Ideation Phase

Defining the Problem Statements

Date	**26-09-2023**
Team ID	**3918**
Project Name	**Create a Chatbot Using Python**

Create a Chatbot Using Python

Problem Definition and Design Thinking

Introduction:

A chatbot is a computer software program that conducts a conversation via auditory or textual methods. This software is used to perform tasks such as quickly responding to users, informing them, helping to purchase products and providing better service to customers. Chatbots are programs that work on Artificial Intelligence (AI) & Machine Learning Platform. Chatbot has become more popular in business groups right now as it can reduce customer service costs and handles multiple users at a time. But yet to accomplish many tasks there is a need to make chatbots as efficient as possible. In this project, we provide the design of a chatbot, which provides a genuine and accurate answer for any query using Artificial Intelligence Markup Language (AIML) and Latent Semantic Analysis (LSA) with python platform.

Problem Statement:

Objective: Artificial intelligence chatbot is a technology that makes interactions between man and machines using natural language possible. A chatbot can give different responses from the same input given by the user according to the current conversation issue". By using our "Intelligent ChatBot" you can overcome all the above-given issues, you do not need humans to do manual work, your clients will be happy. A chatbot is a conventional agent that is capable to communicate with operators by using natural languages. As numerous chatbot platforms already exist, there are still some problems in building data-driven system because a huge amount of data is required for their development.

Key Challenges:

1. **Natural Language Understanding (NLU):** Chatbots must accurately interpret and understand user input, which can be complicated due to variations in language, slang, and context.
2. **Context Handling:** Maintaining context during multi-turn conversations is essential for providing meaningful responses.
3. **Personalization:** Striking a balance between personalization and privacy is crucial.
4. **Scalability:** Ensuring that the chatbot can handle a growing number of users and conversations simultaneously is essential for large-scale deployments.
5. **Integration with Backend Systems:** Chatbots often need to integrate with various data sources, APIs, and existing systems to provide accurate and relevant information.

Design Thinking Approach

Empathize:

To empathize, you need to do your research. Consult with experts (even if that's your customer support team), talk to your users, and immerse yourself in the environment. If you plan to make a Facebook Messenger bot (to add to the more than 100k already available), spend time speaking with your users via the app. Find out what their needs are. Seek out pain points.

Actions:

-Perform actions like sending emails, scheduling meetings, or controlling smart home devices

-Suggest movies, books, music, or other content based on user preferences.

-Provide empathy and emotional support through conversation.

Define:

Time to define the problem. You've got business goals you want to achieve, and you've got mounds of research from the time you spent empathizing with your users.

Objectives:

- Providing quick and efficient customer support by answering frequently asked questions, troubleshooting issues, and assisting with common tasks.

- Engaging users in conversations for entertainment, such as chatbots that tell jokes, play games.

Ideate:

To come up with ideas, throw them at the digital wall, and see what sticks. You know your users, and you've got a well-defined, human-centered problem. Let's ideate. Your goal here isn't to come up with The One Best Idea™. It's to generate as many ideas as possible. Think freely (or "outside the box," if you must). There are plenty of approaches to this part of the process like brainstorming, mind mapping.

Actions:

-Assist users in booking flights, hotels, restaurants, or appointments.

-Implement custom workflows and actions specific to the chatbot's purpose.

-Analyze user feedback and sentiment to improve the chatbot's performance.

Prototype:

The next step is to take some of those ideas and make them real. This is an experimental phase, and the goal is to identify the best possible solutions for the problems identified during the first three stages. Your design team will produce some low-fi, scaled-down versions of your chatbot (and its guiding logic) in an effort to find what works. Every idea that survived the transition into Prototyping will either be rejected (which is what will happen to most of them) or accepted, revised, and improved.

Actions:

- Provide answers to general knowledge questions or specific queries based on its training data.

- Offer support for troubleshooting issues, tracking orders, or handling common customer inquiries.

Test:

This is where all the hard work pays off. The Testing stage is where your designers, your researchers, and possibly even some of your users come together to test the more polished prototypes that were the results of your prototype.

Actions:

- Split the dataset into training and testing sets.

- Train the model on the training set and evaluate it on the testing set.

- Use metrics such as MAE, Root Mean Square Error (RMSE), and R-squared to assess model performance.

- Collect user feedback on the web interface for usability and accuracy.

Implement:

Once the prototype meets the defined objectives and receives positive feedback, proceed with full implementation.

Actions:

- Train the final machine learning model on the entire dataset.

- Deploy the model as part of a production-ready web application.

- Conduct thorough testing to ensure the application is robust and user-friendly.

Iterate:

Implement changes and improvements to the chatbot based on the prioritized list of tasks.

Actions:

- Monitor the model's performance and retrain it periodically with updated data.

- Address user feedback and make necessary improvements to the web interface.

- Stay informed about advancements in machine learning and real estate pricing models for potential enhancements.

Conclusion

In this document, we've outlined our approach to create a chatbot using python. The chatbot can answer only those questions which he has the answer in its AIML dataset. So, to increase the knowledge of the chatbot, we can add the APIs of Wikipedia, Weather Forecasting Department, Sports, News, Government and a lot more. In such cases, the user will be able to talk and interact with the chatbot in any kind of domain. Using APIs like Weather, Sports, News and Government Services, the chatbot will be able to answer the questions outside of its dataset and which are currently happening in the real world.