EDUCATIONAL ASSISTANCE BOT

1 ARINDAM ROY, 2 DHARMAL SINGH and 3 SUDIPTA SAHANA

1 PG Student, 2 Associate Professor, 3 Assistant Professor
Department of Computer Science & Engineering
JIS College of Engineering, Kalyani, Nadia-741235

1 Corresponding author: E-mail: arindamroy.roy35@gmail.com
ABSTRACT

An Internet bot performs automated tasks over the Internet. This software application is also known as a web robot or robot or simply bot. Typically, bots runs simple and structurally repetitive tasks, at a much higher rate than would be possible for a human alone. This effort is to help the users by providing all the basic student’s academic details like subjects, faculty names, books, notes and routine according to their need using bot. Users can also get other information like fees structure, library inventory list, calendar etc. Additional information like top educational sites, most trending educational videos etc will be regularly updated by the admin. This is an all-in-one application which can also be used to fetch and update information which is purely related to COVID-19.

Keywords
Internet bot, Academic Student Details, COVID-19

1. INTRODUCTION

Humans are always fascinated with self-operating devices and today, it's software “Chatbots” which are getting more human-like and are automated. The combination of immediate response and constant connectivity makes them an attractive thanks to extend or replace the online applications trend. But how do these automated programs work? Let’s have a look. Chatbots are designed for conversations with users using natural language. With so many tools available to develop conversational agents nowadays, the building and deployment of a Chabot may look fairly simple. However, providing the Chabot with suitable information is to be able to work as an educational tutor could be difficult. At the time of writing, information on how to design the tutor is scarce and scattered across blog entries and articles which are focused on the Chabot implementation rather than the design and modeling of knowledge. Many of the current development tools (such as Dialog flow and Chatfuel) only handle the implementation phase. The task of abstracting and organizing knowledge is up to the instructor designers. Thus, a methodology is required for knowledge abstraction and organization. This work proposes a methodology that formally defines and models the chatbot structure as an intelligent tutor. Its aim is to help multidisciplinary teams looking to design and implement chatbots in university courses. The methodology focuses on describing a first-order logic framework which can be implemented on different commercially available tools, and sheds some light on how to represent, expand and maintain the knowledge base. The advent of bots in 1966 started with text
bots like Eliza, and it later evolved to voice-based bots during the 80’s. The simplest way of defining a bot would be software that can have intelligent conversations with humans.

Though there are various use cases for bots, one among the familiar examples is live chat platforms where users ask questions, which are met with appropriate responses by a chatbot. This is a quick way of resolving queries and providing timely customer service.

2. RELATED WORK

There are many applications that are incorporating a person's appearance and meaning to simulate human dialog, but in most of the cases the knowledge of the conversational bot is stored in a database created by a human expert. However, very few researches have investigated the idea of creating a chatbot with an artificial character and personality starting from web pages or plain text about a certain person. This paper describes an approach to the thought of identifying the foremost important facts in texts describing the life (including the personality) of an historical figure for building a conversational agent that could be used in middle-school CSCL scenarios.

In education, chatbot technology could help human users practice their conversational skills where Q&A systems respond to users from knowledge bases [5,6]. This knowledge bases contains preloaded questions and there corresponding answers. Bot could help in different ways also, retrieving documents from local document collections and Web pages from global Internet [7,8] for extraction of answers[9,10].

FAQ (Frequently Asked Questions) is a knowledge-based Q&A system, we have heard of this term. This FAQ knowledge base has multiple question-answer pairs that is asked frequently by users. FAQ system usually matches users’ questions to corresponding question-answer pairs [11]. Chat robots (or chatterbots or chatbots) is an artificial conversational system that interacts with users; this is an advanced Q&A system. Among some well known chatbots ELIZA [14] was the first chatbot, developed in 1960s. Another famous chatbot ALICE [12] won the Loebner Prize for the most human-like computer in 2000, 2001, and 2004. Since then, chat robots have found wide application areas[15].

3. METHODOLOGY

The methods section describes actions to be taken to implement a friendly chatbot system that can be very helpful for the pandemic. But for the simplicity, here we are basically implementing a student friendly academic system, it will be the base model to analyze and make further modifications
into it to make a perfect chatting environment for the current situation. In recent years, chatbots have become popular in businesses focused on client service. In education interest for chatbots is increasing, designing a chatbot as an intelligent tutor has been very conventional now a days. This paper expresses a formal methodology to design a model and implement a chatbot as an intelligent tutor to answer the queries and to teach lessons about the preloaded topics. The methodology has been implemented using different commercially available tools. It is basically focusing on knowledge abstraction and modeling that makes a conversation flow.

3.1.1 REQUIREMENT SPECIFICATION

Functional requirements could also be calculations, technical details, data manipulation and processing and other specific functionality that outline what a system is meant to accomplish. Behavioral requirements describing all the cases where the system uses the functional requirements are captured in use cases. Functional requirements are supported by non-functional requirements (also referred to as quality requirements), which impose constraints on the planning or implementation (such as performance requirements, security, or reliability). The plan for implementing functional requirements is detailed within the system design. The plan for implementing non-functional requirements is detailed within the system architecture.

In systems engineering and requirements engineering, a non-functional requirement (NFR) may be a requirement that specifies criteria which will be wont to judge the operation of a system, instead of specific behaviors. They are contrasted with functional requirements that outline specific behavior or functions. The plan for implementing functional requirements is detailed within the system design. The plan for implementing non-functional requirements is detailed within the system architecture, because they're usually Architecturally Significant Requirements.

3.1.2 TOOLKITS

ChatterBot: ChatterBot is a Python library that makes it easy to generate automated responses to a user’s input. ChatterBot uses a selection of machine learning algorithms to produce different types of responses. This makes it easy for developers to create chatbots and automate conversations with users. For more details about the ideas and concepts behind ChatterBot see the following diagram.
Figure 1 Chatterbot

WebSocket server: It uses HTTP protocol for handshake and after handshake is complete, it works over TCP protocol and exchanges data in its agreed-upon format called frames. Connections are bi-directional and any party can send message anytime. Unlike HTTP where new TCP connection is made every time you want to communicate, WebSocket maintains a connection using which any side can send message anytime, reducing the message delivery time by using the existing connection.

Figure 2 WebSocket Server

4. PROPOSED WORK

Proposed methods are furnished into multiple steps as follows:
i. This System is a web application which provides answer to the query of the student.

ii. Students just have to query through the bot which is used for chatting.

iii. Students can chat using any format there is no specific format the user has to follow.

iv. The System uses built in artificial intelligence to answer the query.

v. The answers are appropriate what the user queries.

vi. If the answer found to invalid, user just need to select the invalid answer button which will notify the admin about the incorrect answer.

vii. System allows admin to delete the invalid answer or to add a specific answer of that equivalent question.

viii. The User can query any college related activities through the system.

ix. The user does not have to personally go to the college for enquiry.

x. The System analyzes the question and then answers to the user.

xi. The system answers to the query as if it is answered by the person.

xii. With the help of artificial intelligence, the system answers the query asked by the students.

xiii. The system replies using an effective Graphical user interface which implies that as if a real person is talking to the user.

xiv. The user can query about the college related activities through online with the help of this web application.

xv. This system helps the student to be updated about the college activities.

5. IMPLEMENTATION

The entire system will be implemented in a way shown in Data Flow Diagrams below.
After implementation of the system with a user friendly interface it looks like an automated chatting platform (chatbot) which contains all academic details.
6. RESULT ANALYSIS

The chatbot produce results by adopting 3 classification methods: Pattern Matchers: Bots use pattern matching to classify the text and produce a suitable response for the users. Chatbot knows the answer only because his or her name is in the associated pattern. Similarly, chatbots respond to anything relating it to the associated patterns. But it cannot go beyond the associated pattern. To take it to an advanced level algorithm can help. Algorithms for each kind of question, a unique pattern must be available in the database to provide a suitable response. With lots of combination on patterns, it creates a hierarchical structure. We use algorithms to reduce the classifiers and generate the more manageable structure.

7. CONCLUSION

The main objective of this project is to help the users by providing all the basic information, here as an example we have formed student’s academic details like subjects, faculty names, books, notes and routine according to their need. But this method can be used for more advanced work. We can implement a chatbot facility for COVID-19 information with the same principal. Information are unknown to most of the people about the current pandemic. A chatbot can make them aware by answering several unknown queries about COVID-19. We can make health check up protocol using chatbot, where a patient can analyse his/her symptoms by answering several questions. This proposed technique can be used by Government to survey for the COVID-19 cases. We need to develop a database where 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.

REFERENCES

[1] Aafiya Shaikh, Dipti More, Ruchika Puttoo, Sayli Shrivastav, Swati Shinde. “A Survey Paper on Chatbots”. International Research Journal of Engineering and Technology (IRJET). Volume: 06 Issue: 04 | Apr 2019; p-ISSN: 2395-0072.

[2] Augello, A., Saccone, G., Gaglio, S., Pilato, G.: Humorist bot: bringing computational humour in a chat-bot system. In: International Conference on Complex, Intelligent and Software Intensive Systems. CISIS 2008, pp. 703–708. IEEE (2008)

[3] Balahur, A., Hermida, J.M.: Affect detection from social contexts using commonsense knowledge representations. In: 2012 International Conference on Privacy, Security, Risk and Trust (PASSAT) and 2012 International Conference on Social Computing (SocialCom), pp. 884–892. IEEE (2012)

[4] Balahur, A., Hermida, J.M., Montoyo, A.: Detecting implicit expressions of sentiment in text based on commonsense knowledge. In: Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis, pp. 53–60. Association for Computational Linguistics (2011)

[5] Moldovan, D.I., Rus, V.: Logic form transformation of wordnet and its applicability to question answering. In: Proceedings of the 39th Annual Meeting on Association for Computational Linguistics, pp. 402–409. Association for Computational Linguistics (2001)

[6] Tapeh, A.G., Rahgozar, M.: A knowledge-based question answering system for b2c ecommerce. Knowledge-Based Systems 21(8), 946–950 (2008).

[7] Lab, M.A.: Start (2002). http://www.ai.mit.edu/projects/infolab/start.html.

[8] Microdoft: Encarta (2005). http://encarla.msn.com/.

[9] Katz, B., Lin, J.J., Loreto, D., Hildebrandt, W., Bilotti, M.W., Felshin, S., Fernandes, A., Marton, G., Mora, F.: Integrating web-based and corpus-based techniques for question answering. In: TREC, pp. 426–435 (2003).

[10] Lin, J., Katz, B.: Question answering from the web using knowledge annotation and knowledge mining techniques. In: Proceedings of the twelfth international conference on Information and knowledge management, pp. 116–123. ACM (2003).

[11] Burke, R.D., Hammond, K.J., Kulyukin, V., Lytinen, S.L., Tomuro, N., Schoenberg, S.: Question answering from frequently asked question
files: Experiences with the faq finder system. AI magazine 18(2), 57 (1997).
[12] Wallace, R.: Alice (2009). http://www.alicebot.org.
[13] Wang, X., Khoo, E.T., Fu, C.R., Cheok, A.D., Nakatsu, R.: Confucius chat: promoting traditional chinese culture and enhancing intergenerational communication through a chat system. In: 2013 International Conference on Culture and Computing (Culture Computing), pp. 123–128. IEEE (2013).
[14] Weizenbaum, J.: Eliza computer program for the study of natural language communication between man and machine. Communications of the ACM 9(1), 36–45 (1966).
[15] Jing Huang, Qi Li, Yuanyuan Xue, Taoran Cheng, Shuangqing Xu, Jia Jia, Ling Feng. “TeenChat: A Chatterbot System for Sensing and Releasing Adolescents’ Stress”. International Conference on Health Information Science. LNCS, volume 9085, HIS 2015: Health Information Science pp 133-145.