Simple Chatbot using NLTK library in python :Based on Retrieval Model

By *Aashima Kundu,Akshat Puri,Aditi Suman*
Under the guidance of
Mr. Dinesh Kumar
SRM Institute of Technology,
Ghaziabad
(E-mail: aashimakundu1997@gmail.com)

Abstract— This paper introduces the idea that how a chatbot can be used to provide correct information about specific technical subjects and skills required in various technical fields. The chatbot created will be able to solve student's problem in the areas of all the possible technical fields,this chatbot can answer what all the technical skills are required to be at particular technical designation. For example,if a student wants to be a database administrator ,he can use chatbot to find out all the details of all the subjects and skills required to be the same .This project came when as students we faced the lack of correct information about any specific field.
This project aims to provide the accurate information about each technical field,contributing in the success of the students as gaining accurate information about the carrer is the first step towards success.
It will guide the students about the specific subjects or languages on which they should focus on in order to be at a specific technical postion in their carrer.

Keywords—*chatbot; technical fields; ;technical skills*

I. INTRODUCTION

Chatbots are created with an aim such that they can be used to converse the same way as a human would do and provide a personal assistance. This paper shows how a chatbot can act as a personal assistant and answer the queries of the students ,as a human guide would do.
A chatbot can also be called a talkbot, interactive agent, chatterbot or simply a bot. It is a conversational agent which conducts a conversation using natural language [1].This paper introduces the idea of making a simple Chatbot which is based on retrieval based model . Retrieval based model uses a predefined knowledge to generate outputs to the user's query and generates the best matching answer as a response.

The user inputs the query and then the query is analyzed and the best matching output is found out through machine learning algorithm and the output is displayed to the user as a response.All the data ,in this case - technical subjects and skills required to be at a specific technical designation are stored in a text file which is called raw data. The raw data is then processed by the machine learning .The Chabot will come up with outputs on the basis of pre-defined datasets(pre-defined knowledge) . In case user's query does not have any response, the chatbot outputs –"Sry, I don't have any data related to your query ". It is very important for our chatbot to be user-friendly. So,in order to develop the whole efficiently,it's implementation is divided into several modules.

II. THE WHOLE SYSTEM IS DIVIDED INTO MODULES

A. *Module 1: Data collection and organizing*

Under module one , collection of data and organizing takes place. In order to meet the user requirements and provide user with accurate information ,collecting correct data is important .The chatbot which we aim to design works on pre-defined datasets / knowledge and
hence, the outputs would be fetched from the data source which makes it even more important to collect accurate data.Accurate data about skills (both technical and soft) required to be at a particular technical designation is collected .It's organized accordingly to make it easy for the algorithm to match patters in order to get accurate output, chatbot uses machine learning to analyze the query and give out the best matching answer to the question.

B. *Module 2 : Pattern matching algorithm*

Under module two, pattern matching algorithm is designed and code is written for the same .Pattern matching algorithm is designed which makes chatbot efficient to carry out the task and outputs the correct result .The data collected in the first module is set as the working directory for the algorithm to work . . In the case that the query does not have a response, the chatbot outputs -"Sry! I don't have any related data to your query".

C. *Module 3* : *Graphical User Interface Design (Front-end)*

Under module 3, inputs and outputs for the front-end of the system are determined (for both admin and user). Graphical user interface for the system is designed in order to make the system user friendly ,this is done with the help of :

- data flow diagrams and Use case diagrams

After designing the front - end , coding is written for the same which should match the user requirements and results an efficient workflow in the activites of the system . In order to develop front-end tkinter library in python is used.

D. Module 4 : Database design to store user information

Under 4th module , database to store the information about the users is designed .Required table is constructed which stores the name , email-id and password **o**f the user.In order to secure user information and maintain their privacy ,each user has it'sown username and password which are kept confidential and can only be accessed by admin. User can access their account by passing authentication process and can enjoy chatbot interface.

III. PATTERN MATCHING ALGORITHM IN SIMPLE STEPS

A. The very first step is to create a file which contains all the data about the subjects and skills required for a specific technical designation .The file simply can be a text file, say - "data.txt" and this must be marked as the working directory of the program .
B. The next step would be to import all the required libraries using import keyword.. Libraries like nltk ,numpy ,random and string must be imported .
C. After marking the file as the working directory of the program , the data in the file is read using open function.
D. The whole file is converted in the lowercase using *.lower()* function ,so the algorithm doesn't treat a same alphabet in lower and upper case ,differently.
E. The raw data is further processed by diving the whole data into sentences and then into words ,this is called tokenization.Sentence tokenization is done using a function named ".*sent_tokenize*" in *nltk* library. The following picture shows sentence tokenization.
F. Word tokenization is done using a function named ".*word_tokenize*" in *nltk* library which divides sentence into words and each word is strored with a different index. The next picture shows word tokenization.
G. A function is created,which will return normalized tokens after it inputs the tokens(sentences divided into words and stored in the form of strings),it is called lemmatization.
H. *TFidfverctorizer* module is imported in order to convert the raw document into a matrix of TF-IDF features
I. Cosine similarity module is imported from scikit learn library which will search for similar patterns for the ouput from the pre-defined data.
J. A function say**, *analyze_query*** is created which searches the best matched answer and displays it. If in case no data or matching answer is found regading the query ,following text is retured to the user:*"Sry! I don't have any data regarding your query"*.

IV. INTEGRATING THE MODULES

All the modules are designed separately and has been tested . Integration of the modules takes place in order to make the whole system to work together.

A. Module 1 and 2 are integrated

Module 1 which was about the data collection is now set as the working directory for the algorithm designed under module 2.

Under this the workflow of the algorithm is tested and the efficiency of the algorithm to determine the accurate answers using the pre-defined data is tested.

B. Module 3 and 4 are integrated

The front-end designed for the system is integrated with the database (back-end) created to store the information about the users .

C. The algorithm and graphical user inertface is integrated .

All the modules are integrated to make a whole system.

V. TESTING

Testing of the software ensures the efficiency of the workflow in the whole system. In the process of testing , program is compiled no. of times in order to remove bugs and insures that the application runs fine and provide specific output for specific input. It determines the following things about the system.It tests whether the application developed meet the user's requirements and whether there is a smooth workflow of activities in the system.It also ensures the correctness and efficieny of the data flow in the whole system.

VI. Figures and Tables

A. Table: USER_INFO

User_name	Email_id	Password
Ram	abc@gmail.com	12345
Harry	def1159@gmail.com	abcdef

Fig. 1 table contains information about the users

B. Data flow diagram

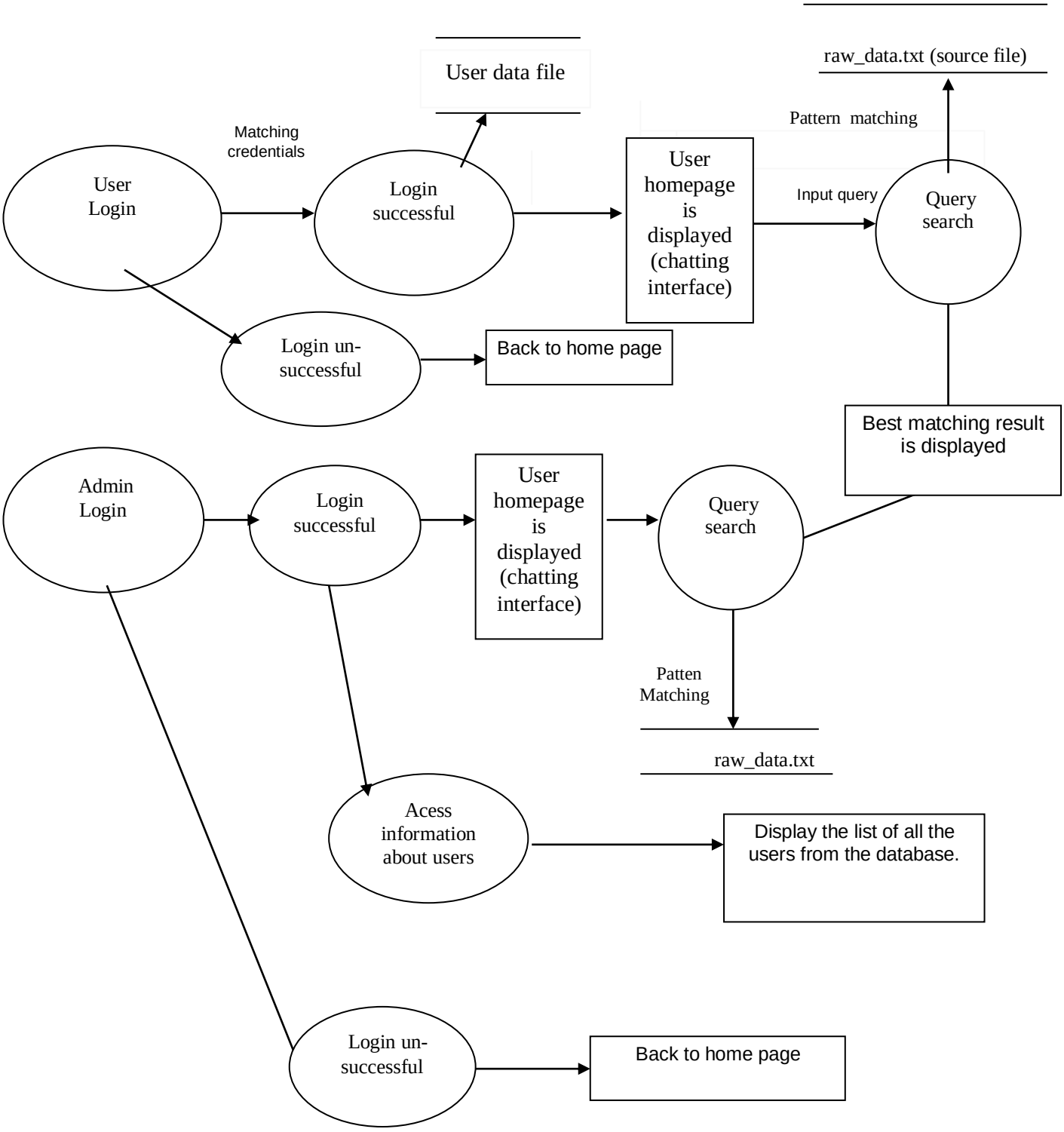

Fig 2. Data flow diagram of the system

C. Architectural diagram

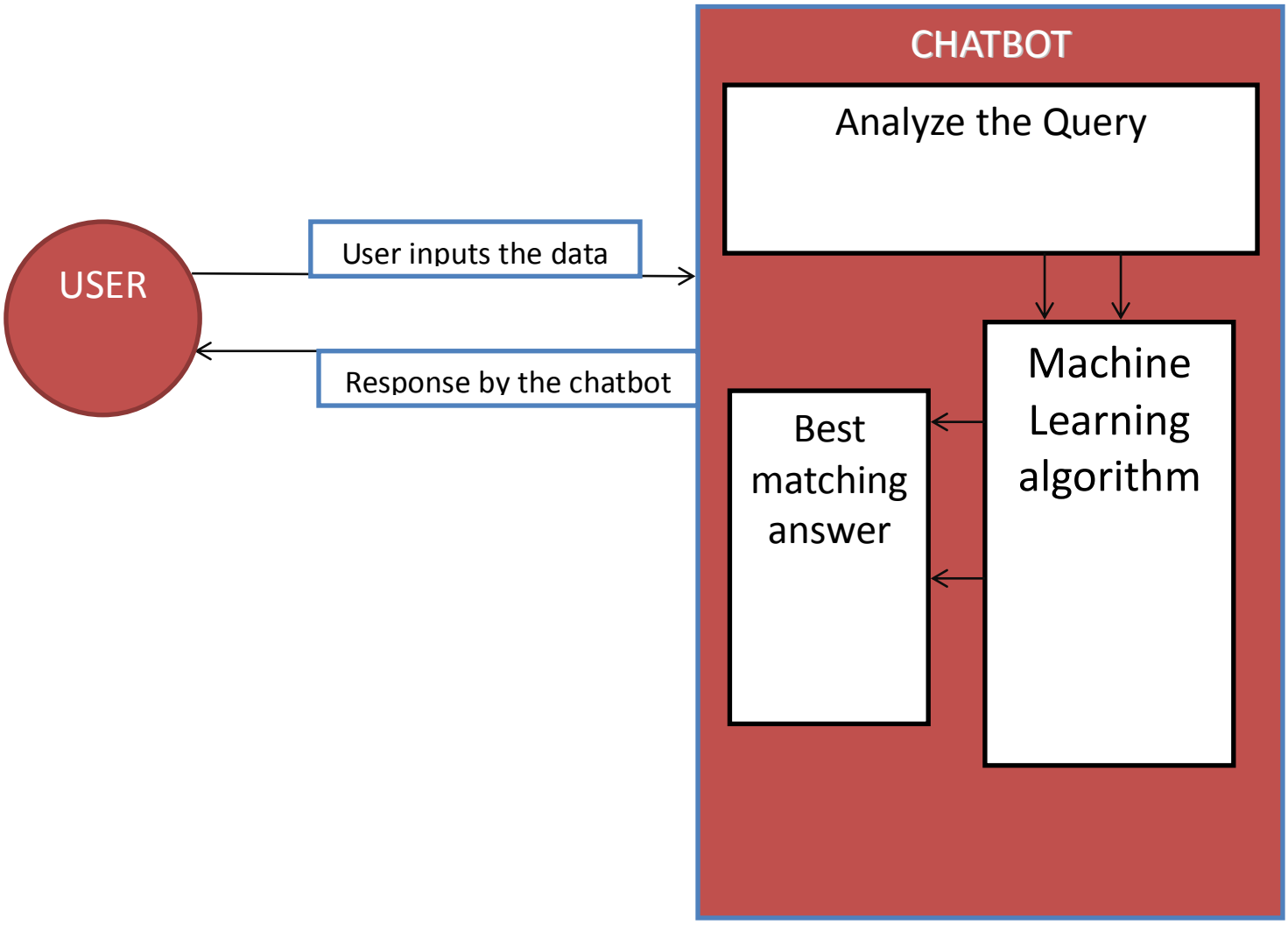

Fig3.Architectural diagram of the chatbot

ACKNOWLEDGMENT

We feel an immense amount of pleasure in submitting this research paper as partial fulfillment for the award of the Degree of B.Tech in Information technology. The satisfaction and euphoria that accompany the successful completion of any project would be incomplete without a mention of people who made it possible and whose constant guidance and encouragement crown all the efforts.We would like to express my sincere gratitude to my guide, **Dr. Dinesh Kumar** for his invaluable guidance and cooperation throughout the work.

REFERENCES

[1] CHATBOT (En.wikipedia.org, 2017) En.wikipedia.org. (2017).Chatbot.[online] Available at: https://en.wikipedia.org/wiki/Chatbot [Accessed 06 Dec. 2017].

[2] Kerly, P. Hall and S. Bull, "Bringing chatbots into education: Towards natural language negotiation of open learner models", Knowledge-Based Systems, vol. 20, no. 2, pp. 177-185, 2007.

[3] Luciana Benotti, María Cecilia Martínez, Fernando Schapachnik, "Engaging High School Students Using Chatbots".

[4] T.D Orin, "IMPLEMENTATION OF A BANGLA CHATBOT", Department of Computer Science and Engineering BRAC University, pp. 15-18, 2017.

Aashima Kundu ,a student at SRM IST, Delhi-Ncr campus. Pursuing B.TECH.(Information Technology).