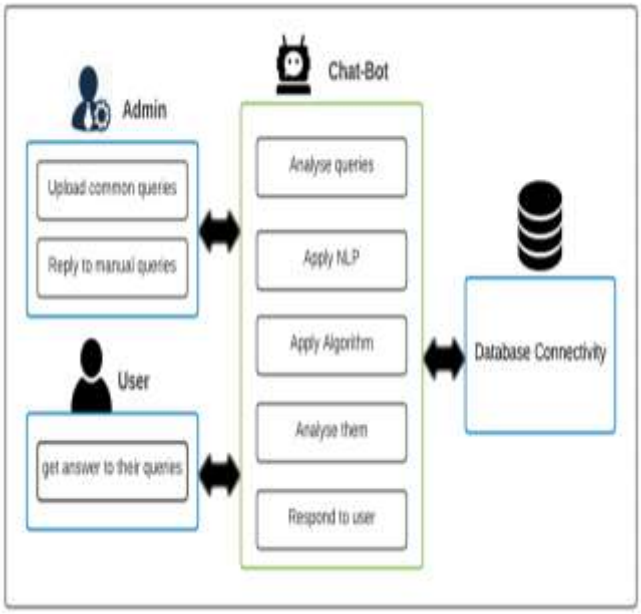


Figure 1: Architecture Diagram of Chatbot System

4. Flowchart of Chatbot System:

The flowchart of the system displays how the chatbot performs. Initially, the user message is pre-processed and connectivity to the database is obtained. Then, based upon conditions satisfied, the chatbot process flows and provide response to the user. If the user cannot find the answer for a query then in such condition chatbot will provide admin's contact details to the user. Admin can view and answer the corresponding query.

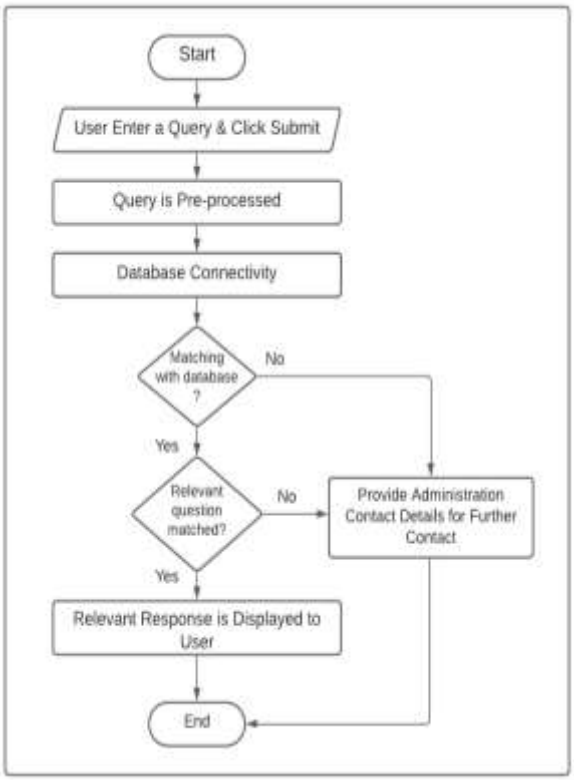


Figure 2: Flowchart of Chatbot System

5. Algorithm Implemented

Naive Bayes' is a powerful algorithm for text classification problems. It is a probabilistic machine learning algorithm which is based on Bayes' theorem[9]. This classifier assumes that the presence of a selected function in a category is unrelated to the presence of another function. In this algorithm a closed domain dataset containing questions/user-responses and corresponding answers is made, which every question is given a label, this will relate the question to its answer. Due to multiple questions could have the same response, there can be multiple questions having the same answer.

The Formula for Naive Bayes' Algorithm is as follows:

$$P(A \mid B) = \frac{(P(B \mid A) * P(A))}{P(B)} \quad (1)$$

Where,

P (A | B) = Probability of 'A' occurring given evidence of 'B' has already occurred.

P (B | A) = Probability of 'B' occurring given evidence of 'A' has already occurred.

P (A) = Probability of 'A' occurring.

P (B) = Probability of 'B' occurring.

6. Different Algorithms Used for Chatbot System:

Some of the most popular algorithms for chatbots are Porter Stemmer[1], Naive Bayes'[9], Support Vector Machines, K-means and atural language processing (NLP). Chatbots mainly use classification algorithms to recognize intent in phrases. Every algorithm has its own advantages and disadvantages according to its working method. Table 1 shows comparison of different algorithms WRT to Naive Bayes' algorithm.

Table 1: Comparison of Different Algorithms

Sr. No.	Porter-Stemmer Algorithm	K-Means Clustering Algorithm	Naive Bayes' Algorithm
1.	The stems generated are not always real words.	Different partitions can result into different final clusters.	Naive Bayes' classifier performs better than other models with less training data if the assumption of independence of features holds.
2.	It has five steps and sixty rules. And takes more time.	Difficult to predict K-Value, hence it is time-consuming.	The algorithm works very fast and can easily predict the class of a test dataset.

3.	This System is limited to English words only.	Doesn't support categorical data.	The algorithm performs well with categorical variables in comparison to numerical variables.

IV. RESULTS AND DISCUSSION

The Chatbot is to carry out a conversation between both human and machine. Figure 3: shows the simple and attractive pop-up Interface for College Chatbot System. Firstly, chatbot will print a welcome message then a user can type and submit the query and bot will provide an appropriate answer to the user's query. In figure 4 'What is college full name?' this question is asked by the user and correct answer is given by the chatbot. The User can query any college related activities through the chatbot system.

Figure 3: Chatbot Interface-1

Figure 4: Chatbot Interface-2

V. CONCLUSION AND FUTURE SCOPE

The aim of the system is to provide a user-friendly and efficient chatbot system for College. The chatbot will be very useful in guiding students to get correct and up-to-date information source. This system will be fruitful for students, teachers as well as parents. They can get information at any time without having to visit the college administration office every time. In the future scope of this project, the system will include voice-based requests and responses. Users are required to provide voice input and the system will output in the form of text. Chatbot will be able to provide a voice output as well, with the help of text to speech or speech to text conversion.

REFERENCES

[1] K. Bala ,M. Kumar, S.Hulawale, and S. Pandita, *"Chat-Bot For College Management System Using A.I"*, International Research Journal of Engineering and Technology (IRJET), Vol. **04**, Issue.**11**, pp.**2395-0072**, **2017**.

[2] P. Nikhila, G. Jyothi, K. Mounika, Mr.K. K. Reddy and Dr. B.V. Ramana Murthy, "*AI and Web-Based Human-Like Interactive University Chatbot (UNIBOT)",*In theProceedings of the Third International Conference on Electronics Communication and Aerospace Technology[ICECA 2019], pp.**1-12,2019**.

[3] B.Setiaji, F. W. Wibowo, "*Chatbot Using A Knowledge in Database- Human-to-Machine Conversation Modeling*", In the Proceedings of the 2016 International Conference on Intelligent Systems, Modelling and Simulation, pp.**2166-0670**, **2016**.

[4] K. Shivam, K. Saud, M. Sharma, S. Vashishth, and S. Patil, "*Chatbot for College Website*", International Journal of Computing and Technology, Vol.**5**, Issue.**6**, pp.**2348-6090, 2018**.

[5] E. Haller and T. Rebedea, "*Designing a Chat-bot that Simulates a Historical Figure*", Faculty of Automatic Control and Computers university of Bucharest, IEEE 978-0-7695-4980-4/13, **2013**.

[6] S. B. Sonawane, A. S.Badwar, R. H. Dalvi, G. N. More and S. A. Talekar, "*Design of Chatbot System for Student Counselling*", International Journal of New Innovations in Engineering and Technology, Vol.**13**, Issue.**3**, pp.**2319-6319**, **2020**.

[7] P. Jain,"*College Enquiry Chatbot Using Iterative Model*", International Journal of Scientific Engineering and Research (IJSER),Vol.**7**, Issue.**1**, pp.**2347-3878,2019**.

[8] N. Hatwar, A. Patil and D. Gondane, "*AI BASED CHATBOT*", International Journal of Emerging Trends in Engineering and Basic Sciences (IJEEBS) ISSN (Online) 2349-6967 Volume **3**, Issue **2** (March-April **2016).**

[9] A. Prajapati, P. Naik, S. Singh and A. Kale, "*Android Based Chatbot For College*",International Journal of Scientific & Engineering Research, Vol.**9**, Issue.**4**, pp.**2229-5518**, **2018**.

[10] A. Ohm, K. Bhavani,"*Chatbot for Career Guidance Using AI*", International Journal of Computer Sciences and Engineering (IJCSE), Vol.**7**, Issue.**6** ,pp.**2347-2693**, **2019**.

[11] K. G. Yohandi, M. I. Sani, "*Designing a Mobile Chatbot For Elementary School Vocabulary*", International Journal of Computer Sciences and Engineering(IJCSE), Vol.**8**, Issue.**7**, pp.**2347-2693**, **2020**.

AUTHORS PROFILE

Shivani Pravin Rashinkar, Bachelor of Engineering in Computer Science, MVP's KBT College of Engineering, Nashik, India

Neha Dilip Wanjol, Bachelor of Engineering in Computer Science, MVP's KBT College of Engineering, Nashik, India

Shivani Nandkumar Rane, Bachelor of Engineering in Computer Science, MVP's KBT College of Engineering, Nashik, India

Mr. Pushkar P. Shinde, pursed Bachelor of Engineering in Computer Science and Master of Engineering in Computer Science. Completed Diploma in Advance Computing, CDAC. He is currently working as Assistant Professor in Department of Computer Engineering, MVPS's KBTCOE, Nashik, India. He has published & presented more than 06 research papers. His research interest is Machine Learning. He has 16 years of teaching experience.

Mr. Sopan A. Talekar, pursued Bachelor of Engineering from North Maharashtra University Jalgaon, Maharashtra in 2004 and Master of Technology from Visvesvaraya Technological University, Belagavi, Karnataka in year 2010. He is currently pursuing Ph.D and currently working as Assistant Professor in Department of Computer Engineering, MVPS's KBTCOE, SPPU, Pune, since 2005. He is a life member of IE & ISTE. He has published & presented more than 07 research papers. He received prestigious Promising Engineer Award from the Institution of Engineers (India), Local center Nasik, in 2013. His research interest is Wireless Network, Mobile Ad hoc Network, Cognitive Radio Network & Artificial Intelligence. He has 16 years of teaching experience.

IMPLEMENTING A COLLEGE ENQUIRY CHATBOT

A Project

Presented to the faculty of the Department of Computer Science

California State University, Sacramento

Submitted in partial satisfaction of
the requirements for the degree of

MASTER OF SCIENCE

in

Computer Science

by

Ujaliben Kalpesh Bavishi

SPRING
2019

IMPLEMENTING A COLLEGE ENQUIRY CHATBOT

A Project

by

Ujaliben Kalpesh Bavishi

Approved by:

_____, Committee Chair
Dr. Pinar Muyan-Ozcelik

_____, Second Reader
Dr. V. Scott Gordon

_____
Date

Student: Ujaliben Kalpesh Bavishi

I certify that this student has met the requirements for format contained the University format manual, and that this project is suitable for shelving in the Library and credit is to be awarded for the project.

_____, Graduate Coordinator _____

Dr. Jinsong Ouyang Date

Department of Computer Science

Abstract
of
IMPLEMENTING A COLLEGE ENQUIRY CHATBOT
by
Ujaliben Kalpesh Bavishi

This project is focusing on creating a chatbot to be used by students to get their queries responded easily from the college website. The College Enquiry Chatbot has the capacity to make friendly conversations; respond the course and faculty details; give the link for the academic calendar; answer the frequently asked questions; calculate the fees based on the student's input; and give the timings, address, contacts, and events information of the departments like Union, Library, IPGE, and AIRC. To build the chatbot, Microsoft Azure bot service as well as Microsoft cognitive services, namely, Text Analytics, LUIS, and QnA Maker are used.

Most of the existing chatbots lack empathy and fail to accommodate anything outside of the script. In order to address these problems, the College Enquiry Chatbot extends the implementation of the current chatbots by adding sentiment analysis and active learning. Although, sentimental analysis correctly recognizes the user's query as positive, negative and neutral, the system was partially successful in adding empathy to the chatbot. It is because the system requires more rigorous training data to handle all queries which are off-script. However, for such queries, active learning helps to improve the chatbot

performance since it correctly understands the user's questions, asks clarifying question, and then retrains the system to give the response what the user intends to get.

The future work include training the chatbot with more varied data; increasing the scope of the chatbot by adding a speech recognition feature so that users can speak to get responses; and including integration with multiple channels such as phone call, SMS, and various social media platforms.

_____ Committee Chair
Dr. Pinar Muyan-Ozcelik

_____
Date

ACKNOWLEDGEMENTS

I would like to express my gratitude towards Dr. Pinar Muyan-Ozcelik who has been a precious help in my voyage of finishing this undertaking. Without her assistance, direction, and support, making this project a reality would have been unimaginable. Her opportune criticism was tremendously essential for me and helped me greatly to reach the targets set for the project tasks. Likewise, I would also like to express gratitude toward Dr. V. Scott Gordon for his commitment and supporting this project as my second reader. In conclusion, I would like to thank the personnel and staff of the Department of Computer Science Engineering at California State University, Sacramento, for their ceaseless regulatory help.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Figures Page

1 INTRODUCTION AND MOTIVATION

This project is focusing on creating a chatbot to be used by students to get their queries responded easily from the college website. A chatbot is a program which can do real conversations with textual and/or auditory methods [1]. Using Artificial Intelligence (AI), chatbots can simulate human conversations. There are two categories of chatbots. One category is command based chatbots where chatbots rely on a databank of replies and heuristics. The user must be very specific while asking the questions so that the bot can answer. Hence, these bots can answer limited set of questions and cannot perform function outside of the code. The other category is chatbots based on AI or machine learning algorithms, these bots can answer ambiguous questions which means the user do not have to be specific while asking questions. Thus, these bots create replies for the user's queries using Natural Language Processing (NLP).

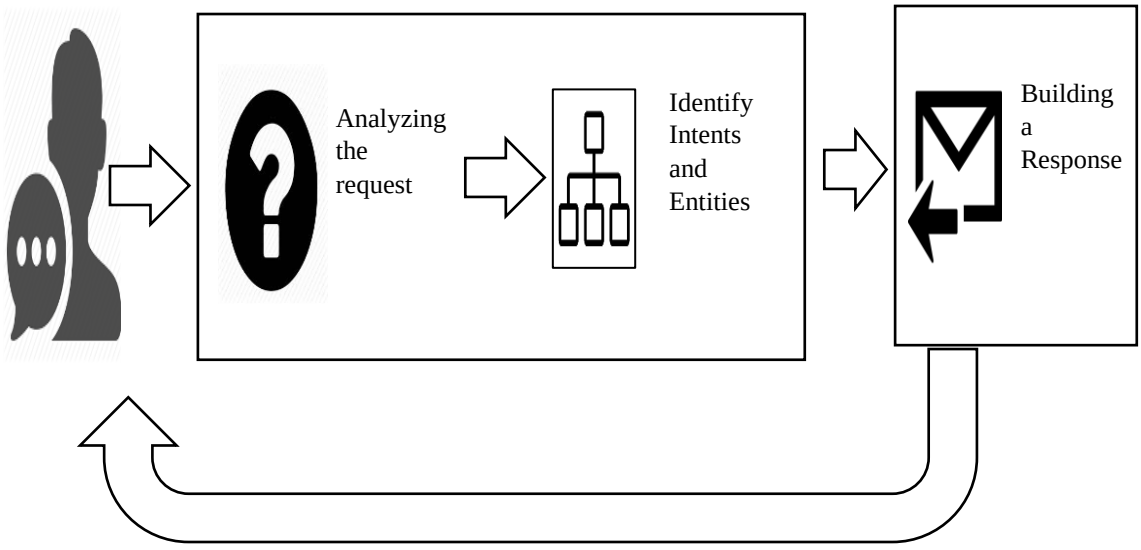


Figure 1: How a Chatbot Works [2]

Figure 1 shows how a chatbot works. Whenever a user asks any query, the bot will first analyze the request, then identifies intents and entities, builds a response and sends it back to the user. Now, intents mean intention of the query and entity means details of that query. For example, if a student wants to know the office hours of a faculty then the intent will be office hours and entity will be name of the faculty in this case.

AI-powered chatbots are motivated by the need of traditional websites to provide a chat facility where a bot is required to be able to chat with user and solve queries. When live agent can handle only two to three operations at a time, chatbots can operate without an upper limit which really scales up the operations. Also, if any school or business is receiving lots of queries, having a chatbot on a website takes off the load from support team. Having a chatbot clearly improves the response rate compared to human support team. In addition, since millennials prefer live chats over a phone call, they find a chatbot, which provide a highly interactive marketing platform, very attractive. Furthermore, a chatbot can automate the repetitive tasks. There can be some scenarios where a business or school receives same queries in a day for many times and support team must respond to each query repetitively. Lastly, the most important advantage of having a chatbot is that it is available 24/7. No matter what time it is, a user can get a query solved. All these advantages of a chatbot constitute the motivation to implement a College Enquiry Chatbot.

Before implementing College Enquiry Chatbot, various existing chatbots were reviewed such as Amazon Shopping App [3], Alexa [4], Bank of America (Erica bot) [5]

and CNN news bot [6]. In order to understand the requirement of a chatbot, consider an example of Amazon Shopping App. In this app, when a customer buys an item, he/she does not have any information about how to return the item. To get this information, the customer must call and wait to talk to customer representative for a long time. However, this whole process is tedious for a customer. Hence, Amazon created a chatbot to answer simple queries of customers.

Similarly, the College Enquiry Chatbot is designed to help students to get their queries solved on a fingertip. The most essential downside I found while utilizing the previously mentioned chatbots is absence of personality and conversational flow. As Storman [7] suggests, the CNN chatbot neglects to give sympathy and effortlessness. To be efficient, the chatbot must have the capacity to relate and associate itself with the user. For example, a discussion with the CNN chatbot is depicted in the article where at whatever point a user says anything aside from news or any current alternatives, it answers with the news comprising of those words and toward the end it says "Not sure I understand what you're looking for. Try again or pick one of the options below." and afterward it gives the choices as programmed. This implies, although it conveys the news proficiently, the CNN bot needs compassion.

Solution to this problem is described by Rahman et al. [8]. This study proposes that “there is a need to understand and consider the stability, scalability, and flexibility issues along with high level of intention of a human language”. Hence, for implementing a chatbot that handles complicated queries, the sentiment analysis is incorporated into College

Enquiry Chatbot. Sentiment analysis aims to obtain writer's feelings expressed in positive, negative or neutral comments. Based on sentiment analysis, the bot is trained to have empathy while answering to the user. For example, if a user says "I am sad today." then bot should reply to it with some empathy like "I'm sorry to hear that, how can I help you today?" and not just reply the standard message like "Sorry did not understand your question."

Another downside which was found during a research on chatbot is that bots are created in such way that they follow a specific route and mostly all of them fails to satisfy anything outside of the previously defined scripts. This means that if they are not part of a predefined scripts, a significant number of the bots will fail in understanding even the most fundamental kind of queries, which results in a repeating and horrendous experience. To resolve this issue, active learning can be introduced to the system to make probabilistic assessments and provide autonomous responses to the users [9]. Active learning is an algorithm which interactively queries user to obtain the desired output. Whenever a user asks anything which is outside of the script the chatbot will ask questions to the user by giving two to three options and based on the user's input, the bot returns the answer to that query. This whole learning process is called as active learning.

2 SYSTEM DESIGN

System design of College Enquiry Chatbot consists of integration of multiple technologies. This system makes use of NodeJS for backend and ReactJS for frontend. Microsoft Azure provides services like Cognitive Service, Bot Service [10], and Web Apps. Cognitive services consist of Text Analytics, LUIS [11], and QnA Maker [12]. The bot service is used to create the bot application on Azure. In addition, Web Apps are used to host the application on Microsoft Azure. System uses Mongo DB to store the conversations history.

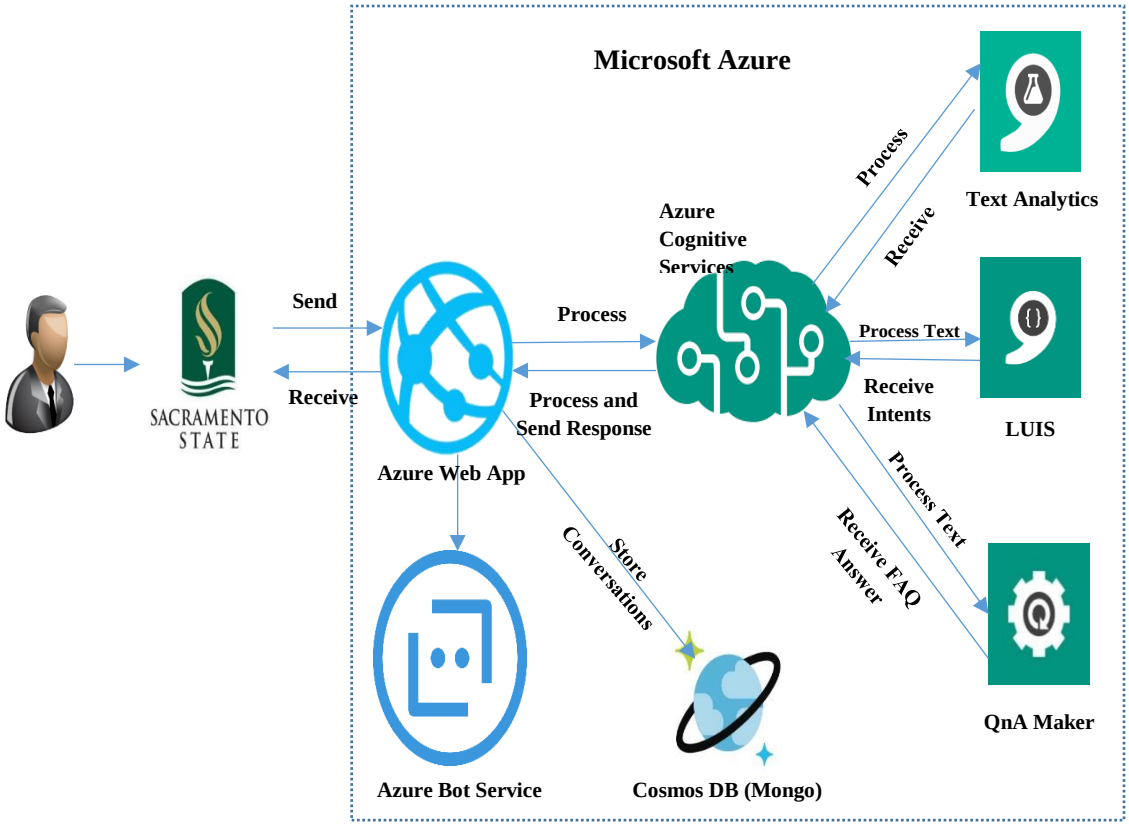


Figure 2: System Design

2.1 Microsoft Azure

Microsoft Azure provides cloud services to build, manage, and deploy applications on a cloud network which helps organization to meet their business needs using their favorite tools and frameworks. College Enquiry Chatbot uses all the services shown in Figure 2 which are provided by Microsoft Azure.

2.1.1 Azure Bot Service

Azure bot service is a service provided by Microsoft Azure which builds and artificial intelligence (AI) chatbot. It offers the ability to add intelligent agents that can do conversations with users without having to commit the resources to develop one's own AI [13].

2.1.2 Azure Web App

I have used Azure web app to host my bot application build by Azure bot service. Azure Web App are the web applications which are hosted on Microsoft Azure cloud without any programming language barrier. Since it is hosted on the cloud, infrastructure maintenance is not needed to host the applications. It also provides various features like auto scaling, automated deployment and also supports both Windows and Linux environments [14].

2.1.3 Azure Cognitive Services

Azure Cognitive services are used to add intelligence to the bots by adding features such as sentiment analysis and language understanding which help in analyzing user's queries [15]. These services can be added to an existing bot by adding service calls to the

Microsoft's SDK to get the desired results. College Enquiry Chatbot utilizes Text Analytics, LUIS, and QnA Maker services which are part of cognitive services and use natural language processing (NLP) for different purposes.

2.1.3.1 Text Analytics

Text Analytics is used to perform sentiment analysis on a user's queries. As shown in Figure 2, a text is sent to text analytics and it returns the sentiment of that text.

2.1.3.2 LUIS

"LUIS stands for Language Understanding Intelligent Services which aims at creating cloud-based machine learning language understanding (LU) models specific to an application and without machine learning expertise" [16]. To implement a College Enquiry Chatbot, I have first created all the possible intents and entities using LUIS tool. Based on these intents and entities, LUIS have built and train a LU deep learning model. Then, the created model needs to be pushed to the endpoint on the cloud. The queries from the user are passed to the endpoint to get the recognized intent and entities in JSON format. Based on various flows, response will be sent back to the user.

Active learning is also implemented using LUIS as shown in Figure 3. LUIS detects the user's query (utterances) in terms of intents and entities. It sends back the response to the service code where those utterances are examined based on the confidence score. Then, College Enquiry Chatbot labels these utterances, re-trains, and publishes the LU model [17]. Another method of active learning is also implemented which involves using user's response to re-train the model. The endpoint provided by LUIS responds with intents,

entities, and a corresponding confidence score. A threshold is maintained based on the confidence score and correspondingly prompts are provided to the user. When the users respond to it, the system keeps track of the responses and uses it to re-train the model.

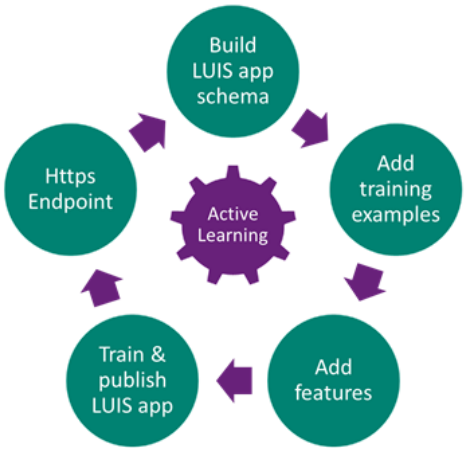


Figure 3: Active Learning Process [16]

2.1.3.3 QnA Maker

For answering simpler queries, instead of detecting intents and entities, College Enquiry Chatbot uses QnA Maker for structuring answers from a semi-structured document like FAQs. This web-based service is useful in incorporating multiple FAQs from an URL, structured documents, product manuals, or editorial contents; and building the LU model for natural language processing.

2.1.4 Cosmos DB (Mongo)

Cosmos DB is a database service provided by Microsoft Azure. College Enquiry Chatbot uses this database service to store my conversational history.

3 IMPLEMENTATION DETAILS

In this section of the report, step by step demonstration of how to setup the bot, LUIS (for NLP), and QnA Maker as well as explanation of service code which includes the implementation of the Active Learning are provided. In addition, setting up databases to store the conversation history is explained. All these steps serve as a template to get a basic bot deployed and published on Azure cloud.

The followings are couple of the prerequisites that are needed in order to get started:

- Microsoft Azure Student Account
- Installation of Git (https://git-scm.com/download/win) and NodeJS (https://nodejs.org/en/download/) on Windows.

3.1 Azure Bot Setup

Open https://portal.azure.com as shown in Figure 4 and follow the steps explained in the following subsections.

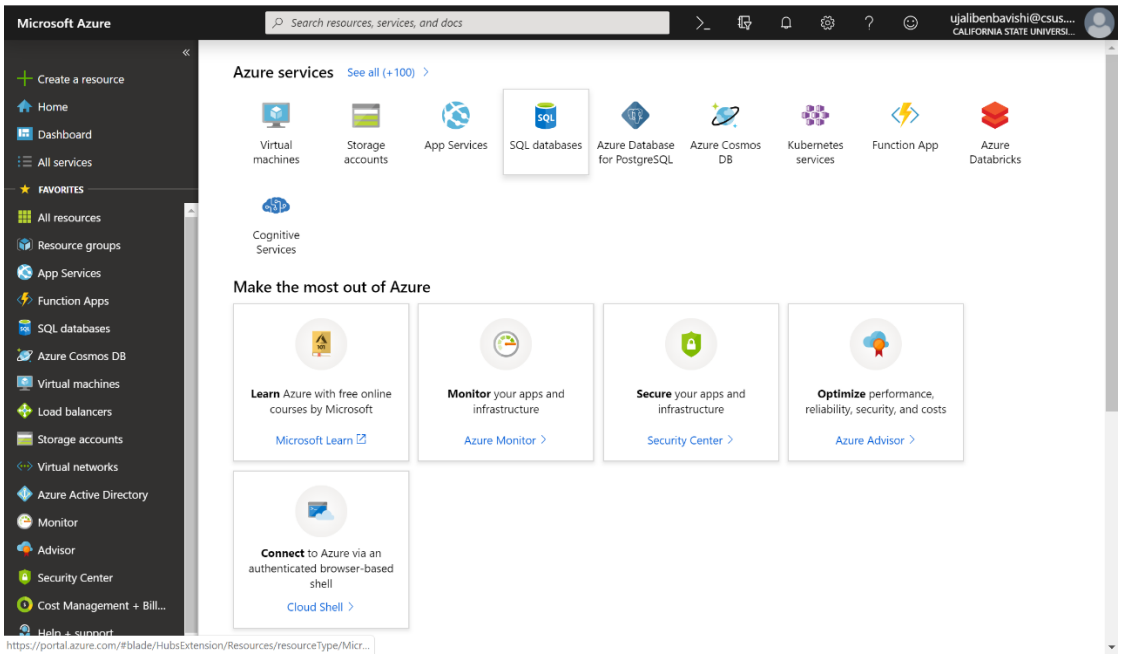


Figure 4: Azure Portal Home Page

3.1.1 Bot Creation

In the left side pane, click on “Create a Resource” and search for “Web App Bot”. As a next step click on “Create” to start creating a bot. Fill up all the required fields as shown in Figure 5.

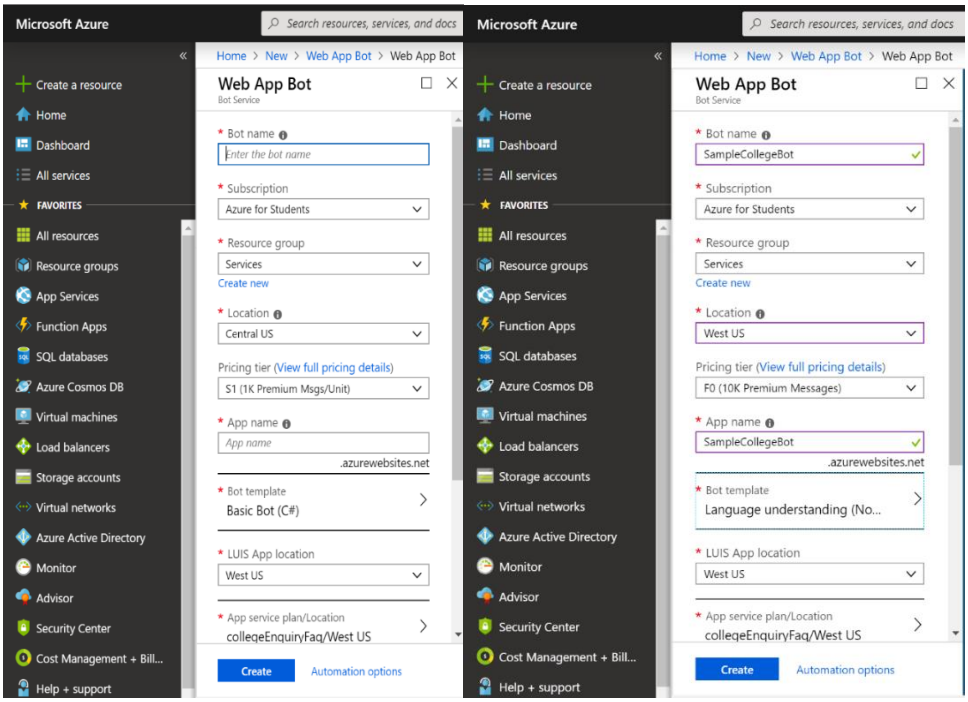


Figure 5: Web App Bot Creation

3.1.2 Adding the DirectLine Channel

After the bot has been created, several channels can be configured to access the bot such as Skype, Webchat, Slack, Facebook, etc. In this project, DirectLine channel is used to communicate with the bot. Figures 6 and 7 show how to add a channel to a bot. After the channel is added, copy the secret key to be used in the service and click on “Done”.

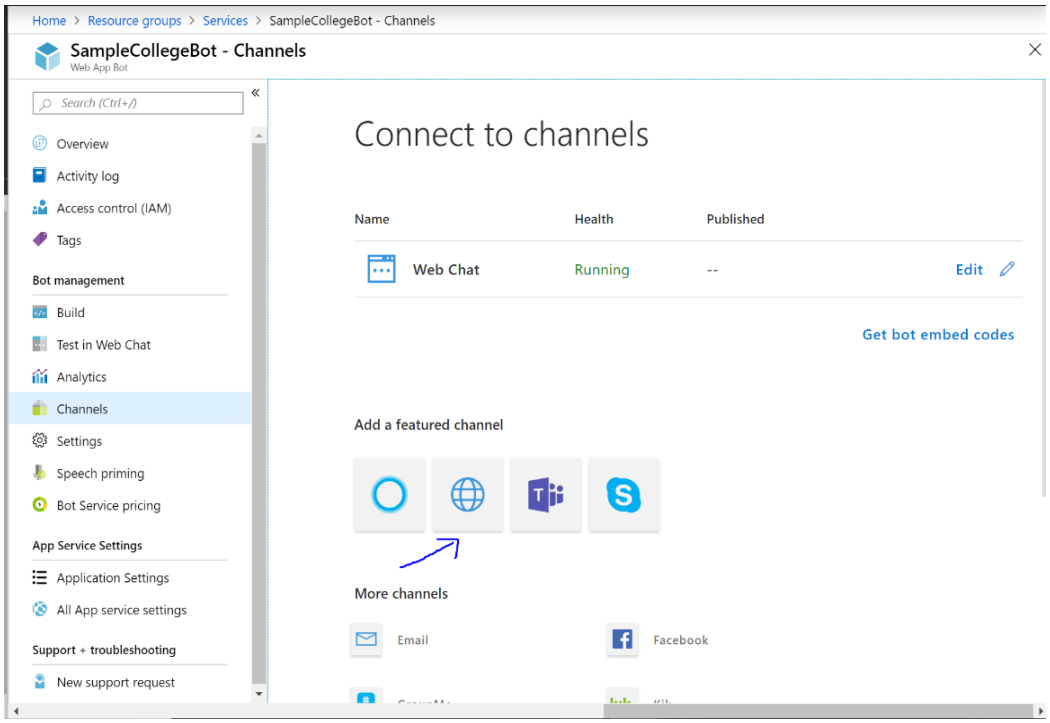


Figure 6: Channel Configuration Page of the Bot

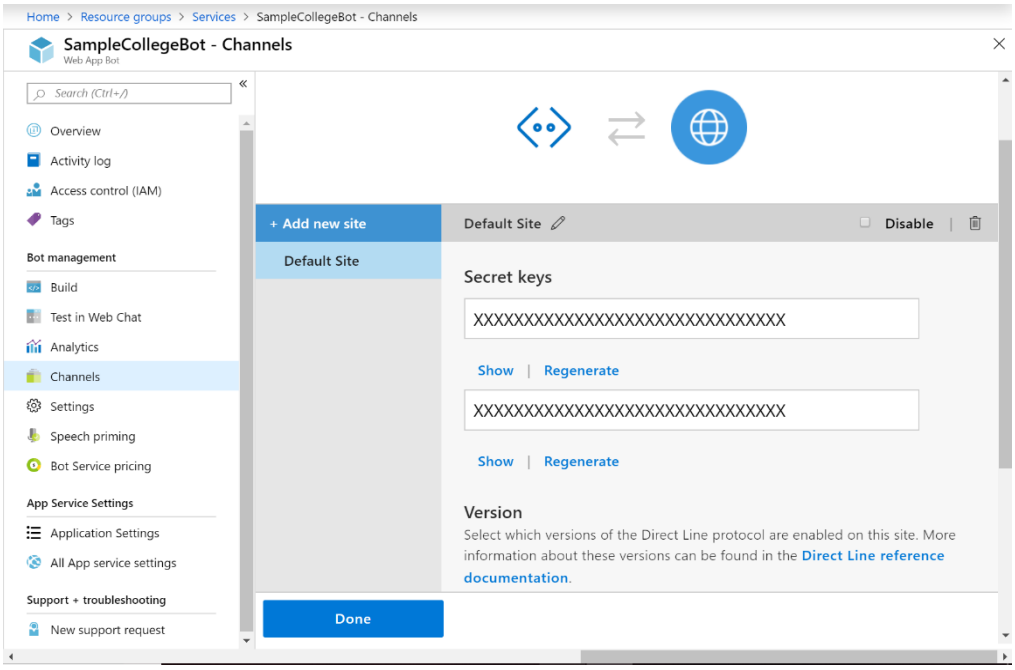


Figure 7: Configure DirectLine Channel for the Bot

3.1.3 Testing the Echo Bot

After the channel is configured, we can test the new bot with some preconfigured bot template where we can echo back whatever the user queries as shown in Figure 8.

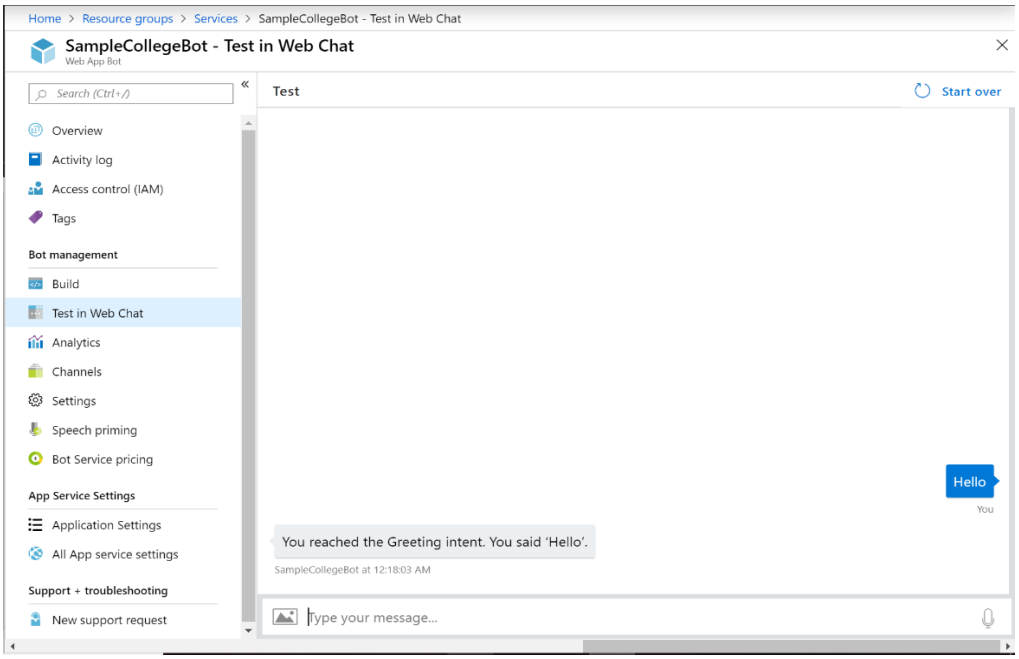


Figure 8: Web Chat Testing of the Created Bot

3.2 LUIS Setup

Go to https://www.luis.ai to setup the intents and entities to create a model and train the natural language processor.

3.2.1 LUIS App Creation

After visiting the website, sign in with the same Azure credentials, scroll down and click on the button “Create a LUIS app now” as shown in Figure 9. Please note that if a Web

App Bot is created with LUIS bot template, then a LUIS App will be created automatically with preconfigured intents and entities to work with.

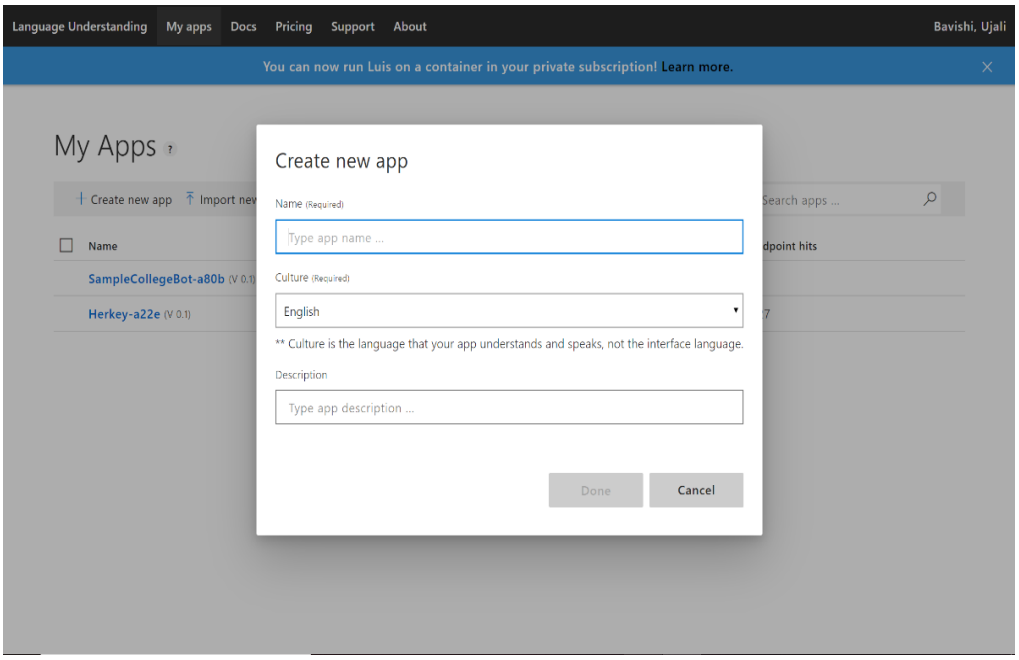


Figure 9: LUIS App Creation

3.2.2 Intents Creation

Go to created LUIS app and click on “Create new intent” and provide a specific intent name as shown in Figure 10. The next step would be to add training data to train the intent with all different types of utterances as shown in Figure 11.

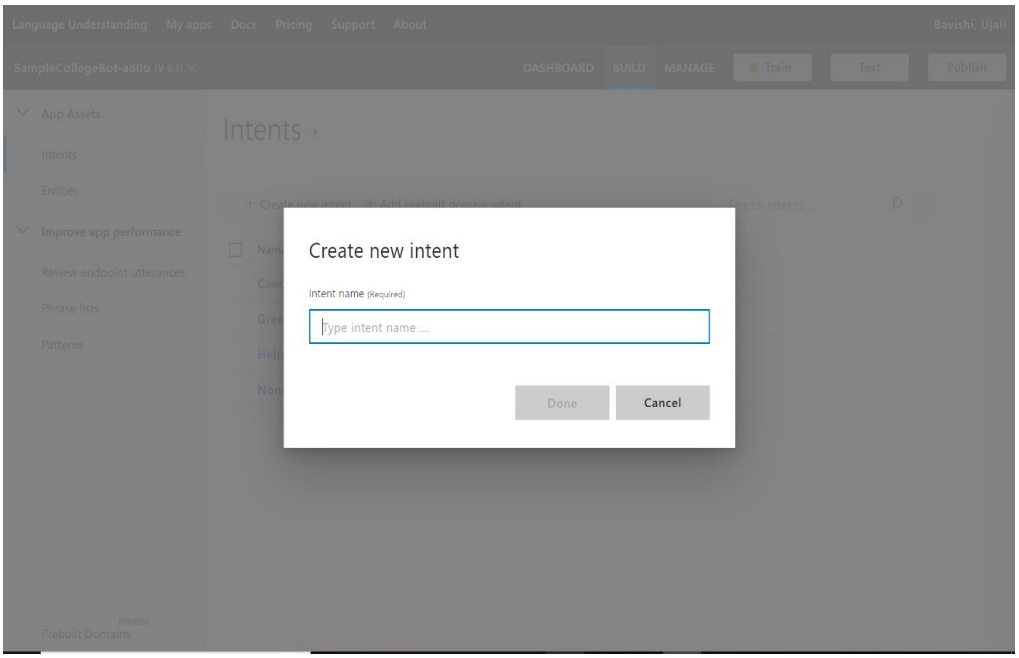


Figure 10: Intent Creation in LUIS

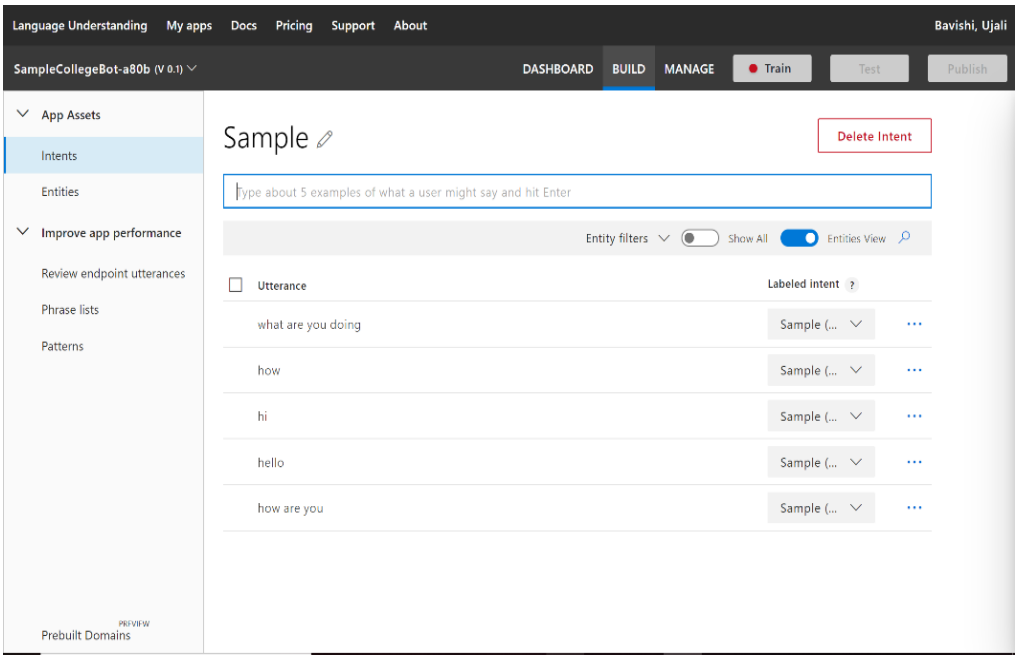


Figure 11: Adding Training Data to Created Intent

3.2.3 Entity Creation

In the left side pain, click on “entities” and then select “Create an Entity” and provide a name for the entity. In this project, a list type of entity is used since the professor and course values are stored in the entity sub list as shown in figures 12 and 13.

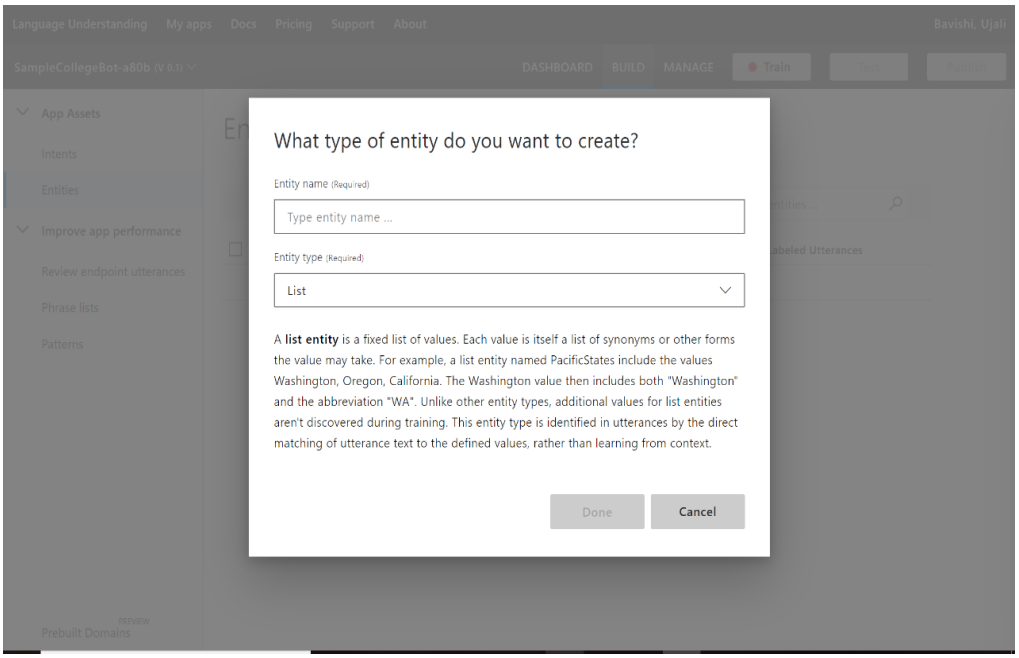


Figure 12: Entity Creation in LUIS - 1

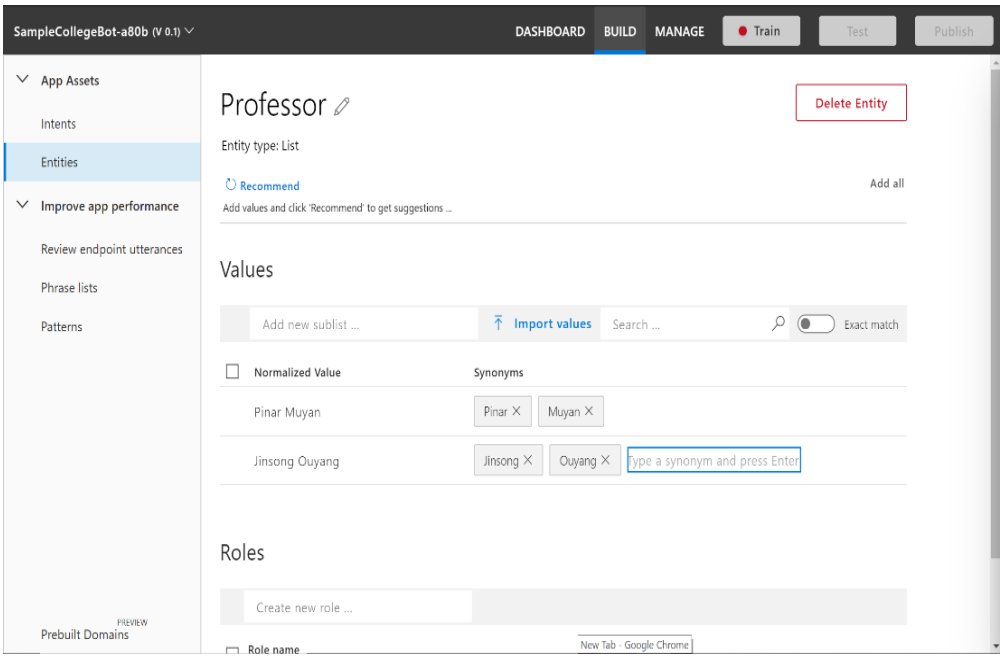


Figure 13: Entity Creation in LUIS - 2

3.2.4 Train and Publish the Model

After adding the intents and entities, the model is trained and published. After the model is published, REST API can be accessed from the service to access the LUIS endpoint to get the intent and entity from the user's query.

3.3 QnA Maker Setup

Go to https://www.qnamaker.ai to setup the questions and answers from the FAQs listed in any website and also to include some of the responses to be displayed depending on user's emotions based on some metadata.

3.3.1 QnA Maker App Creation

After visiting the website, sign in with the same Azure credentials, click on “Create a knowledge base” from the top navigation bar as shown in Figure 14. Then, follow the steps as shown in figures 15 and 16.

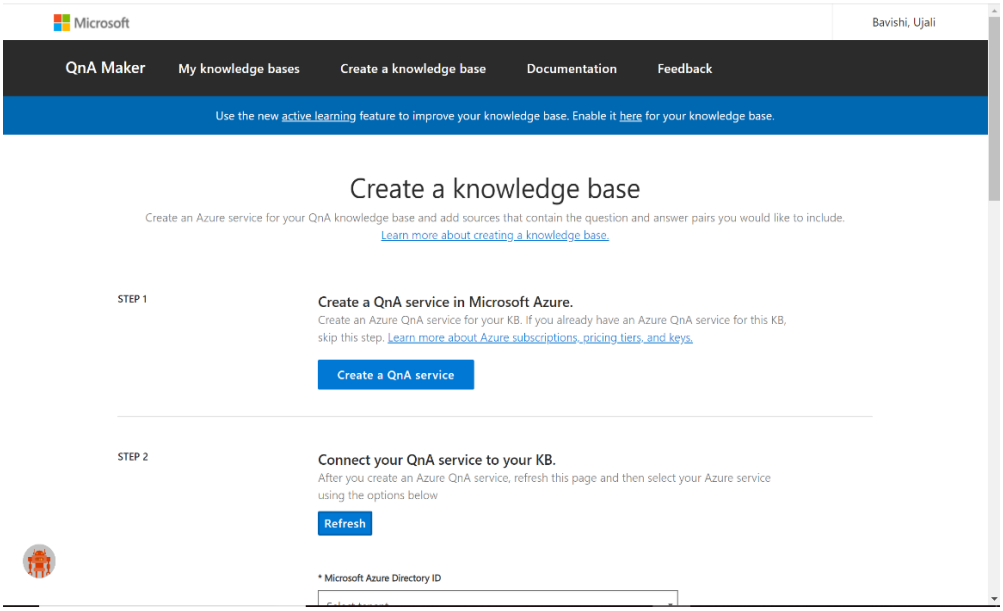


Figure 14: QnA Maker App Creation - 1

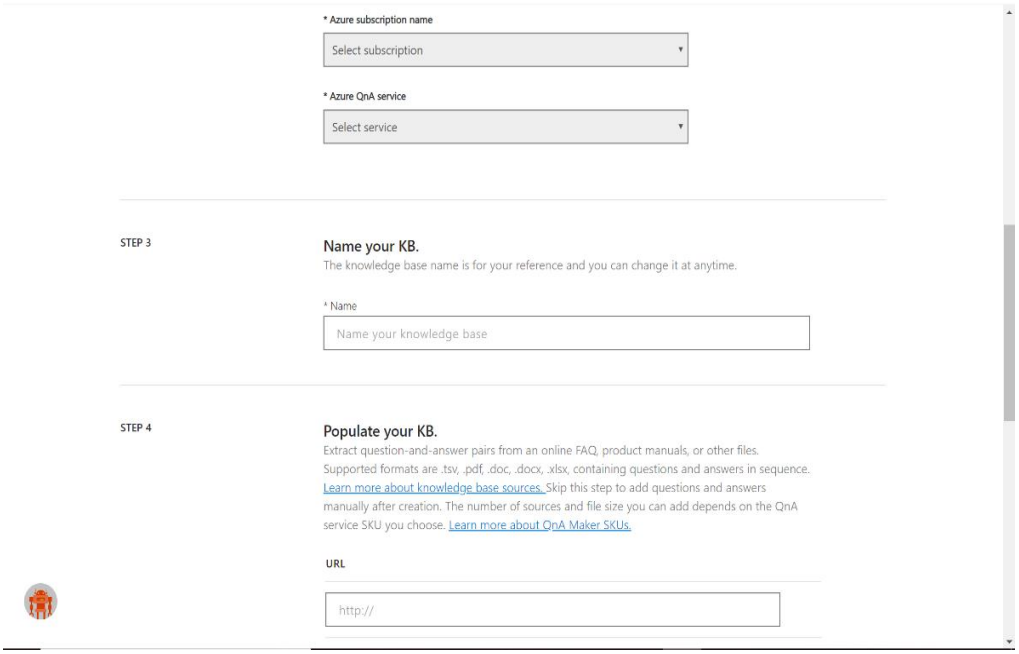


Figure 15: QnA Maker App Creation - 2

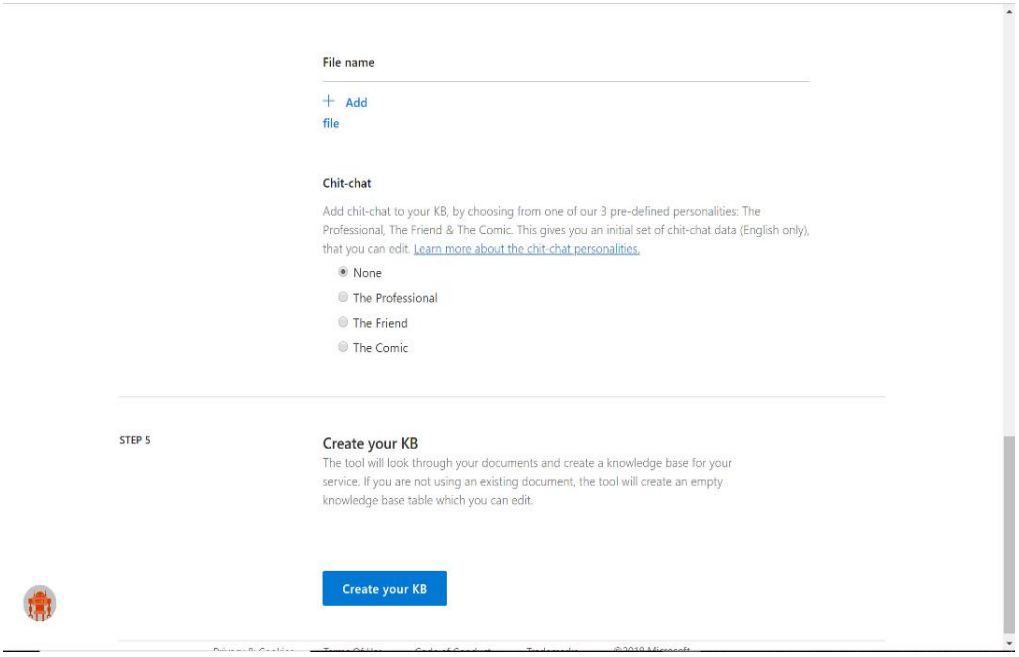


Figure 16: QnA Maker App Creation – 3

3.3.2 Add/Manage Knowledge base

Click on “Settings” from the top navigation and add FAQs URL from the CSUS website which are needed to be included in the knowledge base as shown in Figure 17

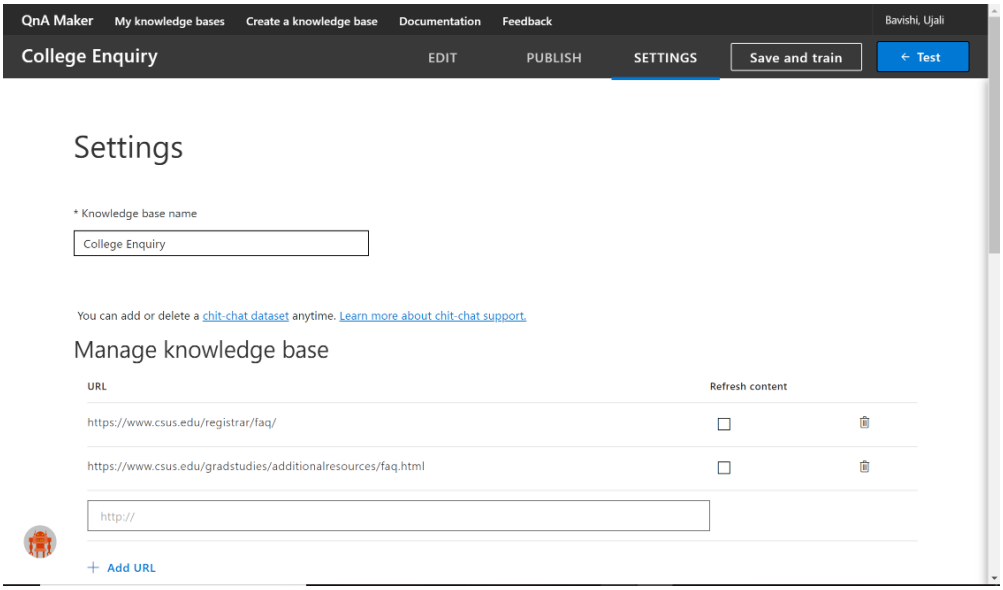


Figure 17: Knowledge Base Configuration Page

3.3.3 Train and Publish the Knowledge Base

After the knowledge bases are added, the next step is to train the app and publish to get a published endpoint to use the QnA Maker service.

3.4 Service Code Setup

I am using Azure DevOps to setup continuous deployment with a source control repository where the code will reside.

3.4.1 Create Project, Repository, and Branches

Sign in to Azure DevOps using the Azure account, create a new project and setup a repository based on the needs as shown in figures 18-20.

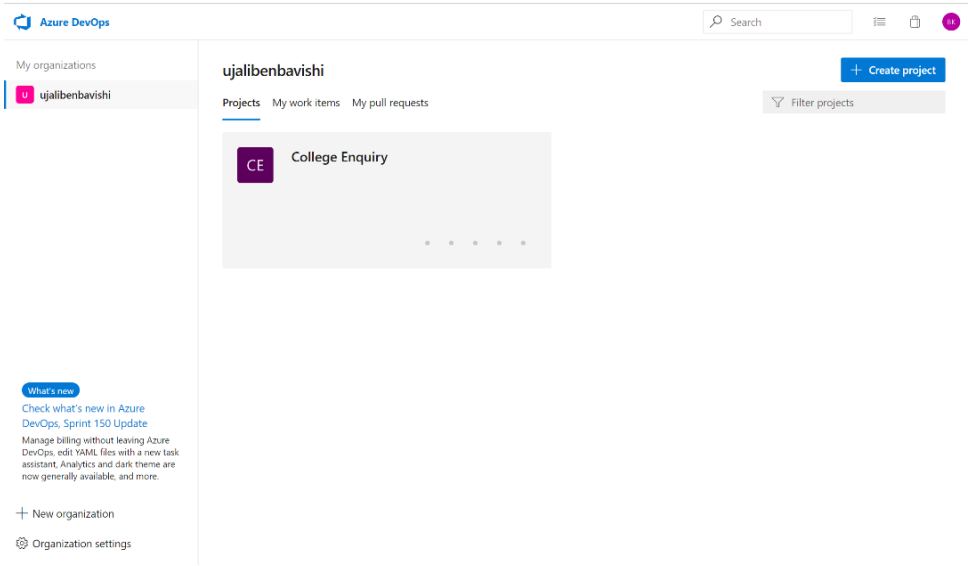


Figure 18: Azure DevOps Source Control Repo

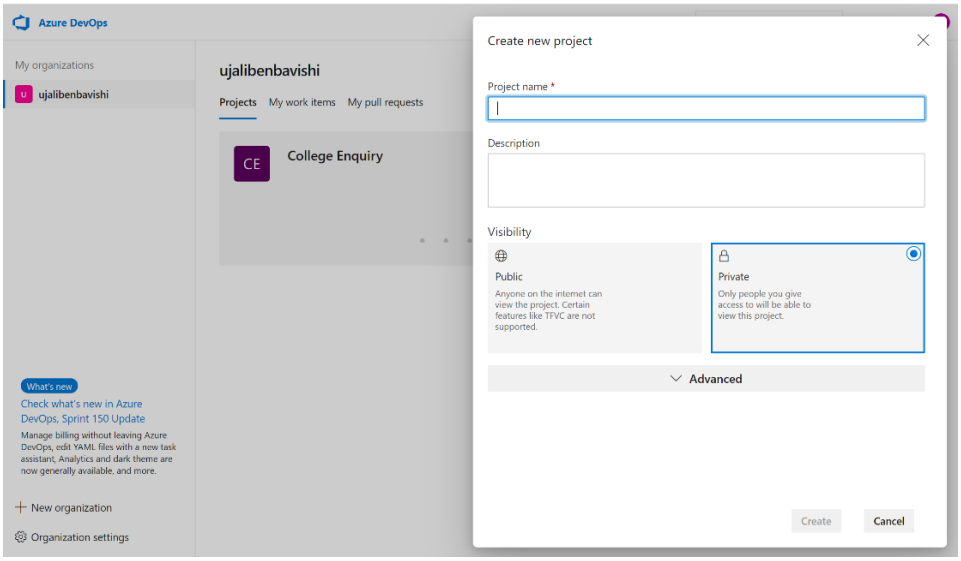


Figure 19: Project Creation on Azure DevOps

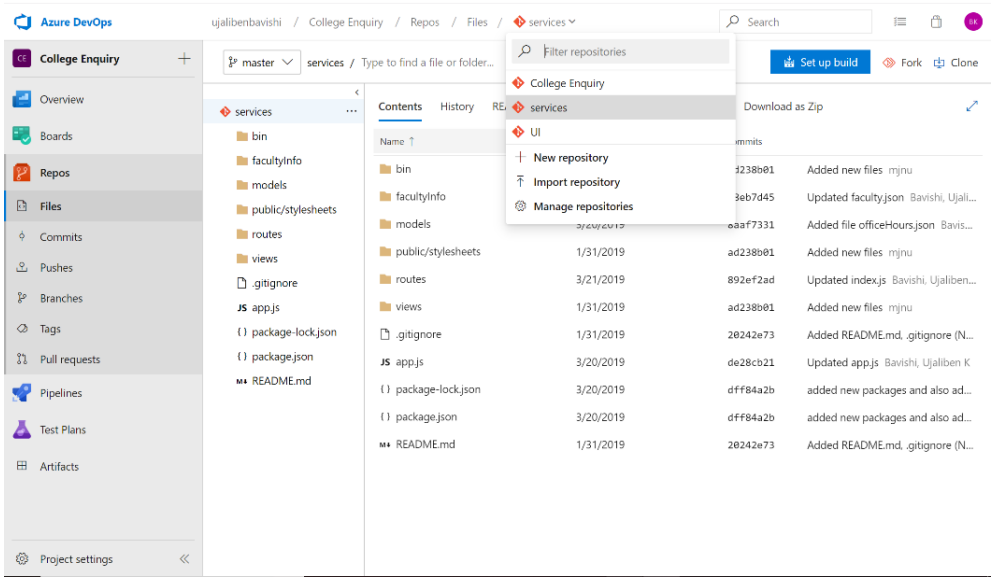


Figure 20: Repository and branch configuration in Azure DevOps Project

3.4.2 Sync Service Code with Azure DevOps

After the repository is setup, use Git clone functionality to clone the repo onto the local system and add all the files related to the service into the project. Finally, when a Git push is made from the same folder, all the files will be copied over to Azure DevOps.

3.5 Setup Continuous Deployment

As a last step, we need to configure the project created on Azure DevOps with the application created on Azure portal as shown in figures 21-24.

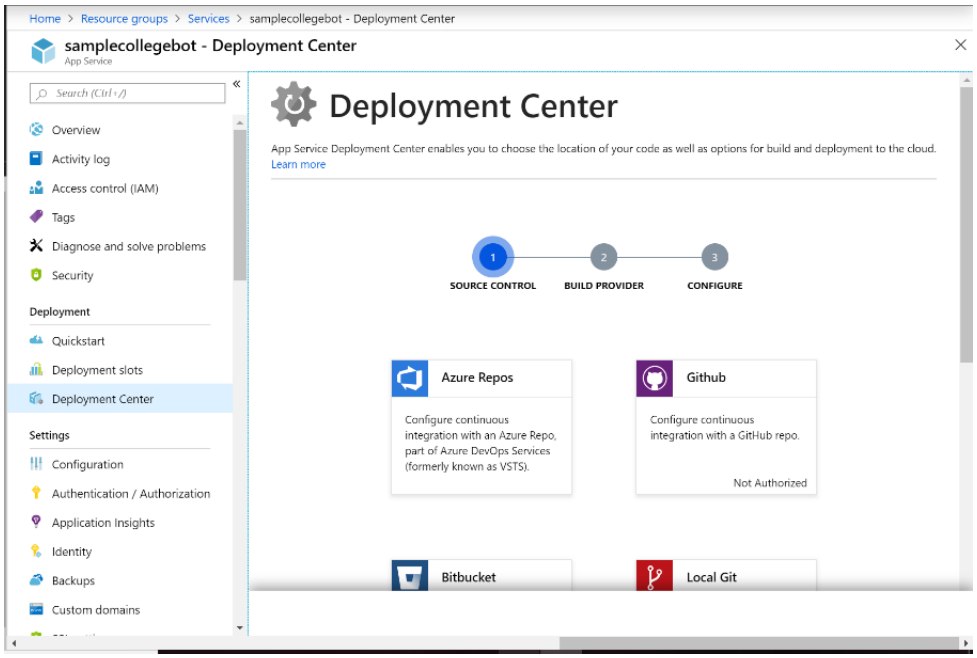


Figure 21: Continuous Deployment Configuration - 1

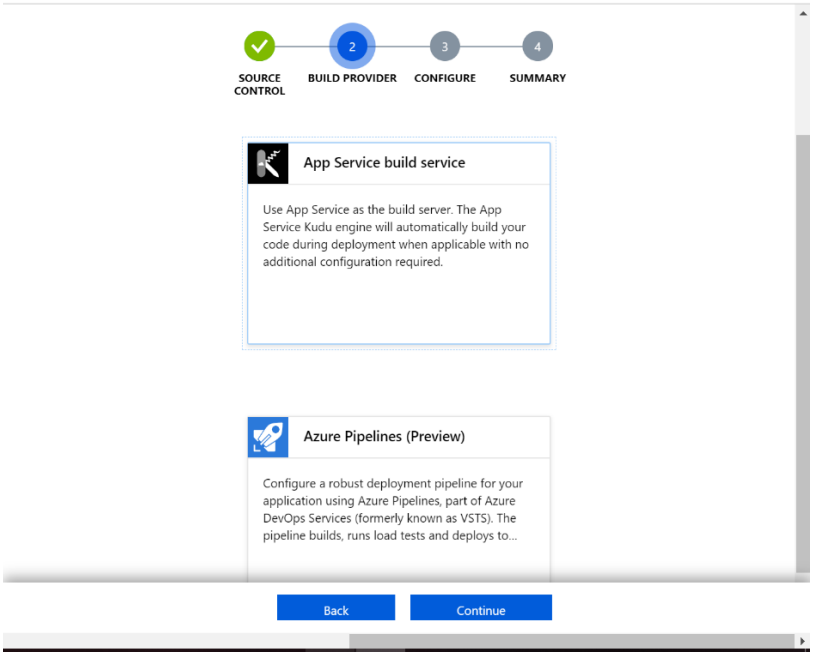


Figure 22: Continuous Deployment Configuration - 2

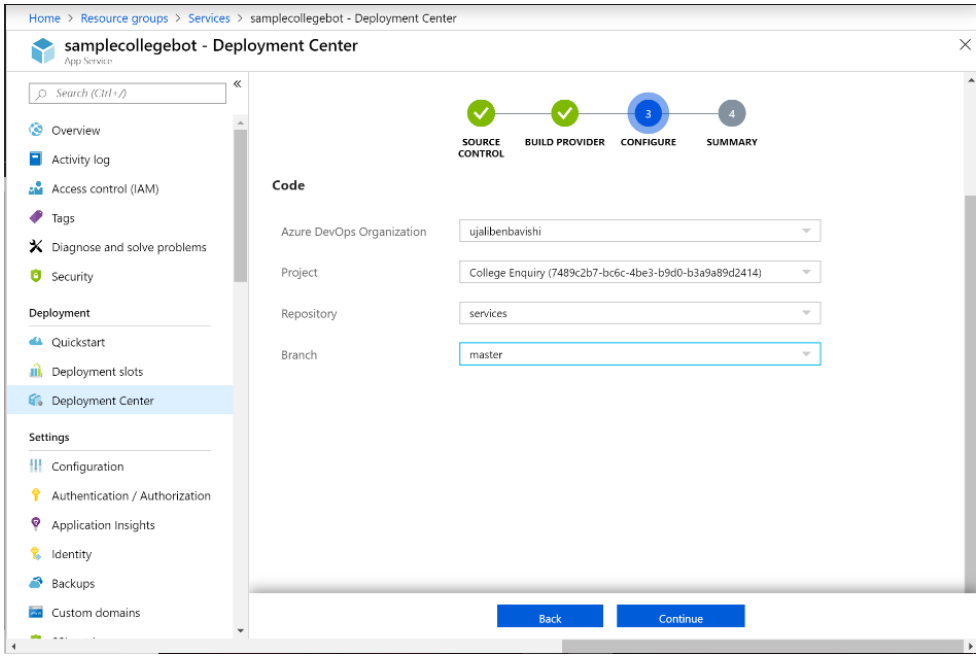


Figure 23: Continuous Deployment Configuration - 3

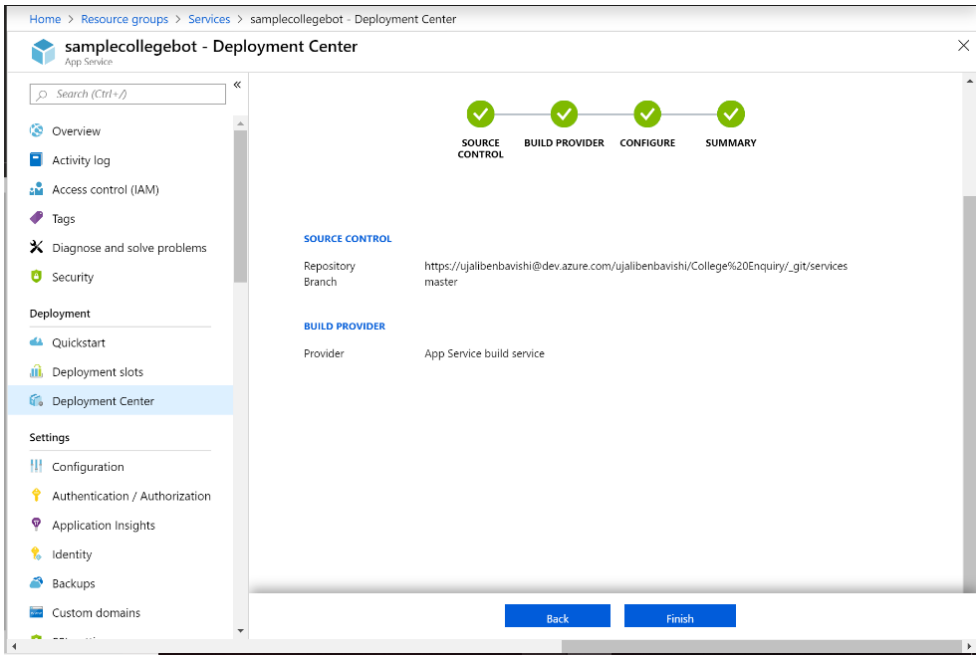


Figure 24: Continuous Deployment Configuration – 4

3.6 Service Code Snippets

In order to provide overview of the service code and some of the utility functions that are used by the bot, this section provides some screenshots. Figure 25 shows the overall structure of the code which utilizes NodeJS and ExpressJS frameworks.

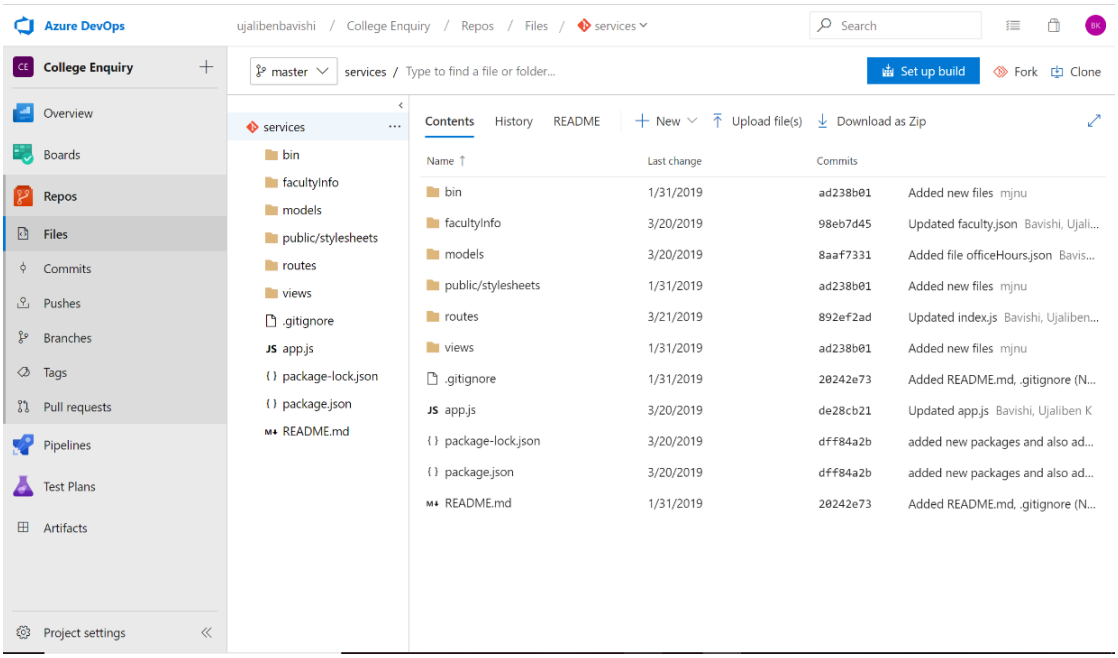


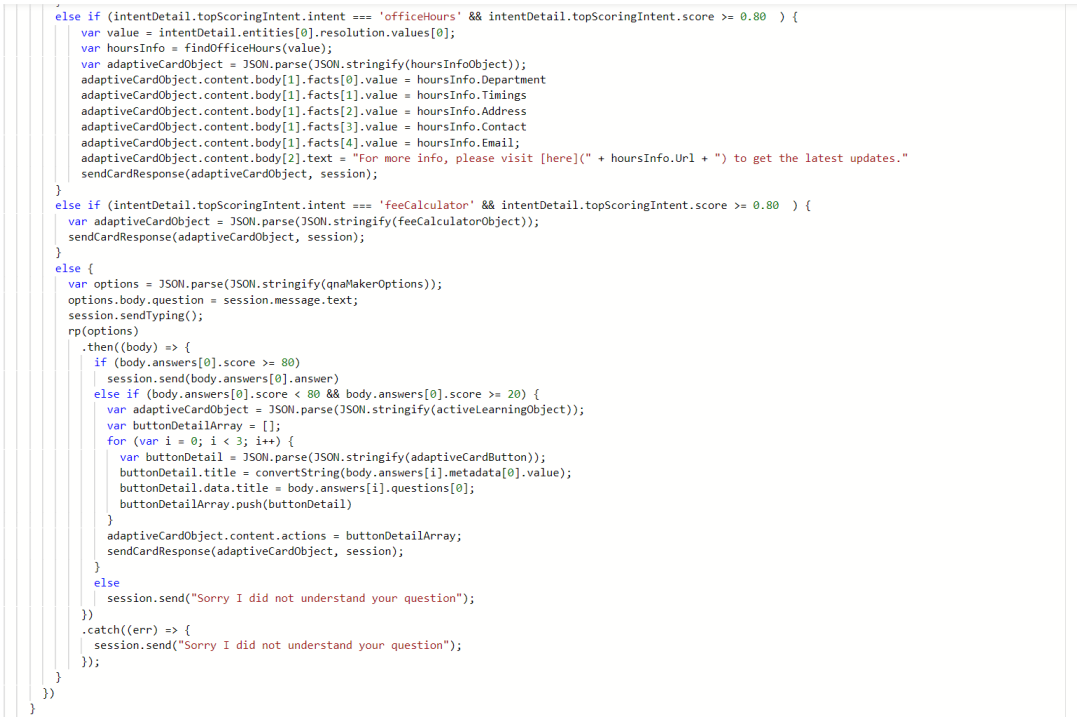
Figure 25: Code Structure Overview

Figures 26 and 27 show various functionalities including calling LUIS REST API, intercepting bot messages to save the chat history to the database and presenting the logic to show feedback card used in active learning.

```
bot.use({
  botbuilder: function (session, next) {
    if (session.message && session.message.value) {
      var studentType = session.message.value.studentType;
      var credits = session.message.value.credits;
      var registrationFee = 0;
      var totalFee = 0;
      if (studentType === "UnderGraduate") {
        if (parseInt(credits) > 6)
          totalFee = 3655 + parseInt(credits) * 396;
        else
          totalFee = 2449 + parseInt(credits) * 396;
      }
      else if (studentType === "Graduate") {
        if (parseInt(credits) > 6)
          totalFee = 4372 + parseInt(credits) * 396;
        else
          totalFee = 2866 + parseInt(credits) * 396;
      }
      var resp = "Based on your input, the total fees would be approximately $" + totalFee + ".00";
      session.send(resp);
    }
    else if (session.message && session.message.text !== "") {
      getIntentOptions.url = LuisModelUrl.concat(session.message.text);
      rp(getIntentOptions).then((result) => {
        var intentDetail = JSON.parse(result);
        console.log(result);
        var timeStamp = moment(new Date()).tz("America/Los_Angeles").format("MM/DD/YYYY hh:mm:ss a");
        var updates = {};
        if (intentDetail.sentimentAnalysis) {
          if (Math.round(intentDetail.sentimentAnalysis.score) === 1) {
            updates = { $setOnInsert: { startTimeStamp: timeStamp }, $set: { endTimeStamp: timeStamp }, $inc: { positive: 1 }, $push: { conversations: { $each: [{ "use
            }
            else {
              updates = { $setOnInsert: { startTimeStamp: timeStamp }, $set: { endTimeStamp: timeStamp }, $inc: { negative: 1 }, $push: { conversations: { $each: [{ "use
            }
          }
          else {
            updates = { $setOnInsert: { startTimeStamp: timeStamp }, $set: { endTimeStamp: timeStamp }, $push: { conversations: { $each: [{ "user": session.message.text,
          }
          insertIntoDb(session.message.address.conversation.id, updates)

          if (intentDetail.topScoringIntent.intent === 'courseInfo' && intentDetail.topScoringIntent.score >= 0.80  ) {
```

Figure 26: LUIS and Active Learning Logic - 1



```
    else if (intentDetail.topScoringIntent.intent === 'officeHours' && intentDetail.topScoringIntent.score >= 0.80  ) {
      var value = intentDetail.entities[0].resolution.values[0];
      var hoursInfo = findOfficeHours(value);
      var adaptiveCardObject = JSON.parse(JSON.stringify(hoursInfoObject));
      adaptiveCardObject.content.body[1].facts[0].value = hoursInfo.Department
      adaptiveCardObject.content.body[1].facts[1].value = hoursInfo.Timings
      adaptiveCardObject.content.body[1].facts[2].value = hoursInfo.Address
      adaptiveCardObject.content.body[1].facts[3].value = hoursInfo.Contact
      adaptiveCardObject.content.body[1].facts[4].value = hoursInfo.Email;
      adaptiveCardObject.content.body[2].text = "For more info, please visit [here](" + hoursInfo.Url + ") to get the latest updates."
      sendCardResponse(adaptiveCardObject, session);
    }
    else if (intentDetail.topScoringIntent.intent === 'feeCalculator' && intentDetail.topScoringIntent.score >= 0.80  ) {
      var adaptiveCardObject = JSON.parse(JSON.stringify(feeCalculatorObject));
      sendCardResponse(adaptiveCardObject, session);
    }
    else {
      var options = JSON.parse(JSON.stringify(qnaMakerOptions));
      options.body.question = session.message.text;
      session.sendTyping();
      rp(options)
        .then((body) => {
          if (body.answers[0].score >= 80)
            session.send(body.answers[0].answer)
          else if (body.answers[0].score < 80 && body.answers[0].score >= 20) {
            var adaptiveCardObject = JSON.parse(JSON.stringify(activeLearningObject));
            var buttonDetailArray = [];
            for (var i = 0; i < 3; i++) {
              var buttonDetail = JSON.parse(JSON.stringify(adaptiveCardButton));
              buttonDetail.title = convertString(body.answers[i].metadata[0].value);
              buttonDetail.data.title = body.answers[i].questions[0];
              buttonDetailArray.push(buttonDetail)
            }
            adaptiveCardObject.content.actions = buttonDetailArray;
            sendCardResponse(adaptiveCardObject, session);
          }
          else
            session.send("Sorry I did not understand your question");
        })
        .catch((err) => {
          session.send("Sorry I did not understand your question");
        });
    }
  })
}
```

Figure 27: LUIS and Active Learning Logic - 2

Figure 28 shows a sample file which contains basic information related to the course like course id, name, information, prerequisites, professor's name and URL of the course.

```
[
    {
        "Course Id": "CSC201",
        "Course Name": "Programming Language Principles",
        "Course Information": "Notations for the specification of programming language syntax and semantics; attribute, translational, operational, axiomatic, algebra
        "Prerequisites of Course": "Fully classified graduate status in Computer Science or Software Engineering",
        "Units": "3",
        "Professor for the course": "Dr. Cui Zhang",
        "Course Url": "https://www.ecs.csus.edu/wcm/csc/pdfs/advising/csc201.pdf"
    },
    {
        "Course Id": "CSC204",
        "Course Name": "Data Models for Database Management Systems",
        "Course Information": "Database management system (DBMS) architecture. Database file organizations and access methods. The relational model and relational alg
        "Prerequisites of Course": "Fully classified graduate status in Computer Science or Software Engineering",
        "Units": "3",
        "Professor for the course": "Bill Mitchell",
        "Course Url": "https://www.ecs.csus.edu/wcm/csc/pdfs/advising/csc204.pdf"
    },
    {
        "Course Id": "CSC205",
        "Course Name": "Computer Systems Structure",
        "Course Information": "Overview of computer systems structure, covering hierarchical structure from software and hardware points of view. Concepts of relocati
        "Prerequisites of Course": "Fully classified graduate status in Computer Science or Software Engineering",
        "Units": "3",
        "Professor for the course": "Dr. Weide Chang",
        "Course Url": "https://www.ecs.csus.edu/wcm/csc/pdfs/advising/csc205.pdf"
    },
    {
        "Course Id": "CSC206",
        "Course Name": "Algorithms and Paradigms",
        "Course Information": "Design and analysis of algorithms. Classical design paradigms including greedy, divide-and-conquer, dynamic programming, and backtracki
        "Prerequisites of Course": "Fully classified graduate status in Computer Science or Software Engineering",
        "Units": "3",
        "Professor for the course": "Dr. Chung-E Wang",
        "Course Url": "https://www.ecs.csus.edu/wcm/csc/pdfs/advising/csc206.pdf"
    },
```


Figure 28: Sample Course Details

Figure 29 shows a sample file which contains basic information related to the professors like name, office hours, location, and email.

```
[
{
    "Name":"Dr. Nik Faroughi, Department Chair (CSC)",
    "Location":"RVR 3018G",
    "Email":"faroughi@ecs.csus.edu",
    "Phone":"916-278-7628 ",
    "Office Hours": "M W F 11:00am-12:00pm",
    "Webpage":"https://athena.ecs.csus.edu/~faroughi/"
},
{
    "Name":"Dr. Scott Gordon, Assoc. Chair (CSC)",
    "Location":"RVR 3018F",
    "Email":"gordonvs@ecs.csus.edu",
    "Phone":"Use Email",
    "Office Hours": "W 2:00pm - 4:00pm",
    "Webpage":"https://athena.ecs.csus.edu/~gordonvs/"
},
{
    "Name":"Dr. Jinsong Ouyang, Graduate Coordinator (CSC)",
    "Location":"RVR 3018I",
    "Email":"jouyang@csus.edu",
    "Phone":"916-278-5769",
    "Office Hours": "Tu R 1:30pm - 3:00pm",
    "Webpage":"https://www.csus.edu/faculty/o/jouyang/"
},
{
    "Name":"Dr. Pinar Muyan-Ozcelik",
    "Location":"RVR 5008",
    "Email":"pmuyan@ecs.csus.edu",
    "Phone":"916-278-6713",
    "Office Hours": "Tu 3:00pm - 3:50pm, R 1:40pm - 3:50pm",
    "Webpage":"https://athena.ecs.csus.edu/~pmuyan/"
},
{
    "Name":"Dr. Haiquan Chen",
    "Location":"RVR 5018",
    "Email":"chenh@ecs.csus.edu",
    "Phone":"916-278-6087",
    "Office Hours": "M W 3:00pm - 4:00pm",
    "Webpage":"https://athena.ecs.csus.edu/~chenh/"
},
```


Figure 29: Sample Faculty Details

4 EXECUTION RESULTS

In this section, execution results of College Enquiry Chatbots are provided by listing several functionalities included in this project and explaining these functionalities with the help of the screenshots.

4.1 How to chat

To start chat with the chatbot, a student will have to go to the university website www.csus.edu and click on the question mark icon as shown in Figure 30.

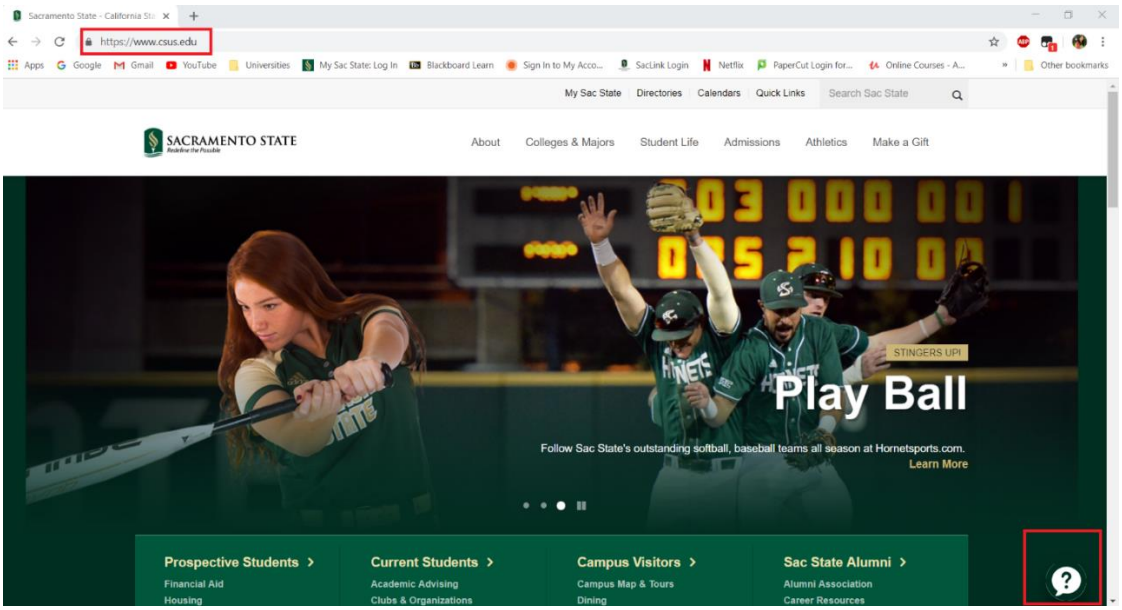


Figure 30: How to Chat

To have the chatbot on a live website, access to the codebase of a website is required. However, for now a software called Tampermonkey is used to inject the Java script into the browser.

4.2 Sentiment Analysis

Figures 31 and 32 show how a bot is answering the irrelevant questions of a user for which normal bots are not showing any kind of empathy. To make the bot to be able to respond these kinds of responses, the bot is trained with the chitchat integration which is provided by QnA Maker in which there are lots of questions and answers pairs are available.

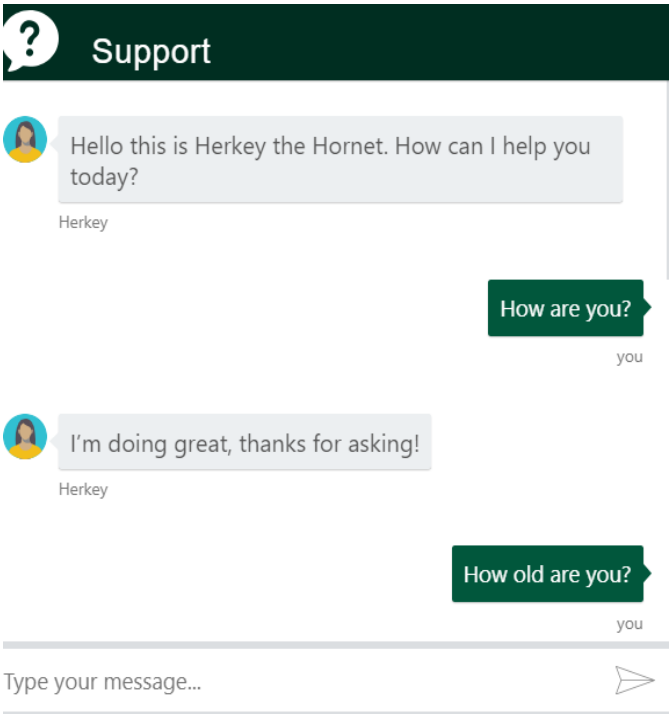


Figure 31: Friendly Conversations - 1

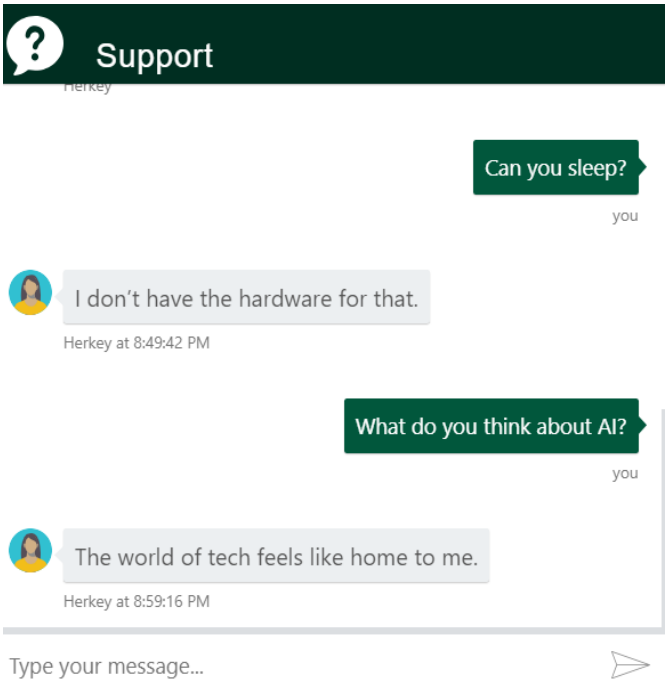
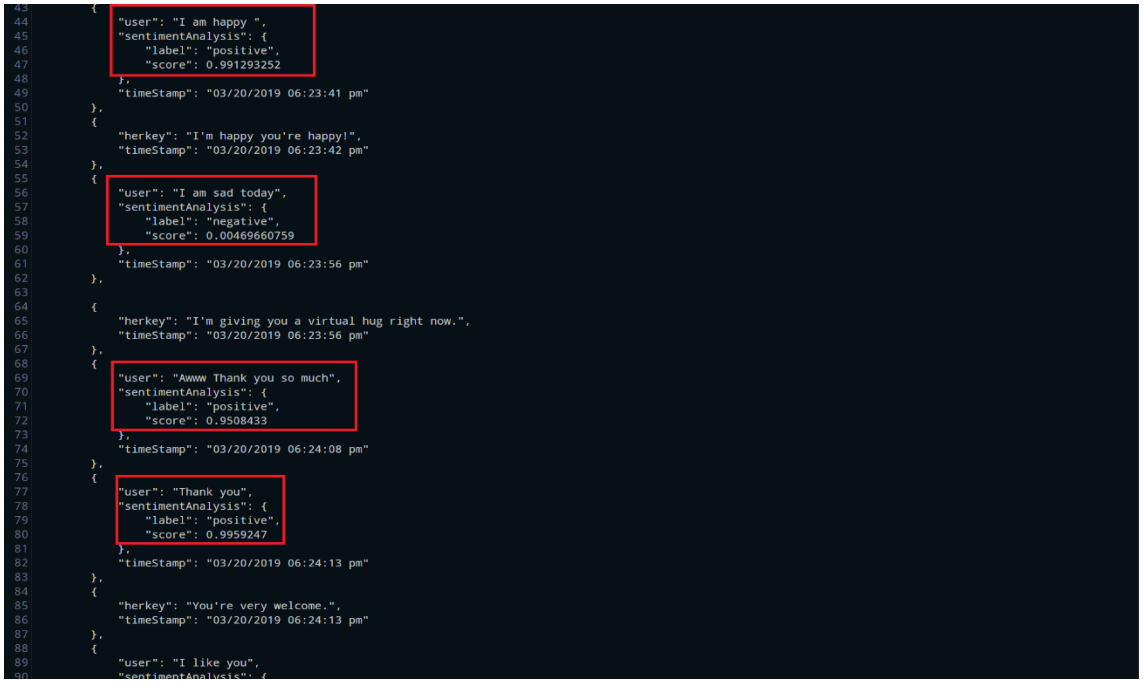


Figure 32: Friendly Conversations – 2

Figures 33 and 34 shows how conversations are being stored in the database and how LUIS performs sentiment analysis on user queries based on which it gives the response back to the user. A bot interceptor is used in the service where conversations are tracked and user's query is passed onto the LUIS endpoint which has an in-built sentiment analyzer to track whether the query is positive, negative or neutral. These results are then stored into the database for any further analysis.

```
{
    "_id": {
        "$oid": "5c92e666fca6e5e29a2628fd"
    },
    "sessionId": "GUOv9o6K91K6B03TidJJtH-g",
    "conversations": [
        {
            "herkey": "Hello this is Herkey the Hornet. How can I help you today?",
            "timeStamp": "03/20/2019 06:18:30 pm"
        },
        {
            "user": "What can you do?",
            "sentimentAnalysis": {
                "label": "positive",
                "score": 0.843127966
            },
            "timeStamp": "03/20/2019 06:20:17 pm"
        },
        {
            "herkey": "I'm here to chat and to try to help out.",
            "timeStamp": "03/20/2019 06:20:18 pm"
        },
        {
            "user": "professor info",
            "sentimentAnalysis": {
                "label": "neutral",
                "score": 0.5
            },
            "timeStamp": "03/20/2019 06:22:47 pm"
        },
        {
            "herkey": "I can help you with that. Please provide the name of the Professor",
            "timeStamp": "03/20/2019 06:22:47 pm"
        },
        {
            "user": "Pinar Muyan",
            "sentimentAnalysis": {
                "label": "neutral",
                "score": 0.5
            },
            "timeStamp": "03/20/2019 06:23:00 pm"
        },
```


Figure 33: Sentiment Analysis on User Queries - 1



```
        {
            "user": "I am happy ",
            "sentimentAnalysis": {
                "label": "positive",
                "score": 0.991293252
            },
            "timeStamp": "03/20/2019 06:23:41 pm"
        },
        {
            "herkey": "I'm happy you're happy!",
            "timeStamp": "03/20/2019 06:23:42 pm"
        },
        {
            "user": "I am sad today",
            "sentimentAnalysis": {
                "label": "negative",
                "score": 0.00469660759
            },
            "timeStamp": "03/20/2019 06:23:56 pm"
        },

        {
            "herkey": "I'm giving you a virtual hug right now.",
            "timeStamp": "03/20/2019 06:23:56 pm"
        },
        {
            "user": "Awww Thank you so much",
            "sentimentAnalysis": {
                "label": "positive",
                "score": 0.9508433
            },
            "timeStamp": "03/20/2019 06:24:08 pm"
        },
        {
            "user": "Thank you",
            "sentimentAnalysis": {
                "label": "positive",
                "score": 0.9959247
            },
            "timeStamp": "03/20/2019 06:24:13 pm"
        },
        {
            "herkey": "You're very welcome.",
            "timeStamp": "03/20/2019 06:24:13 pm"
        },
        {
            "user": "I like you",
```


Figure 34: Sentiment Analysis on User Queries – 2

4.3 Active Learning

Whenever a user asks anything which is outside of the script, the bot asks questions back to the user to understand what exactly user wants to ask and gives the responses accordingly. Figures 35 and 36 shows how a student has asked “on campus” randomly and then bot has asked question in return with three different options to understand what the user means. Once user has selected one of the options it did frame a question based on that option and gave the response to the user.

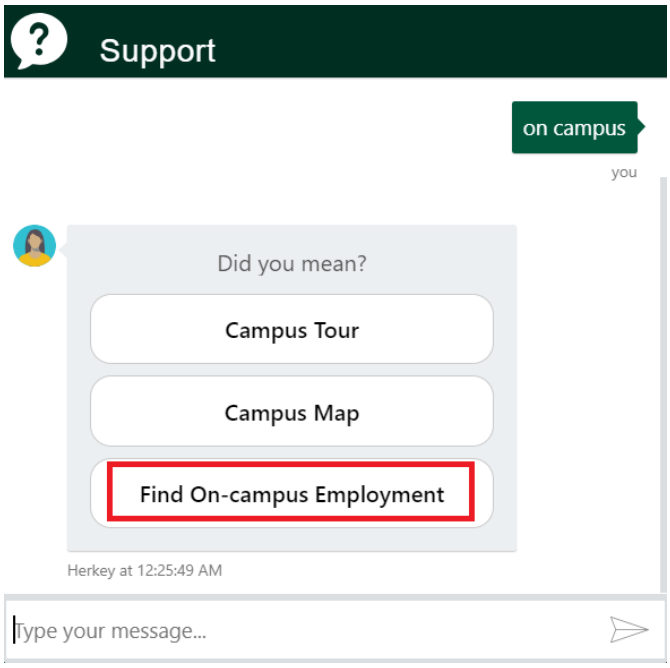


Figure 35: Active Learning - 1

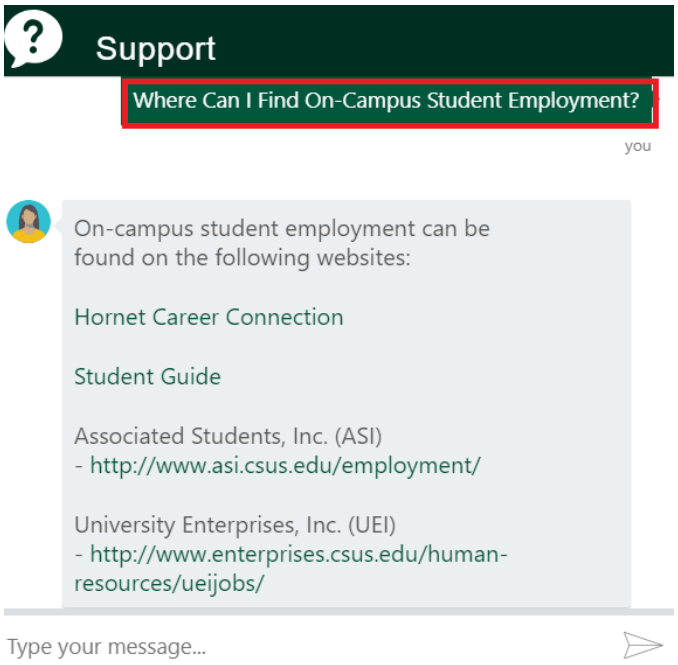


Figure 36: Active Learning – 2

4.4 Frequently Asked Questions

The system inserts FAQs related to academics, registration, records, evaluations/graduation, applications, current students, and campus information. Figures 37 and 38 show two different highlighted questions and answers from two different FAQ URLs which are added to QnA Maker.

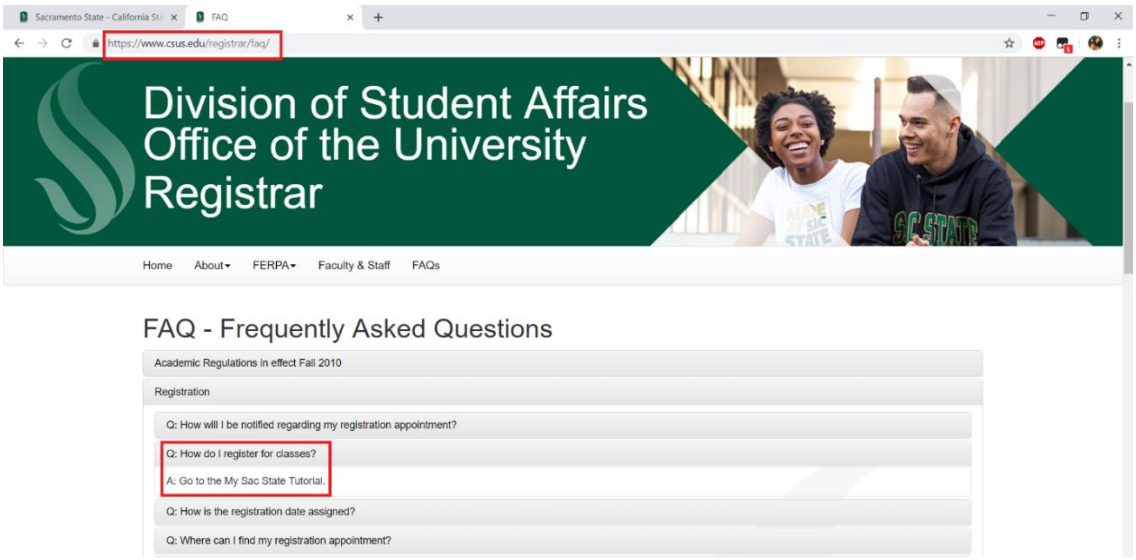


Figure 37: FAQs from Registrar

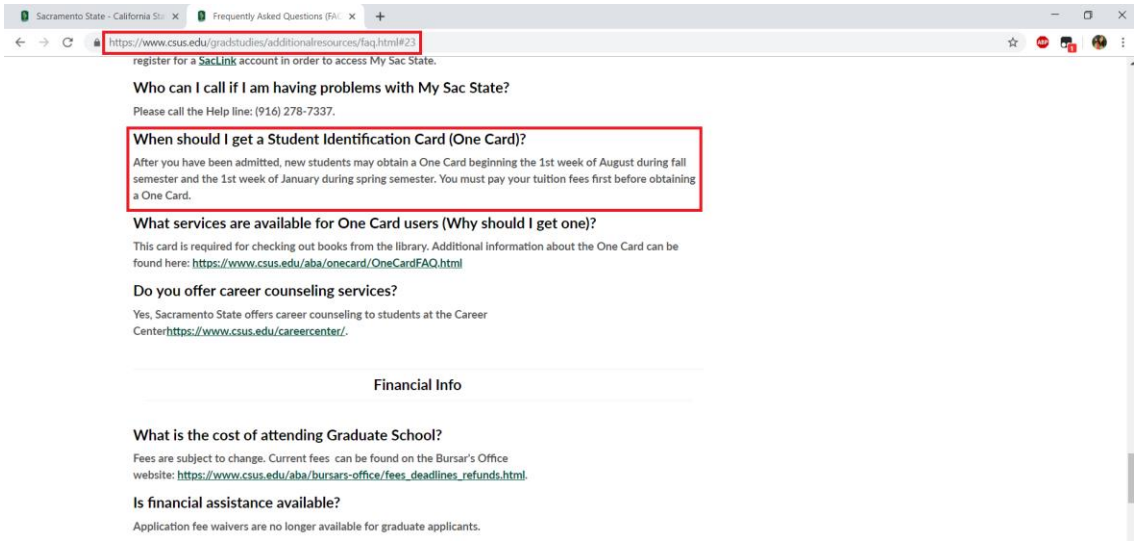


Figure 38: FAQ from Graduate Studies

Figure 39 shows how questions from FAQs are asked and responded in an exact same way. In this case, notice that a student does not have to ask an exact same question included in FAQs. This is the main advantage of an AI-powered chatbot.

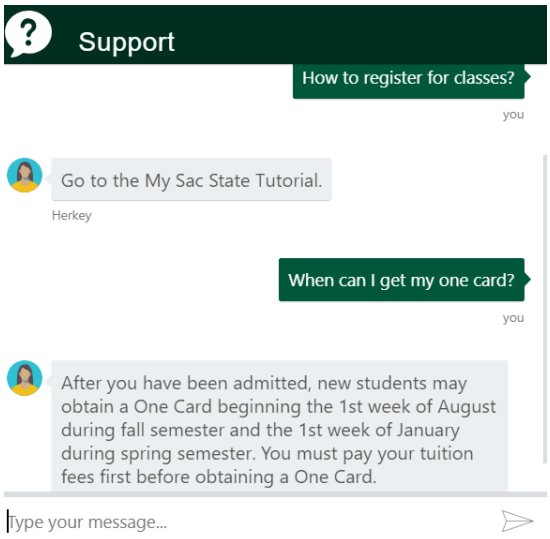


Figure 39: FAQs Asked to Chatbot

4.5 Course and Faculty Details

Figure 40 shows how chatbot is giving the information about course when asked. It shows name, overview, prerequisites, number of units, and professor of the subject. It also gives a course link to the student for more information.

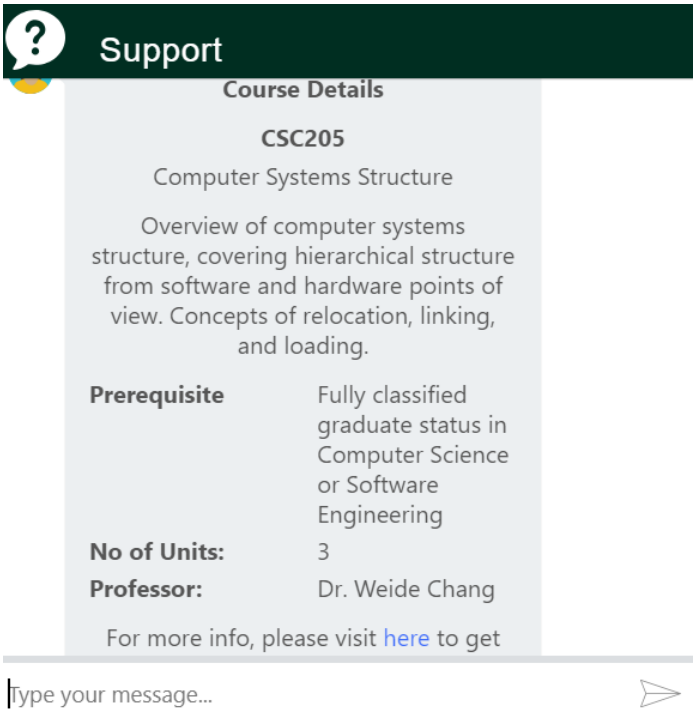


Figure 40: Course Information

Figure 41 shows the chatbot's response about office hours of a faculty. It gives all the necessary details about the faculty such as name, location of the office, email, phone, and office hours of the faculty. It also gives a link to the faculty's website for more information.

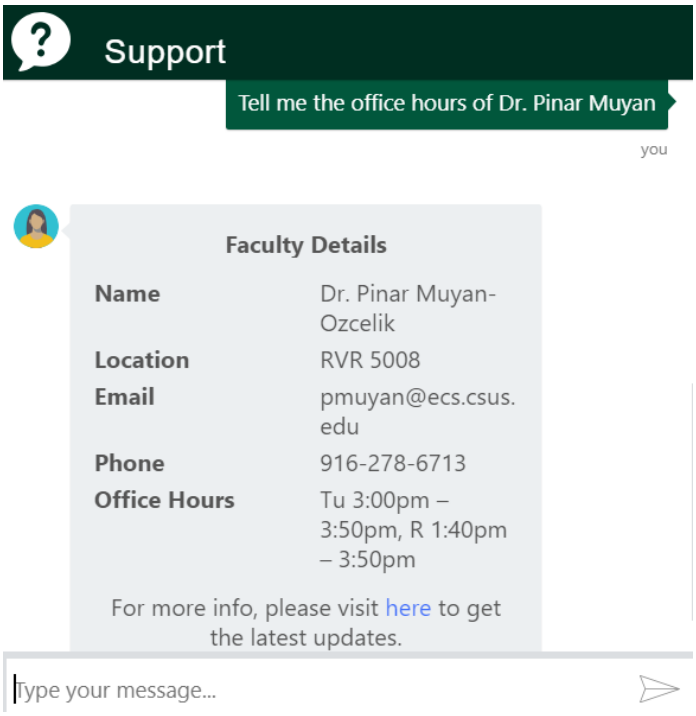


Figure 41: Faculty Information

4.6 Academic Calendar

Figure 42 shows how the bot gives a link to an academic calendar if a student is asking about all the holidays of the semester.

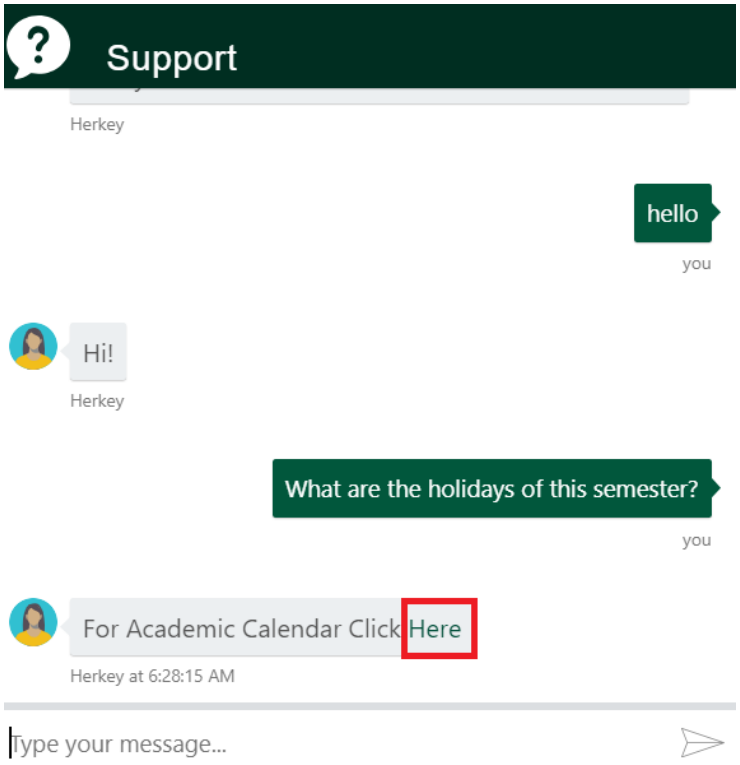


Figure 42: Academic Calendar

4.7 Fee Calculator

Figures 43 and 44 shows about the chatbot providing facility of calculating the fees by just selecting undergraduate or graduate and inserting number of units.

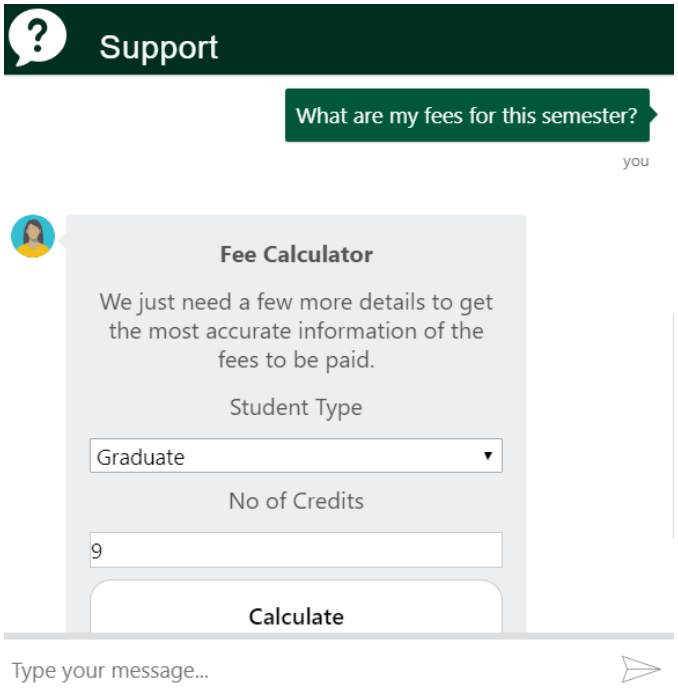


Figure 43: Fee Calculator – 1

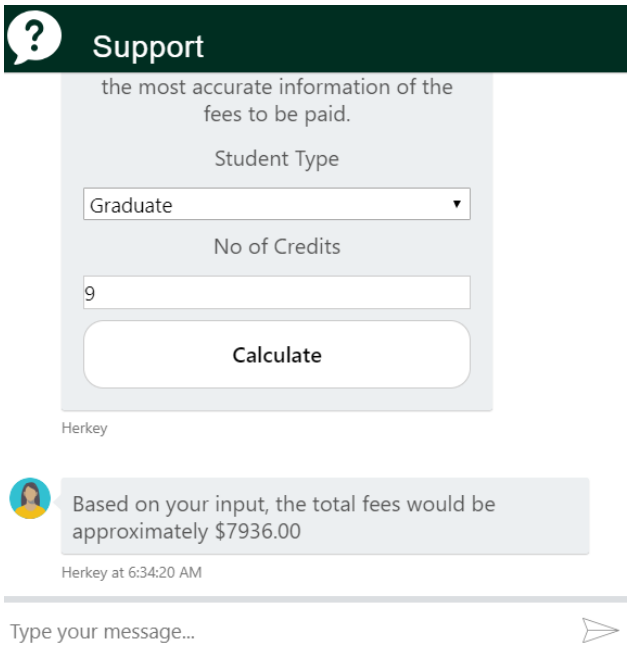


Figure 44: Fee Calculator - 2

4.8 Other Departmental Information

Figures 45 and 46 shows how the chatbot gives information about IPGE and Union office hours, address, contact info, and email. It also gives a link to the respected department for more information.

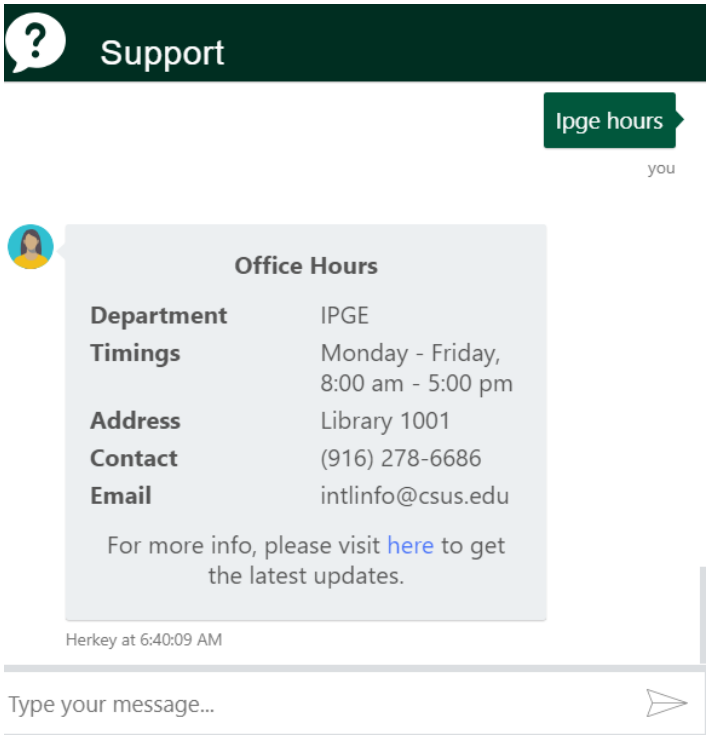


Figure 45: IPGE Hours Information

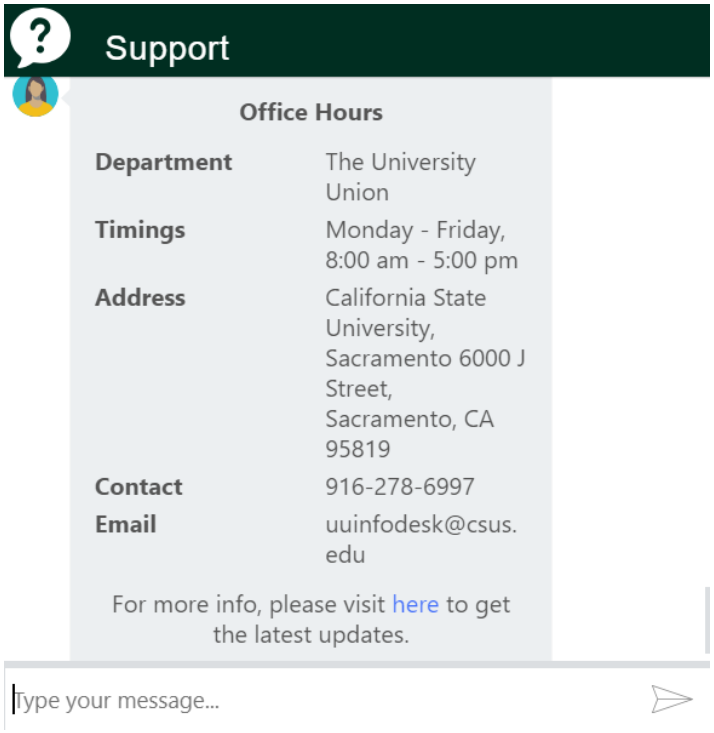


Figure 46: Union Hours Information

4.9 Results

The results of this project are measured in whether sentiment analysis and active learning is correctly implemented or not. Sentiment analysis correctly recognizes the user's query such as positive, negative, and neutral by storing all the conversations in the database (as shown in Figures 33 and 34). These results are used to add empathy to the bot. However, the system was partially successful in adding empathy to the bot. It is because, although large amount of data was added to include some common answers to the queries which are off scripts and to add empathy to the bot (so that it understands what is the current mood of the user and responds accordingly), since scope of these queries is vast, the system

requires more rigorous data to handle all the questions which are out of script. This can be illustrated in the Figure 47 where the bot understood that I am in a negative sentiment, so it answered with a proper response. But, it did not understand the meaning of 'I am excited' since bot did not have any training data related to the typed query as shown in Figure 48.

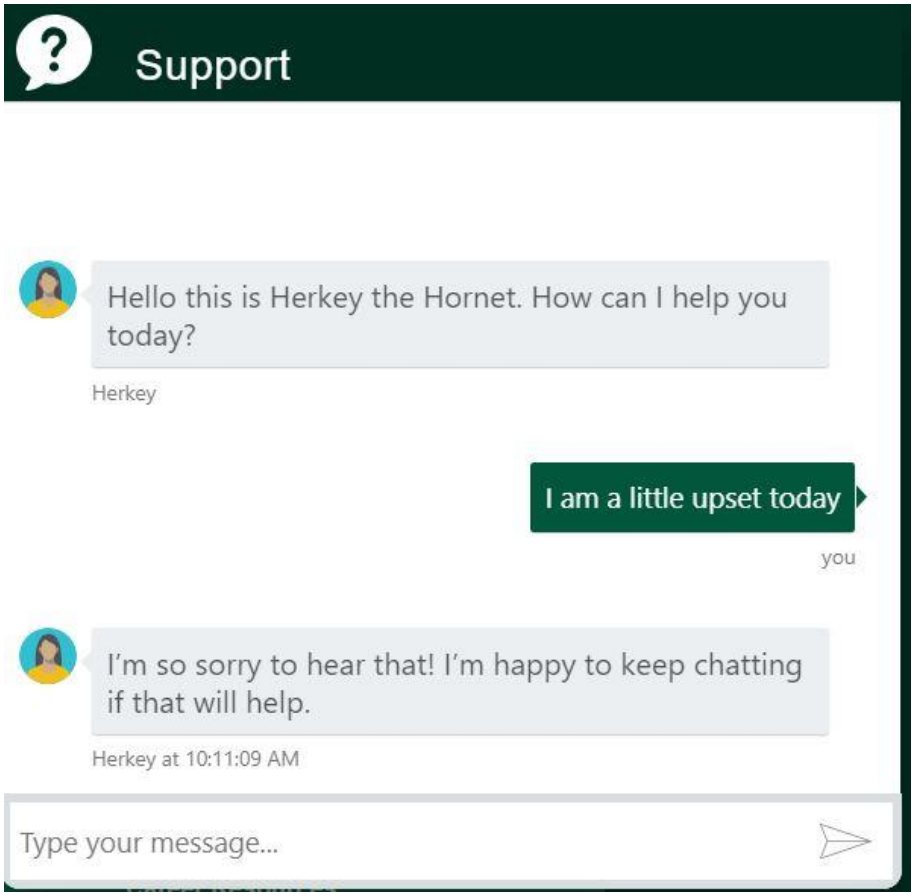


Figure 47: Bot successfully detecting the sentiment of the query

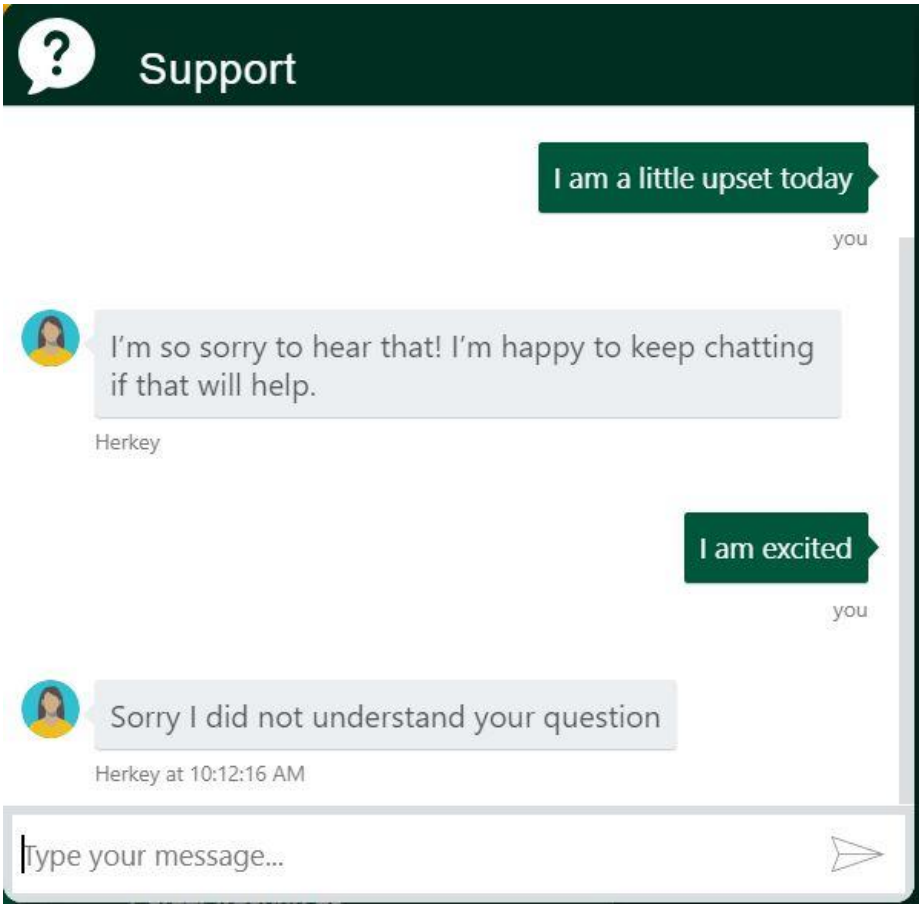


Figure 48: Bot failed to detect the sentiment of the query

On the other hand, active learning helps to improve the bot performance for handling off-script queries. It correctly understands the user's questions, asks clarifying question, and then re-trains the NLP to give response what the user is intended to get.

5 CONCLUSION AND FUTURE WORK

To conclude, College Enquiry Chatbot is helpful in guiding students with correct and most up to date sources of information. It is advantageous for international applicants for queries such as fee payment and academic matters. Students can get the information at their fingertips rather than visiting college office. It improves efficiency by taking over tasks for which humans are not essential.

Sentiment analysis implemented in College Enquiry Chatbot correctly recognizes the user's query such as positive, negative, and neutral by storing all the conversations in the database. However, the system was partially successful in adding empathy since scope of these queries is vast and the system requires more rigorous data to handle all the questions which are out of script. Nevertheless, active learning helps to improve the bot performance for handling off-script queries.

To improve the current functionalities of College Enquiry Chatbot, in the future, the scope of the chatbot can be increased by inserting data for all the departments, training the bot with varied data, testing it on live website, and based on that feedback inserting more training data to the bot. Some of the new features which can be added to the bot are 1) speech recognition feature through which students can ask their queries verbally and get the answers from the bot, 2) integration with multiple channels such as phone call, SMS, and various social media platforms like Skype, Facebook and Twitter, 3) handling context aware and interactive queries in which bot will be aware of the context of an ongoing conversation with a student, 4) integration with services such as password reset and course

enrollment, and 5) adding a capability for the bot to perform analytics based on user's sentiment based on which the bot can be re-trained on human emotions so that more empathy can be added to the bot.

References

[1] "Chatbot", en.wikipedia.org, 2010. [Online]. Available: https://en.wikipedia.org/wiki/Chatbot. [Accessed Jul. 10, 2018].

[2] "Chatbot: What is Chatbot? Why are Chatbots Important? -", Expertsystem.com, 2018. [Online]. Available: https://www.expertsystem.com/chatbot/. [Accessed Sep. 11, 2018].

[3] "Amazon.com: Online Shopping for Electronics, Apparel, Computers, Books, DVDs & more", Amazon.com, 2017. [Online]. Available: https://www.amazon.com/. [Accessed Sep. 12, 2018].

[4] "Amazon Alexa - Build for Amazon Echo Devices", Developer.amazon.com, 2017. [Online]. Available: https://developer.amazon.com/alexa. [Accessed Sep. 12, 2018].

[5] "Meet Erica, Your Financial Digital Assistant From Bank of America", Bank of America, 2018. [Online]. Available: https://promo.bankofamerica.com/erica/. [Accessed Sep. 12, 2018].

[6] C. Doug Gross, "Facebook: Get Messenger app or else - CNN", CNN, 2014. [Online]. Available: https://www.cnn.com/2014/07/29/tech/social-media/facebook-messenger/index.html. [Accessed Sep. 12, 2018].

[7] A. Storman, "5 Chatbot Challenges and How to Overcome Them", Chatbots Magazine, 2016. [Online]. Available: https://chatbotsmagazine.com/5-chatbot-challenges-and-how-to-overcome-them-caccc3a26d7c. [Accessed Sep. 11, 2018].

[8] A. M. Rahman, A. A. Mamun and A. Islam, "Programming challenges of chatbot: Current and future prospective," *in 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Dhaka, 2017.*

[9] "Problems with Chatbots Today - Attunix", Attunix, 2018. [Online]. Available: https://attunix.com/problems-chatbots-today/. [Accessed Sep. 11, 2018].

[10] "Azure Bot Service - chatbot | Microsoft Azure", Azure.microsoft.com, 2017. [Online]. Available: https://azure.microsoft.com/en-us/services/bot-service/. [Accessed Jul. 22, 2019].

[11] "LUIS (Language Understanding) – Cognitive Services – Microsoft Azure", Luis.ai, 2017. [Online]. Available: https://www.luis.ai/home. [Accessed Jul. 15, 2018].

[12] "QnA Maker", Qnamaker.ai, 2016. [Online]. Available: https://www.qnamaker.ai/. [Accessed Jul. 17, 2018].

[13] "What is Azure Bot Service? - Definition from WhatIs.com", WhatIs.com, 2018. [Online]. Available: https://whatis.techtarget.com/definition/Azure-Bot-Service. [Accessed Jan. 17, 2019].

[14] "Azure App Service Documentation - Tutorials, API Reference", Docs.microsoft.com, 2015. [Online]. Available: https://docs.microsoft.com/en-us/azure/app-service/. [Accessed Dec. 17, 2018].

[15] "Cognitive Services | Microsoft Azure", Azure.microsoft.com, 2017. [Online]. Available: https://azure.microsoft.com/en-us/services/cognitive-services/. [Accessed Dec. 22, 2018].

[16] "LUIS.AI: Automated Machine Learning for Custom Language Understanding", azure.microsoft.com, 2017. [Online]. Available: https://azure.microsoft.com/en-us/blog/luis-ai-automated-machine-learning-for-custom-language-understanding/. [Accessed Jul. 22, 2019].

[17] "Review user utterances - Language Understanding - Azure Cognitive Services", Docs.microsoft.com, 2019. [Online]. Available: https://docs.microsoft.com/en-us/azure/cognitive-services/luis/luis-how-to-review-endoint-utt. [Accessed Sep. 14, 2018].

Chat-Bot For College Management System Using A.I

Article · September 2018

CITATIONS
55

READS
11,537

4 authors, including:

Kumkum Bala
Bharati vidyapeeth's college of engineering lavale pune
4 PUBLICATIONS **96** CITATIONS
SEE PROFILE

Sahil Pandita
South Indian Education Society's Graduate School of Technology
1 PUBLICATION **55** CITATIONS
SEE PROFILE

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 04 Issue: 11 | Nov -2017 www.irjet.net p-ISSN: 2395-0072

Chat-Bot For College Management System Using A.I

Prof.K.Bala[1],Mukesh Kumar[2], Sayali Hulawale[3], Sahil Pandita[4]

[1]*Professor, Computer Department, Bharati Vidyapeeth's College of Engineering, Lavale, Pune, Maharashtra*
[2,3,4] *Student, Computer Department, Bharati Vidyapeeth's College of Engineering, Lavale, Pune, Maharashtra*

Abstract- *A chat-bots aims to make a conversation between both human and machine. The machine has been embedded knowledge to identify the sentences and making a decision itself as a response to answer a question. Chat-bots will be completely based on a text-based user interface, allowing the user to type commands and receive text as well as text to speech response. Chat-bots are usually stateful services, remembering previous commands in order to provide functionality. It can be utilized securely by an even larger audience when chat-bots technology is integrated with popular web services. The college inquiry chat-bots will be built using artificial algorithms that analyze user's queries and understand user's message. The response principle is matching the input sentence from a user. The User can ask the question any college-related activities through the chat-bot without physically available to the college for inquiry. The System analyses the question and then answers to the user. With the help of artificial intelligence, the system answers the query asked by the students. The system replies using an effective Graphical User Interface as if a real person is talking to the user. The user just has to register himself to the system and has to login to the system. The chat-bots consists of core and interface that is accessing the core in (MySQL).Natural language processing technologies are used for parsing, tokenizing, stemming and filtering the content of the complaint.*

KEYWORDS: **NLP (Natural language processing), Sentiment Analysis, synsets, Word Net**

1. INTRODUCTION

Chatbot (also known as a talkbot, chatterbox, Bot, IM bot or Artificial Conversational Entity) is a computer program that mimics human conversations in its natural format including text or spoken language using artificial intelligence techniques such as Natural Language Processing (NLP), image and video processing, and audio analysis.

Chat-bot for college management system project will be developed using artificial intelligence algorithms that will analyze users queries. This system will be a web application which will provide answers to the analyzed queries of the user. Users will just have to select the category for queries and then ask the query to the bot that will be used for answering it. Artificial intelligence will be used to answer the user's queries. The user will get the appropriate answers to their queries. The answers will be given using the artificial intelligence algorithms. Users won't have to go personally to the college for inquiry.

The Users has to register to the system and has to login to the system. After login user can access the various helping pages. There will be various helping pages through which the user can chat by asking queries related to college activities. The system will reply to the user with the help of effective graphical user interface (GUI). The user can query about the college-related activities with the help of this web application. College-related activities such as annual day, sports day, Intake and other cultural activities. It will help the students/user to be updated about the college activities.

2. RELATED WORK

[1]Question Answering (QA) systems can be identified as information accessing systems which try to answer to natural language queries by providing answers instead of providing the simple list of document links. QA system selects the most appropriate answers by using linguistic features available in natural language techniques. They differ mainly from the knowledge sources, the broadness of Dialog Systems (NLDS) is an appropriate and easy way to access information. QA system based on Semantic enhancement as well as the implementation of a domain-oriented based on a pattern-matching chat-bots technology developed within an industrial project (FRASI). The proposed approach simplifies the chat-bots realization which uses two solutions. First one is the ontology, which is exploited in a twofold manner: to construct answers very actively as a result of an deduction process about the domain, and to automatically populate, off-line, the chat-bots KB with sentences that can be derived from the ontology, describing properties and relations between concepts involved in the dialogue. Second is to pre-process of sentences given by the user so that it can be reduced to a simpler structure that can be directed to existing queries of the chat-bots. The aim is to provide useful information regarding products of interest supporting consumers to get what they want exactly. The choice was to implement a QA system using a pattern-matching chat-bots technology.

[2]This paper describes an approach to the idea of identifying the most important facts in texts describing the life of a historical figure for building a conversational agent

that could be used in middle-school CSCL scenarios. This paper presents a method for building a chat-bot that can simulate a historical figure. The can receive as "input" a plain text or a web page about the historical figure and has as "output" a trained conversational agent which is able to answer all kind of questions about the life experience of that specific person. the purpose is to provide a generic solution to this problem, so the goal is not to simulate the life and behavior of

[3]Chat-bots are mainly to used to provide conversation between both human and machine. Admin feeds some knowledge to the machine so that machine can identify the sentences and taking a decision itself as a response to answer a question. The chat used is actually Indonesian conversational pattern and the database used in this project is MySQL. It can miss in defining a sentence and how to the response it while connecting chat application to the database. So knowledge representation and implementation of SQL in the pattern-matching operation are needed. A data that has been modeled based on the pattern of the conversation would be tested by the help of a series of scenarios. The conversation with the chat-bots would be crosschecked back to the basic pattern. It is done so that it can add some knowledge to the database as it has not been modeled before. If in case the input sentences in the database did not match then it will be remodeled.

3. PROPOSED SYSTEM

1. User Login and Complaint:

User registers himself/herself on Chat-Bot application. Then submits his/her complaints and queries regarding the electronic and home appliances purchased.

2. Chat BOT Responding System:

a. NLP Processing and Sentiment Analysis for Complaint:

When user complaint is submitted to the system, NLP is applied and sense of the complaint is detected. The sense of the words is found using part of speech tagging and wordnet dictionary.By Using the sentiment analysis negation level of a complaint is detected. And user complaints are prioritized accordingly.

b. Search Questions in knowledge database:

Once the negation level of the complaint is detected, furthermore, the exact question in the complaint is detected using WorldNet.

As the complaint description can change from person to person. The same question may be asked differently from multiple users. One user ask a question so simply and clearly while another user may ask the same question with more negatively. So it is necessary to find what is the exact technical issue with the particular product to give a correct solution

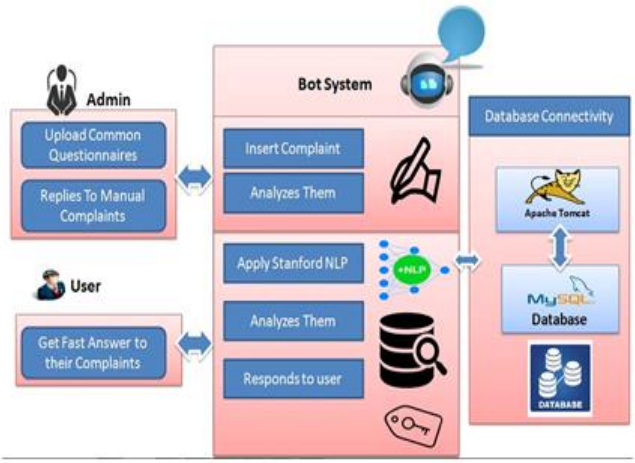


Fig -1: Architecture Diagram For ChatBot

3. Answer the Complaints

As described above whenever user submits a complaint, the negation level and exact issue/question of the complaint are detected. Then it is checked that is there such question registered in database. If the answer is found then that answer is sent to that User. If a particular question is not found in the database such questions are answered by admin person. Once he answered the question the answer is sent to that user. And that question along with answer is stored in database so that whenever such questions will be asked so that they get answered directly from the database. Due to this admin doesn't need to answer same question manually anymore

Interface. WorldNet is a lexical and semantic database for the English language. It is used to group English words into the set of synonyms called synsets, it provides short definitions and usage examples, and records a number of relations among these synonym sets or their members.

4. PROPOSED METHOD

Algorithm Used:

1. Porter Stemmer Algorithm

Porter stemming algorithm (or 'Porter stemmer') is a process for removing suffixes from words in English. Removing suffixes automatically is an operation which is

especially useful in the field of information retrieval. Following are the steps of this algorithm:-

[1]Gets rid of plurals and -ed or -ing suffixes

[2]Turns terminal y to i when there is another vowel in the stem

[3]Maps double suffixes to single ones: -ization, -ational, etc.

[4]Deals with suffixes, -full, -ness etc. Takes off -ant, -ence, etc. Removes a final –e.

2. Word Order Similarity Between Sentences

Let's consider a particular case to illustrate the importance of word order. For example, for two sentences:

T1: A dog jumps over the lazy fox.
T2: A fox jumps over the lazy dog.

These two sentences containing words are exactly same and most words appear in the same order. The only difference is that dog appears before fox in T 1 and dog appears after fox in T 2. As above given two sentences contain the same words, any methods based on "bag of word" give a decision that T 1 and T 2 are exactly the same. However it is clear for a human interpreter that T 1 and T 2 are only similar to some extent. T 1 and T 2 are dissimilar only in word order. Therefore any efficient computational method for sentence similarity must take into account the impact of word order. Sentences consisting of exactly the same words but in different orders may result in very different meanings. It is easy for humans to process word order information. However the incorporation of order information in to computational methods for understanding natural language is a difficult challenge. This may be the reason why most existing methods do not tackle this type of information. In this section we introduce a method that stores the information of word order into account when computing sentence similarity. Let's assume that for the given two sentences, the joint word set is T. Recall the above mentioned two sentences T 1 & T 2, their joint word set is: T = {A dog jumps over the lazy fox} . A unique index number has been assigned to each word in sentences T1 & T2 respectively. The index number is simply the order number that the word appears in the sentence. For example, the index number is 4 for dog and 6 for over in T 1 . In computing word order similarity, a word order vector r is formed for T 1 and T 2 respectively based on the joint word set T . For each word wi in T, we try to find the same or a similar word in T 1 as follows: 1. If T 1 contains an occurrence of the same word, we fill the entry for this word in r 1 with the corresponding index number in T 1 . Otherwise we try to find the most similar word i w ~ in T 1 . 2. If the similarity between wi and i w ~ is greater than a pre-set threshold, the entry of wi in r 1 is filled with the index number of i w ~ in T 1 3. If the above two searches fail, the entry of wi in r 1 is null. After applying the above procedure for given sentences T 1 and T 2 , the word order vectors are r 1 and r 2 respectively. For the example sentence pair, we have: r 1 = {1 2 3 4 5 6 7 }, r 2 = {1 7 3 4 5 6 2} Thus a word order vector is the basic structural information carried by a sentence. The task of dealing with word order is then to measure how similar the word order in two sentences is. For measuring word order similarity of two sentences the proposed measure is as follows:

$$S_r = 1 - \frac{\|\mathbf{r}_1 - \mathbf{r}_2\|}{\|\mathbf{r}_1 + \mathbf{r}_2\|}$$

5.CONCLUSION

We create a software tool which will be used by any company to help the users to freely upload their queries. Once the complaint is registered in the database, automatic tokens are generated and conveyed to the customer through a text message and email for further tracking of the complaint. Natural language processing technologies are used for parsing, tokenizing, stemming and filtering the content of the complaint. The output is fed to the algorithm where the strength of the sentence is calculated. The intensity of negation is calculated, which helps prioritize the complaint automatically for the service provider to resolve the complaint.

In this way, the proposed system will help many organizations to ensure quality service provision and customer satisfaction with less human efforts.

REFERENCES

[1] Agnese Augello, Giovanni Pilato, Alberto Machi' ICAR - Istituto di Calcolo e Reti ad Alte Prestazioni CNR - Consiglio Nazionale delle Ricerche Viale delle Scienze , 978-0-7695-4859-3/12 $26.00 © 2012 IEEE . "An Approach to Enhance Chatbot Semantic Power and Maintainability: Experiences within the FRASI Project".

[2] Emanuela Haller, Traian Rebedea Faculty of Automatic Control and Computers university Politehnica of Bucharest, 978-0-7695-4980-4/13 $26.00 © 2013 IEEE. "Designing a Chat-bot that Simulates a Historical Figure".

[3] Bayu Setiaji, Ferry Wahyu Wibowo , Department of Informatics Engineering STMIK AMIKOM Yogyakarta, Yogyakarta, Indonesia, 2166-0670/16 $31.00 © 2016 IEEE "Chatbot Using A Knowledge in Database-Human-to-Machine Conversation Modeling".

[4] Dungeon Lee, Kyo-Joong Oh, Ho-Jin Choi School of Computing, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, 978-1-5090-3015-6/17/$31.00 ©2017 IEEE." The ChatBot Feels You – A Counseling Service Using Emotional Response Generation".

[5] Wen Hua, Zhongyuan Wang, Haixun Wang, Member, IEEE, Kai Zheng_, Member, IEEE, and Xiaofang Zhou, Senior Member, IEEE." Understand Short Texts by Harvesting and Analyzing Semantic Knowledge".

[6] Anuja P Jain, Asst. Prof Padma Dandannavar , Computer Science, and Engineering, Gogte Institute of Technology, Belgaum, India. 978-1-5090-2399-8/16/$31.00_c 2016 IEEE.