Iteration #03: Project Methodology

Pranav Sai Yada, Pratyush Rokade

March 2025

1 Introduction

FootGen AI is an intelligent system that allows users to query football match data using natural language. It supports data from both the English Premier League and Spanish La Liga, providing match statistics, results, and historical information through an intuitive chat interface. The system converts natural language queries to SQL, retrieves relevant data from a PostgreSQL database, and presents the results in a user-friendly format while maintaining conversation history. Figure 1 illustrates the system architecture and workflow of FootGen AI.

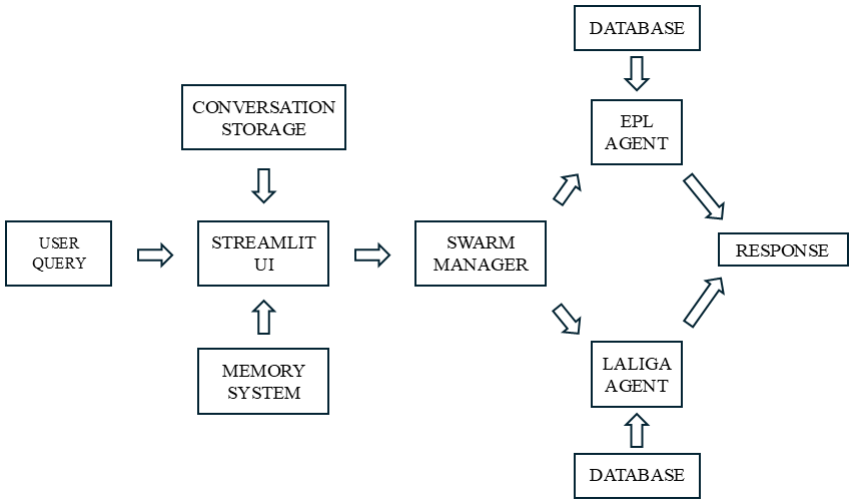

Figure 1: Figure 1: FootGen AI System Architecture

2 Purpose of Methodology

FootGen AI system employs several key methodologies to enable natural language understanding of football queries:

- Natural Language to SQL Conversion: Allows non-technical users to query complex football data without SQL knowledge.
- Agent-Based Routing: Employs specialized agents for different football leagues to ensure accurate query handling.
- Vector Memory Storage: Maintains conversation context by storing and retrieving relevant parts of previous interactions.
- Persistent Conversation Management: Enables users to revisit and continue previous conversations.

These approaches collectively create an accessible interface for exploring football statistics while preserving context across multiple user sessions.

3 Problem Statement

3.1 Defining the Problem

FootGen AI addresses several interconnected problems:

- Natural Language Understanding: Converting unstructured football queries into structured SQL queries (NL2SQL problem).
- Multi-Database Query Routing: Determining which football league database to query based on user intent.
- Context Preservation: Maintaining conversation history to support follow-up questions and provide coherent responses.
- Result Interpretation: Transforming raw database results into natural, human-readable responses.

3.2 Significance

This system has significant relevance in several domains:

- Sports Analytics: Democratizes access to football statistics for fans, journalists, and analysts without technical expertise.
- Database Accessibility: Demonstrates how natural language interfaces can make databases more accessible to general users.
- Conversational AI: Advances the field of multi-turn conversational systems that maintain context across interactions.
- Educational Tool: Serves as a learning resource for football history, statistics, and trends.

4 Data Collection and Preparation

4.1 Data Sources

The system utilizes football match data from Football-Data.co.uk which is a comprehensive source for football statistics covering multiple seasons of various football leagues around the world.

4.2 Data Description

The dataset comprises match records with the following characteristics:

- EPL Data: Contains match information from the English Premier League across multiple seasons.
- La Liga Data: Contains match information from the Spanish La Liga across multiple seasons.
- Key Attributes: Match date, teams, scores, statistics (shots, corners, fouls, cards), referee information, and betting odds.
- Temporal Coverage: Historical data spanning multiple football seasons, allowing for trend analysis.

4.3 Preprocessing Steps

The data undergoes several preprocessing steps:

- Data Cleaning: Standardization of team names and handling of missing values.
- Database Schema Creation: Structuring data into appropriate tables for EPL and La Liga.
- Data Conversion: Although stored as TEXT in PostgreSQL for flexibility, the system handles type conversions when performing calculations (using CAST as demonstrated in the SQL generation code).
- Memory Vectorization: User queries and responses are vectorized for semantic retrieval of relevant conversation history.

5 Selection of LLM Model

5.1 Model Consideration

Our system for FootGen AI relies on several AI models for different aspects of its functionality.

- NL2SQL Conversion: OpenAI GPT-3.5-turbo for converting natural language to SQL queries.

- Query Routing: Agent selection system to determine which database to query based on user intent.
- Response Generation: GPT-3.5-turbo for converting SQL results into natural language responses.
- Memory System: Vectorization and similarity search for retrieving relevant conversation context.

5.2 Final Model Selection

The system primarily utilizes GPT-3.5-turbo because it does not provide detailed answers to football-related questions and instead relies on attaching web links for further self-research.

6 Model Development and Tuning

6.1 Architecture and Configuration

The system employs a multi-component architecture:

- Frontend: Streamlit web application providing the user interface and conversation display.
- Agent System: Swarm-based routing logic to direct queries to appropriate handlers.
- NL2SQL Engine: Prompt-engineered GPT instances with database schema context for accurate SQL generation.
- Memory System: Vector store (Chroma) for semantic retrieval of conversation history.
- Persistence Layer: JSON-based storage for maintaining conversations across sessions.

6.2 Training Process

The system leverages pre-trained language models with specialized prompt engineering:

- Schema-Aware Prompting: The NL2SQL component includes detailed database schema information to guide SQL generation.
- Team Name Standardization: Prompts include lists of standardized team names to ensure accurate entity recognition.
- SQL Pattern Guidance: Example patterns are provided to ensure proper query structure and prevent SQL injection.
- Type Casting Instructions: Explicit guidance on handling text-to-numeric conversions for aggregation functions.

6.3 Hyperparameter Tuning

The system configuration includes several tuned parameters:

- Temperature Setting: SQL generation uses temperature=0 for deterministic outputs, while natural language responses use temperature=0.7 for more varied human-like text.
- Token Limits: Max tokens are constrained (150-250) to ensure concise and focused outputs.
- Memory Relevance: Vector similarity thresholds for retrieving relevant conversation context.
- UI Configuration: Optimized sidebar and chat display settings for user experience.

7 Evaluation and Comparison

The FootGen AI system can be evaluated across several dimensions:

7.1 Query Accuracy

- SQL Generation: Success rate of converting natural language to valid, executable SQL queries.
- Result Relevance: Proportion of queries that return information relevant to the user's question.
- Entity Recognition: Accuracy in identifying football teams, seasons, and statistics.

7.2 Response Quality

- Natural Language: Clarity and readability of generated responses.
- Context Utilization: Effectiveness of incorporating previous conversation context.

7.3 User Experience

- Response Time: End-to-end latency from query submission to response display.
- Conversation Management: Ease of creating, switching between, and deleting conversation threads.
- Error Handling: Graceful handling of queries outside the system's knowledge domain.