**Chatbots beyond business: A comprehensive study with implementation using an educational institution as a use case**

**1Jayapraveen AR**

1 Rajalakshmi Engineering college

Email: 1 jayapraveen.ar.2015.cse@rajalakshmi.edu.in

Contact: 1+91-9042616330

**Abstract:** These days, conversational agents present a new way for individuals to interact with the computer systems. The Chatbot is a conversational agent that can chat like a human being and they can learn from their experience through interactions. From the early stage of rule-based chatbots to the era of speedy development in Artificial Intelligence (AI), the performance of chatbots keeps improving. Natural Language Processing (NLP) is the core technology in the rise of these chatbots. The Chatbot can also act as an onboarding assistant. The main goal of this paper is to study, design and build a chatbot for an Educational Institution. The chatbot will be personalized to guide the visitors of an educational institute and help them to find the information what they are searching for without the need for going through the entire website. This chatbot can answer to related queries and help people self-serve information.

***Index******terms:*** Chatbot, NLP, Conversational Agents.

1. **INTRODUCTION**

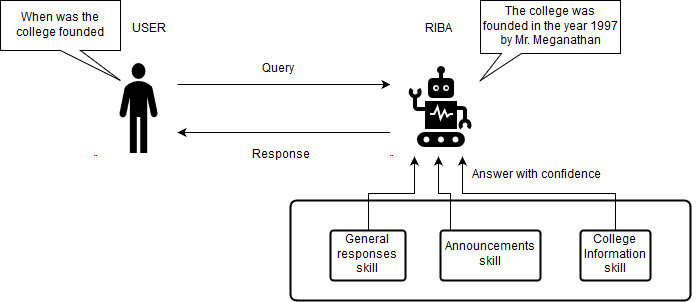
Commercial implementations of chatbots are mostly in the sense to drive sales and reduce human labor in answering repetitive questions. We would like to see if chatbots can be made useful in other sectors where in person to person is still a effective practice.

In our work, we came up with RIBA – Rajalakshmi Institutions Bot agent, a chatbot that can serve knowledge about our college when asked. RIBA’s skills are not just limited but can be expandable. The approach we followed to build RIBA is a modular design kind of interaction where skills can be added and removed based on the need. Usually chatbots are based on a machine learning algorithm where it is trained with a large amount of corpus. In our use case we found that it alone won’t be as efficient as it would be in a conventional use case such as Question and answering bot, as the information used to serve the students in colleges may vary over time. Hence, we came with the modular approach. The Implementation of chatbot in a place like college involves more understanding of the users of certain personas. Here we could determine the characteristics of the users as they belong to specific age groups and their behaviors could be more alike, which is beneficial while designing the chatbot as we could make it more realistic to the users.

1. **System design**

The system overview of RIBA is given in FIGURE 1. When a query is asked by the user RIBA forwards the query to various skills in the system Each skill based on the implementations use various algorithms such as searching, classification to determine the closest answer and return the answer with a confidence value. The answer with the highest confidence value is chosen as the answer and is returned to the user. Sometimes based on the need a particular skill can be used by biasing the confidence value such that for a particular query a particular skill’s answer will be showed to the user.

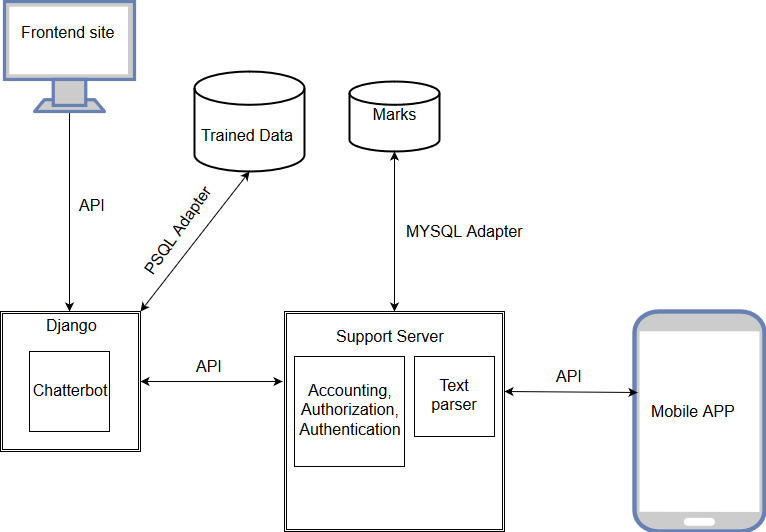
The confidence value is determined by the Naïve Bayes algorithm, which is an eager learning algorithm so it can be used for real-time prediction.



**Figure 1: RIBA interaction diagram**.

1. **Architecture of the proposed system**

The architecture of RIBA is given as a diagram in FIGURE 2. The chatbot uses a modified version open source python library for NLP, Training and replying to queries called Chatterbot. The library Chatterbot is modified to interact with the web. The chatterbot is connected to the internet through django ORM. A REST API is built using Django to accept input from various devices. The chatterbot has PostgreSQL as its backend datastore. The chatbot stores its trained data in datastore. A front-end site is built and can send user’s query to The REST API and parse the output to display it to user. The mobile app also follows the same principle.



**Figure 2: Architecture Diagram**

1. **Innovative designs implemented in the project**

The structure of the Implementation is based on Scalability, security and redundancy.

The backend is hosted on HEROKU which is a SAAS Platform where users can create and deploy their own applications. The application can scale from 10 users to 1000 users in a second since the datastore is based on PostgreSQL several simultaneous connections can be held at a second. Since one of the critical skills of the app is to display marks, during results announcement the traffic might be very high at these times the application can automatically scale and handle the load with ease, unlike traditional SQL Databases.

The platform has a supporting parsing server which scans all the data incoming from the users which enforces special character and SQL queries escaping. So it can deter almost most of the so the attacks that can be done by the users. The database applies password salting and hashing in accordance to User Data Security policy before storing it to the database so even if the data is leaked it is almost impossible to crack the password by conventional methods. If the

Institution doesn’t trust third party vendors, the application can be deployed and managed on premise if required.

The datastores do not contain valuable assets like user’s data with respect to the chatbot. The training can be done whenever required and can be made changes whenever required. With respect to skills such as marks information the datastore will be maintained by the institution itself and the chatbot will only have read permission to the database. When in terms of scaling a copy can be maintained in order to improve caching and serve data faster.

1. **Implementation procedures**

The application is built with python 3.6 as its code base in the backend and is using stricter versions of libraries in order to ensure reputability in dependencies.

However, the proposed system can be improved when more skills are needed so the base chatbot module can interact with newer skills module by using logical adapters. A new skill can be introduced by using the logic adapters and declaring it in the settings config file.

The following commands are invoked to start the webserver for the chatbot

1. Sudo python manage.py migrate
2. Sudo python manage.py train
3. Sudo gunicorn riba.wsgi start

The following commands create new tables and update schemas in the database , train the bot and start the webserver.

1. **Results and analysis**

The bot as we designed with the users persona in mind and how it should react to various situations. No formal study on results has been done yet , as more improvements are needed for it to be fully working in a holistic environment.

**CONCLUSION**

The bot is a work to see how well a chatbot can perform and be useful in an educational institution. We could see the potential of these as many startups are being formed based on these and several companies are spending millions of dollars to make them more realistic and friendly to use.Chatbots can be a good source to fetch information quickly and process them. The purpose of its existence is not just to replace humans but to be a part of our being.

**REFERENCES**

[1] https://github.com/gunthercox/chatterbot

[2] https://github.com/gunthercox/chatterbot-corpus

[3] https://chatterbot.readthedocs.io

[4] https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/

[5] http://www.aclweb.org/anthology/P/P17/P17-4.pdf#page=107

[6] https://www.slideshare.net/cocolero/chatbot-and-ai-design-principles

[7] https://www.slideshare.net/cocolero/personas-bootcamp/40-40Using\_Personas\_eectivelyGoaldirectedDesign1\_Dening\_the

★★★