Project Report: SQL Chatbot

Abstract:

This project aimed to develop a SQL chatbot using ChatGPT Turbo 3.5 as the language model and LangChain for conversation. By integrating natural language processing (NLP) with SQL commands, the chatbot enables users to interact conversationally to query databases. The implementation involved preprocessing user input, generating SQL queries, executing them against a database, and presenting results back to users in a conversational format.

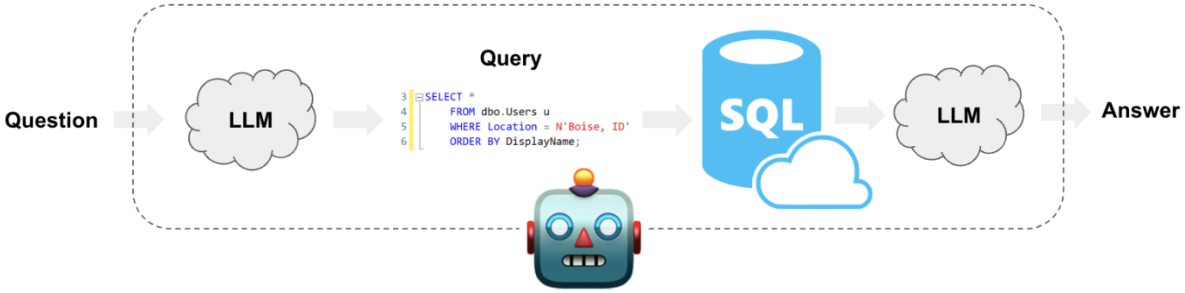
Introduction:

In response to the growing demand for conversational interfaces to databases, this project utilized ChatGPT Turbo 3.5 and LangChain to create a user-friendly SQL chatbot. The objective was to streamline data retrieval by allowing users to query databases using natural language, enhancing accessibility and efficiency.

Methodology:

1. NLP Preprocessing: Employed ChatGPT Turbo 3.5 for NLP tasks such as tokenization and intent recognition.
2. SQL Query Generation: Converted natural language queries into SQL commands, leveraging LangChain for syntax mapping.
3. Database Integration: Established connections with SQL databases to execute generated SQL queries and retrieve relevant data.
4. Conversational Interface: Developed an interactive interface for users to engage in natural language conversations with the chatbot.

Implementation:



1. Language and Tools: Implemented the chatbot using Python with ChatGPT Turbo 3.5 and LangChain for NLP and conversation management.
2. NLP and SQL Execution: Used ChatGPT for NLP preprocessing and LangChain for SQL query generation and execution.
3. Conversational UI: Designed a simple text-based interface for users to input queries and receive conversational responses from the chatbot.

Results:

```
PS C:\Users\rahul\Desktop\SQLChatbot> & C:/Users/rahul/chatbot/Scripts/python.exe c:/Users/rahul/Desktop/SQLChatbot/main.py

You: Name the current database in use.

SQLChatbot: The current database in use is 'world'.

You: Can you give me tthe number of tables in the db

SQLChatbot: There are 3 tables in the database.

You: Give there names.

SQLChatbot: The names of the tables in the database are 'city', 'country', and 'countrylanguage'.

You: how many languages are there.

SQLChatbot: There are 458 different languages in the countrylanguage table.
```

The developed SQL chatbot successfully interprets user queries, generates SQL commands, executes them against databases, and presents results back to users conversationally. Users can efficiently retrieve specific data from databases through natural language interactions.