A Study on Building a Web based Chatbot from Scratch

Dr. N Naveen Kumar[1], T Krishna Karthik[2], B Prasanna[3], Sai Kiran Reddy[4], A Siddharth[5]

[1]Associate Professor, IT Department, JBIET
[2,3,4,5]Under Graduate, IT Department, JBIET

---***---

Abstract—The popularity of Chatbots are increasing tremendously day by day. They are being used in many fields like   Entertainment, News, Health, Customer Service, Real Estate, E-Commerce. Chatbots are intermediary interactive software applications which can emulate human conversation when conversed with them. They make us believe that there is a human on the other side of conversation. This is what it is pushing everyone to sneak Chatbots in their applications. Chatbots receive requests from users and give them appropriate response with whatever knowledge it has, understanding the intent of the user. This may involve using computer science concepts like NLP (Natural Language Processing), Deep Learning, Machine Learning, Artificial Intelligence. This paper presents a study that has been conducted on how to build a chatbot from its scratch.

Keywords—*Chatbots, Technologies, Methods, Computer Science, Natural Language*

1. INTRODUCTION

Chatbots are successful because they give users exactly what they want. The bot is intelligent enough to understand the user's intent, retrieve and present what is needed when user makes a request in his own language or in what is known in Computer Science as Natural Language. They are virtual personal assistant and are user friendly. Chatbots improve service provided to the users in an application. Chatbots help users to obtain timely and efficient assistance or information. In his book [1], Sumit Raj mentions steps that should be followed during building a chatbot. The three main steps which needed to be followed to build a new chatbot from scratch according to Sumit Raj in his book [1] are,

- Analysing and thinking about the scope of chatbot's knowledge and collect different questions that chatbot can face.
  Every task that chatbot do will define an intent.
- Each question that we list or intend can be represented in different ways.
  Example: What is rate of jeans pant?
    How much does a jeans pant cost?
    What cost is a jeans pant?
    What is the worth of a jeans pant?
    What is the price of a jeans pant?

In all the above cases the intent of the users is to know the cost of a jeans pant. The way of presenting the question is different, but the intension of the questions is same.

- Prepare the logic so that bot can give precise response after recognizing the intent of the user. In the above-mentioned example, the questions are of same context, but the answers to these all questions is only one. So, a logic should be written so that bot identifies the context or intent precisely and give the answer infallibly.

II. METHODS AND MATERIAL

A. Choosing type of Chatbot you want to build:

Depending on the usage, knowledge scope, technology used chatbots have been categorized in following ways [2]:

i. Based on the scope of knowledge bot possess, we have Open Domain and Closed Domain.

Open Domain: Open Domain Chatbots can be asked with anything a user want to ask without any restriction on topic or subject of the conversation. Open Domain Chatbot's knowledge base is so big that they are expected to respond sensibly to anything that a user converse. Indirectly, Open Domain Chatbots behave like search engines. Example for Open Domain Chatbots are some of the personal assistants like Google Assistant, Siri (Apple), Cortana (Windows) that are conversational agents try to answer every task they receive, at least by providing corresponding internet search results.

Closed Domain: Closed Domain Chatbots are Domain-Specific Chatbots, whose knowledge base and scope are restricted to certain topic or subject and certain tasks as well. So, user has to contain himself when conversing with a Closed Domain Chatbots to that extent that the tasks or requests that he post to chatbot are not out of domain. But developers tend to reserve responses to the requests that are out of domain so as to make the chatbot look like more natural and human like. Examples for Closed Domain Chatbots are found now days in some of business and E-Commerce websites where, they tend to answer questions posted by the users regarding the services that are provided to them. So, for user it is key to recognise these bots and restrict them from having unnecessary conversation.

ii. Retrieval Based and Generative Based Chatbot classification is mainly of how the backend of the bot creates response.

Retrieval Based Chatbot: Retrieval Based Chatbots uses predefined repository of responses stored by developers, expecting and guessing the context of requests that users may ask. In Retrieval Based Chatbots the intent of the users may be same. But the way they post the request, sentence formation and structure, may be different. It is the work of the bot to recognize the intent and give the most relevant response, which is predefined indexed for that while developing. We identify the most relevant response results by using a Deep Learning Algorithm called Word2Vec as mentioned in [3].

Generative Based Chatbot: Generative Based Chatbots generates responses on their own, then and there and also not predefined. Here Bot understands the intent or context and generates answer on its own. According to [4] Generative Chatbots have edge over Retrieval Based Chatbots as they respond to user accordingly rather than giving same answer to all the questions of same context. This make them more realistic. Generally Deep Learning Algorithms are used to develop Generative Chatbots. But they are difficult to build as they have to be trained to give responses that are not only precise contextually, but also grammatically. In [3] there is a Generative model mentioned, uses LSTM as a training model for generative dialogue.

B. Technologies:

According to [5] for implementing Chatbots as a web application, different programming languages are used be it developing front end or backend.

(i) Front End Development:

The programming languages that are used here are HTML5, PHP, CSS, JS, Bootstrap.

PHP: PHP is being used extensively as it is an open source for general purpose scripting language that is especially used for developing a web application. PHP pages has HTML with embedded code that can do "something". What is different to PHP from websites using client-side JavaScript is that the code is executed directly on the server, generating the HTML code which is sent to the client. He would get the final results after running that script, but he will not know what the underlying code was. You can process all your HTML files just by configuring the web server to do that and then there is no chance that users can tell what you have up your sleeve.

HTML: HTML is Hyper Text Markup Language, which extensively used to develop simple Websites with GUI (Graphical User Interface) providing user with different elements like Button, Textbox, Textfield, Checkboxes, Radio Buttons. Here language is in the form of tags.

Multi-media contents like Videos, Images can be handled using HTML.

CSS: CSS is Cascading Style Sheets is a style sheet language which is used to add more look and appearance a HTML page. A tag <style> is used and the style that developer wants in his HTML page in that place of the page is written in CSS.

JavaSript: JavaScript is used to handle behaviours of different elements in an HTML page. It is an Object Oriented, prototype-based language. Javascript handles complex tasks. It enables you to create dynamically updating content, control multimedia, animate images, and pretty much everything else.

Bootstarp: Bootstrap is essentially a stylsheet. There are bootstrap components that require a JavaScript file but for the most part it is a collection of CSS styles that make building a responsive site easier. It includes some prepackaged styles for creating screen elements, styling tables, buttons, forms etc - but still just a .CSS file.

(ii) Back End Development:

The Programming Languages used to develop Back End are, Python3, MySQL Databases.

Python3: Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms. It has very extensive Library. Here we use Python's Standard Library to implement Deep Learning Algorithms, which are crucial for Chatbot performance.

MySQL Databases: MySQL is a powerful open source database server built based on a relational database management system (RDBMS) and is capable of handling a large concurrent database connection. MySQL is capable of working with different programming languages like PHP, PERL, C++, JAVA, Python etc. MySQL works very quickly and works well even with large data sets. MySQL is very friendly to PHP, the most appreciated language for web development. We can insert, retrieve, alter, delete data easily using queries.

(iii) Ajax and JSON:

Ajax is the abbreviation for Asynchronous JavaScript and XML. It is a good technique to code with for the building of interactive websites applications. The intention of it is that the websites become faster and more acceptable by changing in background some small information with the server. This means that the user should not refresh the web page every time he did something. Ajax has the goal

to grow the interactivity, to make them faster and to use easily the websites. Ajax is not a technology by itself. The term is used to define the websites that use a set of technologies. Ajax is used for the transfer of data between the server and the database to the front end.

JSON is the abbreviation for JavaScript Object Notation that it is a format for representing and interchanging the data between informatics applications. It has people text format, used to represent objects and other structures of data and it is used mostly to send structured data through the network, and the process is called serialization. JSON is the simple and easier alternative of XML coding language. The elegance format of JSON comes from the fact that it is a subset of the JavaScript, being used along to this coding language.

III.   CONCLUSION

With the help of information provided above, a user can build his own chatbot according to his convenience. As mention in Methods and Materials, Requests to Chatbot should come in either under Open or Closed Domain. Responses should come in either under Retrieval or Generative based models. As mentioned in Technologies, we can use above languages to develop a good interactive interface which can be user friendly.

IV.   REFERENCES:

1. Sumit Raj. Building Chatbot with Python, Apress Publications.
2. Nitirajsingh Sandu, Ergun Gide. Adoption of AI-Chatbots to Enhance Student Learning Experience in Higher Education in India.
3. Min-Yuh Day, Chi-Sheng Hung. AI Affective Conversational Robot with Hybrid Generative-based and Retrieval-based Dialogue Models.
4. Enza Varghese, Prof. M T Rajappan Pillai. A STANDALONE GENERATIVE CONVERSATIONAL INTERFACE USING DEEP LEARNING.
5. Petre Anghelescu, Stefan Vladimir Nicolaescu. Chatbot Application using Search Engines and Teaching Methods.