SVKM'S NMIMS Deemed to be UNIVERSITY

The Cricket Times

By Vaibhav Khamitkar, Het Upadhyay , Masters in Data Science

Sometime in Future - RCB Raises the Cup: Cricket Analytics Gets a Nod, Players Get the Glory !

The victory was made possible by this research, that started it all.

Introduction

Cricket analytics began with basic stats in the late 20th century, evolving with the 1970s Player Performance Index. But it wasn't until the late 1990s, with better technology and data, that sophisticated models emerged, like the Duckworth-Lewis method. Today, machine learning delves deeper into vast data sets, predicting player performance and optimizing strategies. This data-driven approach, credited for England's World Cup win, highlights a gap between cricket experience and data analysis expertise. This paper proposes a solution to bridge this gap using advanced technologies, empowering cricket stakeholders to conduct their own data analysis for better decision-making.

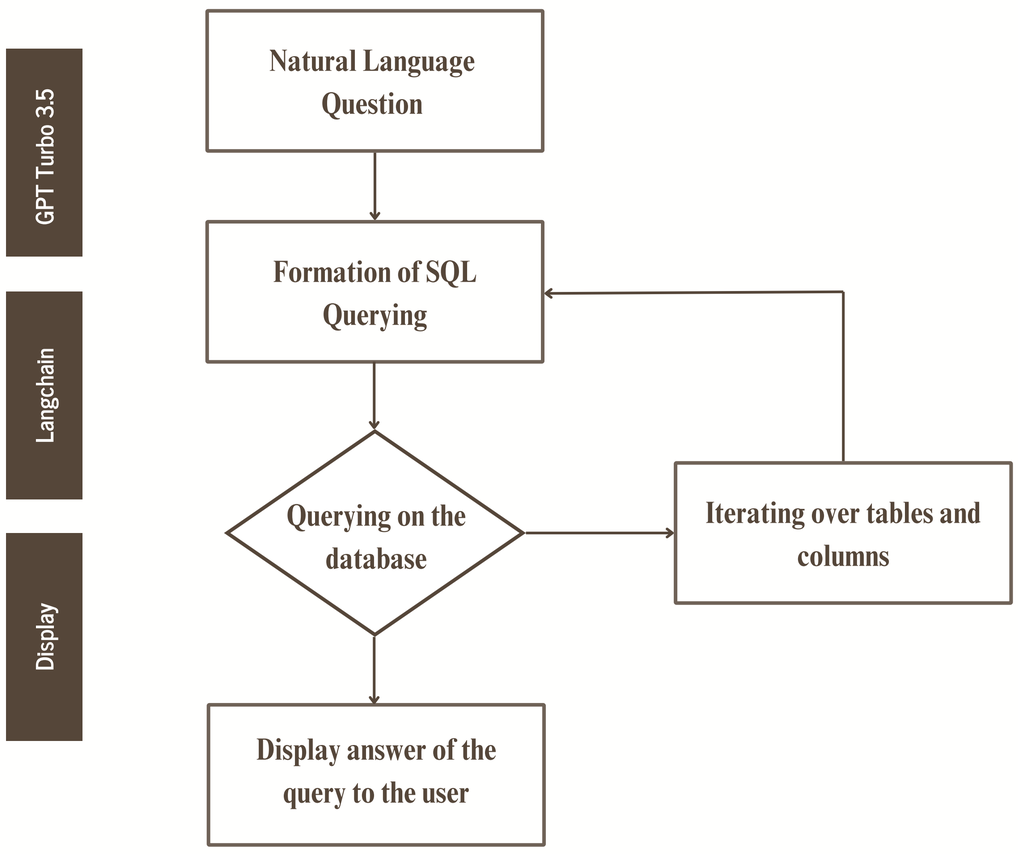
CricGPT: Use of LLM powered SQL agents to query cricket data

Abstract

Cricket, a global multi-billion dollar industry, has seen a transformation with the advent of analytics. Stakeholders increasingly rely on data analysis for performance enhancement and strategy development. Yet, there's a gap in utilizing experts' experience for independent analysis. To bridge this gap, we developed a chatbot powered by the Language Model (LLM) and SQL queries executed via an RDBMS. Our study assesses LLM's proficiency in handling queries, revealing strengths in simple queries but challenges with complexity.

Methodology

The Langchain system integrates natural language processing with SQL queries to facilitate user interaction with databases. Users formulate questions in plain English, which Langchain interprets using GPT Turbo 3.5. The Sql agent then translates these queries into structured SQL queries, retrieves data from the database, and generates natural language responses for the user.

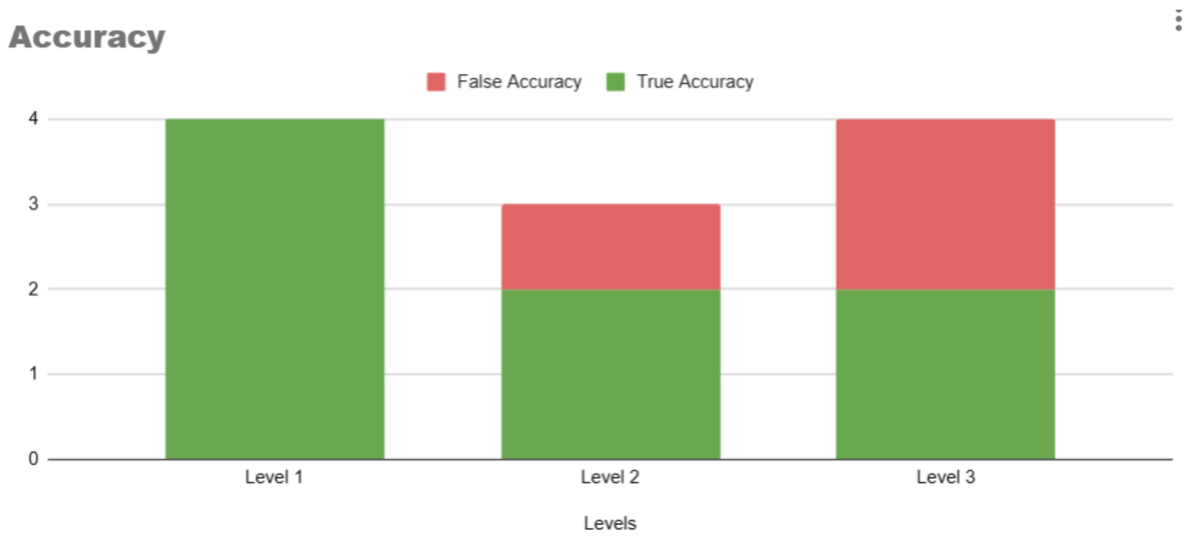


Technology used

LangChain · OpenAI · Microsoft SQL Server Management · Python · Streamlit

Results and Findings

The agent performs well with simple queries but exhibits mixed results with intermediate ones. However, it struggles significantly with complex queries, indicating a need for enhancement in handling intricate data manipulations.



Conclusion

The evaluation highlights the agent's proficiency in handling varying query complexities. It effectively executes simple queries, but struggles with complex aggregations due to limitations in SQL query formulation. While the agent demonstrates adaptability with intermediate queries, addressing these limitations is crucial for enhancing its effectiveness in real-world applications. Despite its inefficiency in querying over databases, the method offers advantages such as privacy preservation and reduced computing power requirements, making it suitable for handling ever-increasing data without frequent fine-tuning.