**<CHATBOT>**

**Submitted for**

**Statistical Machine Learning CSET211**

Submitted by:

**(E23CSEU1268) Titirsha Singh**

**(E23CSEU1265) Megha Singh**

**(E23CSEU1272) Koushiki Das**

Submitted to

**DR. Sanchali Das**

**July-Dec 2024**

**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

A close-up of a logo

Description automatically generated

**INDEX**

|  |  |  |
| --- | --- | --- |
| Sr.No | Content | Page No |
| 1 | Abstract |  |
| 2 | Introduction |  |
| 3 | Methadology |  |
| 4 | Hardware/Software required |  |
| 5 | Experimental result |  |
| 6 | Conclusion |  |
| 7 | Future Scope |  |
| 8 | Related work |  |
| 9 | Github Link |  |

ABSTRACT

This project presents a chatbot built using machine learning techniques for natural language processing (NLP) and deployed as a web application using Streamlit. The chatbot is trained on a dataset of conversation patterns and responses, enabling it to understand user queries and generate relevant responses. Users interact with the chatbot through a user-friendly interface on a web browser.

INTRODUCTION

Chatbots are computer programs designed to simulate conversation with human users. They are increasingly used in various applications like customer service, education, and entertainment. This project aims to develop a simple chatbot capable of understanding basic user queries and responding accordingly.

Related Work:

Having a unique dataset can be a significant advantage in your project. Here are some ways to highlight the unique contribution of your dataset:

**1. Novel Insights and Discoveries:**

* **Uncharted Territory:** Explore new insights and knowledge that wouldn't be possible with existing datasets.
* **Groundbreaking Research:** Contribute to groundbreaking research and scientific advancements.
* **Industry-Shaping Innovations:** Develop innovative solutions and products that address real-world problems.

**2. Improved Model Performance:**

* **Enhanced Accuracy:** Train more accurate and robust models by leveraging the unique characteristics of your dataset.
* **Better Generalization:** Improve the model's ability to generalize to unseen data.
* **Reduced Bias and Overfitting:** Mitigate potential biases and overfitting issues.

METHODOLOGY

**Data Preprocessing:**

* The code snippet shows downloading NLTK resources for tokenization and lemmatization.
* It loads the chatbot training data from a JSON file containing intents (user queries) and corresponding responses.
* Words are tokenized (broken down into individual words), converted to lowercase, and lemmatized (reduced to their base form) using the WordNetLemmatizer.
* A "bag-of-words" representation is created for each conversation pattern, which is a vector containing the presence or absence of each word in the vocabulary.

**Model Training :**

* You likely trained a machine learning model (e.g., a recurrent neural network) using the Keras library.
* The model is trained to predict the appropriate intent (user query category) based on the provided conversation pattern.

**Deployment using Streamlit:**

* The code demonstrates how Streamlit is used to create a web application interface for the chatbot.
* Users can type their messages into a text input field and submit them to the chatbot.
* The application processes the user message using the pre-trained model and generates a response based on the predicted intent.

Hardware/Software Required

* Programming language: Python
* Libraries: NLTK, Keras, Streamlit, NumPy (likely used for model training)
* Framework (optional): TensorFlow (likely used for model training)
* Additional tools: Streamlit (for web app deployment), ngrok (for creating a public URL)

Experimental result

**Training Progress:**

* **Epoch 200/200:** This indicates the model has completed 200 training epochs, which is a full cycle of training on the entire dataset.
* **189/189 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step:** This shows the progress within the final epoch (189 out of 189 training steps). Each step processes a batch of training data. The time per step (4ms) suggests relatively fast training.

**Model Performance:**

* **accuracy: 0.6065:** This is the training accuracy on the final epoch. An accuracy of 60.65% implies the model correctly classified ~61% of the training examples. This may not be ideal and could benefit from further training or hyperparameter tuning.
* **loss: 1.6277:** This is the training loss on the final epoch. Lower loss values generally indicate better model performance.

**Evaluation Accuracy:**

* **Accuracy: 97.88%:** This is likely the evaluation accuracy on a separate test dataset that wasn't used for training. A high accuracy of 97.88% suggests the model generalizes well and performs well on unseen data.

**Overall:**

While the training accuracy seems low, the high evaluation accuracy indicates the model is learning a good representation of the data and can perform well on new user queries.

Conclusion

This project successfully developed a chatbot application using machine learning and Streamlit for deployment.

The provided code demonstrates the data preprocessing and Streamlit application aspects.

In this dataset has limited vocabulary, inability to handle complex questions

Future Scope

* Improve the chatbot's accuracy and ability to handle more complex conversations.
* Integrate with external APIs or databases to provide additional functionalities.
* Explore different deployment options beyond Streamlit.

Related Work:

Having a unique dataset can be a significant advantage in your project. Here are some ways to highlight the unique contribution of your dataset:

**1. Novel Insights and Discoveries:**

* **Uncharted Territory:** Explore new insights and knowledge that wouldn't be possible with existing datasets.
* **Groundbreaking Research:** Contribute to groundbreaking research and scientific advancements.
* **Industry-Shaping Innovations:** Develop innovative solutions and products that address real-world problems.

**2. Improved Model Performance:**

* **Enhanced Accuracy:** Train more accurate and robust models by leveraging the unique characteristics of your dataset.
* **Better Generalization:** Improve the model's ability to generalize to unseen data.
* **Reduced Bias and Overfitting:** Mitigate potential biases and overfitting issues.