CRICKET ASSISTANT FOR IPL STATS

HR GUIDE: MR TAMIL SELVAN
CTO: MR PRAGADESWARAN
BY: ANIRUDH DEEPAK

JUNE 6, 2025
AXTR LABS

PROBLEM STATEMENT & OBJECTIVE:

- *Build a Modular AI Agent for Domain-Specific Knowledge Retrieval, Summarization, and Conversational Q&A.*
- *Design, implement, and demonstrate an AI system (agent) that can:*

  • Ingest and process a domain-specific dataset (e.g., clinical trial reports, scientific papers, or legal contracts).

  • Summarize documents using a fine- tuned LLM.

  • Answer user questions using a Retrieval-Augmented Generation (RAG) pipeline.

  • Act as a conversational agent with memory and multi-turn reasoning.

  • Deploy as an API or simple front-end demo.

DATASET USED:

- *For the given model I have taken a dataset of my interest where I took the IPL 2016 stats to find the stats of a batsman In that particular year. I took it as a .csv file as it is the simplest format to read and perform data-preprocessing.*
- *Why I took this dataset is:*
  - *IPL is a widely followed cricket league with rich player performance data.*
  - *Cricket stats are intuitive and engaging to work with.*
  - *The data is well structured and domain- specific.*
  - *The 2016 season especially has well balanced stats which makes it easier to work with.*

CODE EVOLUTION & DEPLOYMENT:

- *After importing the necessary data set and pre-processing it, I used basic set queries and get queries which would just get the result using its index and the text which was very basic. Only If a keyword was entered it could understand the prompt.*
- *After that I referred to a lot of websites to find out how I can perform a fine-tuned LLM since I am not well equipped with this part of the code. Hence first I decided to embed the code with Sentence-Transformer form Hugging Face which really good for semantic search.*
- *Then I used the cosine similarity to compare each query to the embeddings.*
- *Then I tried to use a pre-trained LLM model like OpenAI but due to certain limitations I could not use it, hence I had to use a free model. Then after going through few YouTube videos, I learnt about flan-t5-base which is a light weight pretrained LLM used for this purpose. It is good for basic multi-turn Q&A. I used the RAG function with the above model help with the help of the question context model.*
- *Then for the front-end I used GRADIO, since it is easy to use compared to Stream-lit. Hence, I used GRADIO for that purpose.*
- *Initially the model did not run that well but after adding some keywords in the set query function it worked fine.*

Q&A TESTING:

- *I tested the model with over 20 questions with the help of ChatGPT to give about 20 to 25 queries to test it out, the model worked just fine to answer at least 15 to 18 questions.*
- *From this I understood that the model could answer:*
    - *Basic Stats questions. (Ex: who has higher strike rate, Kohli or Warner?).*

  - *Player specific stats. (Ex: How many runs did Virat Kohli score?).*
  - *Superlative questions. (Ex: Who scored most centuries?)*
- *From this I understood that the model could not answer:*
  - *Outside dataset questions. (Ex: Who won IPL 2016?).*
  - *Complex Questions. (Ex: Who were the top 5 scorers of the season?).*
  - *Live Stats. (Ex: How much did Virat score in 2025?).*

PROS & CONS:

PROS:

- *Tailored specifically for IPL 2016 stats.*
- *Combines semantic search and generative Q&A model.*
- *Simple setup using GRADIO.*
- *Uses a lightweight LLM model.*

CONS:

- *Limited to the current dataset cannot predict it beyond this particular point.*
- *No conversational memory each query is treated Independently.*
- *Sometimes misses out on relevant context.*
- *Hallucinates the answers if it does not know the question.*

CODE LINK:

- *I have coded the project on* ***GOOGLE COLAB*** *since it helps in auto suggestion of the code which made it easy for me to type.*
- *Code:* *https://colab.research.google.com/drive/1kT23N04DdKNZPSIx8s0WW6nQT0bepMc9?usp=sharing*

SUMMARY:

This project is a smart question-answering system built to analyze and interact with IPL 2016 batting statistics. Using a CSV dataset of player performance, it allows users to ask natural language questions and receive relevant answers.

Key features include:

- ***Data retrieval via semantic search*** *using Sentence Transformers and cosine similarity.*
- ***Answer generation using the FLAN-T5 base language model*** *in a lightweight Retrieval-Augmented Generation (RAG) setup.*
- ***Custom rule-based logic for direct player comparisons*** *(e.g., averages, high scores).*
- ***Interactive front-end built with GRADIO****, making the tool easy to use.*

This modular agent bridges tabular sports data and natural language queries, making IPL stats exploration intuitive and engaging.

LEARNING OUTCOMES:

Through this project, I gained practical experience in working with real-world IPL 2016 data by performing data analysis using Pandas and NumPy. I learned how to use sentence embeddings through Sentence Transformers and applied cosine similarity to retrieve the most relevant context for user queries. I also integrated transformer-based models, particularly FLAN-T5, to generate accurate answers, helping me understand the Retrieval-Augmented Generation (RAG) approach. By transitioning from basic hardcoded logic to a more modular and scalable pipeline, I improved the system's flexibility and performance. Using Gradio, I built an interactive front-end, allowing real-time interaction with the AI agent. Overall, this project helped me understand the end-to-end flow of a question-answering system, sharpened my debugging and iteration skills, and gave me insights into balancing rule-based methods with deep learning models.