

## **College Chat-bot**

Rachit Kulkarni, Ankit Methwani, Nakul Pawar , Charmi Valecha  
Project head- Pooja Shetty

### **Abstract**

The College chat-bot project is built using artificial intelligence algorithms that analyses user's queries and understand user's message. This System is a web application which provides answer to the query of the student. Students just have to query through the bot which is used for chatting. Students can chat using any format there is no specific format the user has to follow. The System uses built in artificial intelligence to answer the query. The system provides appropriate answers as per user queries. The User can query any college related activities through the system. The user does not have to personally go to the college for enquiry. The System analyses the question and then answers to the user. The system answers to the query as if it is answered in person. 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. After logging in the user has access to the various helping pages. The user can query college related activities such as date and timing of annual day, sports day, and other cultural activities. The system replies to the user with the help of effective graphical user interface. The user can query about the college related activities online with the help of this web application. This system helps the student to be updated about the college activities.

### **1 Introduction**

The project-Botman is implemented on following three platforms. The purpose of implementing on these platform was to extend the reachability of the bot

- Android
- Java
- Web

#### **1.1 Technologies**

- **API.AI**

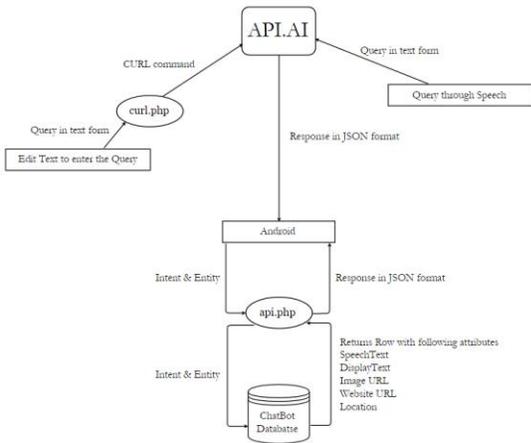
Api.ai is a voice-enabling engine that allows the addition of voice interfaces to apps based on Android, iOS, HTML5, and Cordova to third party developers. The SDK's contain features like voice recognition, natural language understanding, and text-to-speech. It also offers a web interface to build and test conversation scenarios. The platform is based on a natural language processing engine which allows Internet of Things(IoT) developers to include natural language voice interfaces in their products.

- **Android**

Android is a mobile operating system developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. Android's user interface is mainly based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input.

- **Web Technologies**

There are many Web technologies, from simple to complex, and explaining each in detail is beyond the scope of this article. However, to help you get started with developing your own Web sites, beyond simple WYSIWYG designing of Web pages in FrontPage, this article provides brief definitions of the major Web technologies along with links to sites where you can find more information, tutorials, and reference documentation.



## 2 Overview

### 2.1 Question-Answering

When questions such as “Who is the principal of college?” and “What is the time table of FE?” are posed to a question-answering system, the program should return the answers it finds in documents rather than just returning a link to a document that may contain the answer as search engines do. There are a few different systems available to try online, and research is still incredibly active.

### 2.2 Interactive Question-Answering

Interactive question-answering systems turn the focus to the interaction between the user and the program, not just the question-answering. These systems allow either the user to drive the dialogue or the system to play a greater role by suggesting related materials or even refinements to a users query. An interactive layer in a QA system also allows for more complex queries to be issued by the user as mistakes made by the system can be rectified through user input; or the system can engage the user to check its understanding of a users query is correct.

### 2.3 Chat-bots

When humans converse we generally do so by taking turns. These pairs of turns are known as adjacency pairs and this is also how chat-bots engage in dialogue with their human users. This

natural conversational method applies particularly well to QA systems due to the fact that when they are queried by a user they should return some meaningful result to continue the conversation. This is not to say that the system must know the answer to a query but rather by returning an answer that bears little relevance to the users utterance.

### **3 Project Aims**

The projects aim is to create a chat-bot based interactive question-answering system capable of pronominal anaphora in a user-driven dialogue. The intention is that a user will be able to collect data on a given subject faster. To this end the following are requirements:

- A clean interface showing the dialogue.
- Answer a query posed in the users natural language (The system is restricted to English).
- Rewrite follow up queries for the answering module.
- Engage the user without cluttering the dialogue.
- Do not include extraneous data in an answer.

### **4 Requirements**

#### **4.1. Functional Requirements**

In order for the program to perform well in its role of a question-answering system with the ability to perform anaphora resolution and query rewriting the following specifications must be met:

##### **Correctly identify elliptic queries**

The system must identify an elliptical question. This is necessary in order to ensure the query is not sent to the answering service. Elliptic queries are identified but they are not resolved in IQABOT.

##### **Interface with existing question-answering service**

Use existing question-answering services in order to maintain the systems status as a chat-bot like interface. This will allow the system to be used with several different QA platforms.

#### **4.2 Non-Functional Requirements**

##### **Simple user-interface**

Cha-tbots generally have very simple interfaces; showing only what is necessary to continue the flow of a conversation. With this in mind we will use a very simple command-line interface which will display previous queries and answers as adjacency pairs.

##### **Timely response**

The system should return answers in a few seconds, rather than minutes.

##### **Techniques studied**

The designer must be familiar with a number of techniques:

- 1) Parsing: this technique includes manipulating it by using a number of NLP functions; for example, trees in Python NLTK.

- 2) Pattern matching: it is the technique that is used in most Chat-bots and it is quite common in question-answer systems depending on matching types, such as natural language enquiries, simple statements, or semantic meaning of enquiries.
- 3) AIML: it is one of the core techniques that are used in common Chat-bot design.
- 4) Chat Script: is the technique that helps when no matches occur in AIML. It concentrates on the best syntax to build a sensible default answer. It gives a set of functionalities such as variable concepts, facts, and logical and/or.
- 5) SQL and relational database: is a technique used recently in Chat-bot design in order to make the Chat-bot remember previous conversation.
- 6) Markov Chain: is used in Chat-bots to build responses that are more applicable probabilistically and, consequently, are more correct.
- 7) Language tricks: these are sentences, phrases, or even paragraphs available in Chatbots in order to add variety to the knowledge base and make it more convincing.

## **5. LITERATURE REVIEW**

The purpose of a chat-bot system is to simulate a human conversation; the chat-bot architecture integrates a language model and computational algorithm to emulate information chat communication between a human user and a computer using natural language. With the improvement of data-mining and machine-learning techniques, better decision-making capabilities, availability of corpora, robust linguistic annotations/processing tools standards like XML and its applications, chat-bot have become more practical in daily life applications such as help desk tools, information retrieval tools, automatic telephone answering systems, advertising, tools to aid in education, business and E-commerce. In E-commerce, chat bot helps in information retrieval tasks, such as for searching and browsing, as menu based navigation poses difficulties in locating the appropriate information. The dialogue system provides additional information on products and simplify decision making process to find a product that satisfy customer's requirements. The study focused more on user attitudes rather than on chat-bot efficiency.

### **References**

[http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7072837&queryText=chat%20bot&ranges=2013\\_2016\\_Year](http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7072837&queryText=chat%20bot&ranges=2013_2016_Year)

[http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6569323&queryText=chat%20bot&ranges=2013\\_2016\\_Year](http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6569323&queryText=chat%20bot&ranges=2013_2016_Year)

[https://www.chatbots.org/images/uploads/research\\_papers/K.KULIGOWSKA\\_Commercial\\_chat\\_bot-Warsaw\\_University\\_Poland.pdf](https://www.chatbots.org/images/uploads/research_papers/K.KULIGOWSKA_Commercial_chat_bot-Warsaw_University_Poland.pdf)