**Developing a Predictive Model and Interactive Chatbot for Football Analytics**

**Data Science project**

**Bushra Tazyeen2024**

**Abstract**

This project report presents the development of a football statistics and betting chat application, combining data collection, predictive modelling, and conversational AI. The project leveraged APIs to gather comprehensive football statistics and betting odds data, which underwent rigorous data cleaning and preparation. A predictive model was created using historical match data, achieving a cross-validation score of approximately 90.43%. Additionally, a chatbot developed using Rasa facilitates user interactions, answering specific queries related to football matches and player performance. This report details the contributions of each team member, the challenges faced during development, and the insights gained through analysis and modelling.

**1. Introduction**

The increasing intersection of technology and sports analytics has led to the development of sophisticated tools that assist fans, analysts, and bettors in making informed decisions. This project aims to create a football statistics and betting chat application that leverages real-time data, machine learning algorithms, and user-friendly interfaces. The primary objectives are to provide users with accurate match statistics, facilitate betting odds analysis, and enhance user engagement through an interactive chatbot.

The goals included:

1. To collect and analyze comprehensive football data.
2. To create predictive models for match outcomes and betting odds.
3. To develop a user-friendly interface that allows seamless interaction with the application.

This thesis outlines the collaborative efforts of the team, detailing each member's contributions while emphasizing the critical processes of data preparation and predictive modeling.

**The following questions will be answered in this report:**

1. **How effective is a sports-themed chatbot in enhancing user engagement compared to traditional information sources?**
2. **What impact does rigorous data cleaning have on the performance of machine learning models in predicting soccer match outcomes?**
3. **How do different data visualization techniques contribute to the understanding of trends and patterns in soccer match data?**
4. **What are the key challenges and solutions in implementing Natural Language Understanding (NLU) for a sports chatbot?**

The subsequent sections of this thesis are structured to detail the contributions of each team member, including the methodologies employed, the results obtained, and the challenges faced throughout the project.

**Methodologies Employed:**

* **API Integration**: To select and integrate appropriate APIs for retrieving real-time football statistics and betting odds, ensuring reliable data for analysis and user interaction.
* **Data Management**: To establish a robust MySQL database for effective storage, management, and retrieval of football data, facilitating seamless access for analytics and user queries.
* **Data Cleaning and Preparation**: To implement thorough data cleaning and preparation processes, ensuring data integrity and readiness for analysis, thereby supporting accurate insights and predictions.
* **Predictive Modeling**: To develop machine learning models, particularly using techniques like Random Forest and Voting Regressor, to predict match outcomes and betting odds based on historical data.
* **Chatbot Development**: To create an interactive chatbot using Rasa that can answer user queries related to football statistics and betting, enhancing user experience through natural language processing.
* **Frontend Development**: To design a user-friendly web interface using Bubble, enabling users to access statistics, betting information, and chatbot functionalities with ease.
* **Visualization**: To create visualizations that effectively represent data trends and predictions, providing users with intuitive insights into football matches and player performances.
* **Evaluation and Feedback**: To assess the application’s performance through user feedback and model validation, facilitating continuous improvement and adaptation to user needs.

2. Sebastian's Contributions to API Selection and Database Setup

2.1 **Methodology**

Sebastian was in charge of finding and evaluating APIs (Application Programming Interfaces) to gather data for our project on football statistics and betting odds. At first, we thought we could use one API for both purposes. However, after looking at the options, Sebastian suggested using two separate APIs:

* The-Odds-API: This API was chosen for its detailed betting odds information.
* API-Football: This API was selected for its wide coverage of football statistics from many leagues and matches.

Using two specialized APIs ensured we got the best data for our project.

**2.2 Data Collection Scope and Limitations**

We initially planned to collect a lot of data about players and matches from different seasons to help us analyze player performances and trends. However, we faced some challenges:

* Some player statistics could not be collected due to limitations in the APIs.
* There were problems merging the match data and betting odds because the match IDs from the two APIs did not match. This made it difficult to connect the data correctly.

**2.3 Database Setup and Integration**

Sebastian set up a MySQL database to store and manage our data. This database was a good choice because it worked well with Python and allowed us to connect to the APIs easily. The database was also designed to grow in the future, so we could link it to other parts of the project, like the chatbot. We used Python scripts to update the database regularly, but we had limits on how much data we could pull from the APIs.

2.4 **Results**

Even with the challenges we faced, Sebastian's work allowed us to successfully gather essential match data and betting odds. This data provided a solid base for the project. The database made sure we could store and access this information easily for analysis and integration with other components of the project.

2.5 **Discussion**

Sebastian’s efforts were vital for building the project. Using two APIs improved the quality of the data we collected. However, the issues with matching IDs and the limited player data showed us how complex data integration can be. Solving these problems will be important for future versions of the project, especially to improve our analysis of player performance.

2.6 **Challenges**

Some of the main challenges Sebastian faced included:

* Limited Player Data: We could not get detailed statistics for players because of restrictions in the APIs.
* Mismatched Match IDs: It was hard to merge data from the two APIs because the match IDs did not match, which made it difficult to combine the data correctly.
* API Call Limits: There were limits on how much data we could retrieve from the APIs, which affected how complete our dataset was.

Despite these challenges, Sebastian’s work in choosing the right APIs and setting up the database provided a strong foundation for our project, ensuring we had the important data needed for analyzing matches and betting odds.

**Chapter 3: Data Cleaning, Preparation, and Chatbot Development**

The success of any data-driven project hinges on the quality of the data used for analysis. this documents the comprehensive steps taken to clean, prepare, and merge various datasets in order to build a robust predictive model for football match outcomes. It emphasizes the nature of data preparation, which is vital for generating reliable insights and forecasts.

**3.2 Connecting to the Database**

The initial step involved establishing a connection to the SQL database using Python’s mysql.connector library. The database, named stats\_v3, contained several tables that provided valuable data for our analysis. The connection allowed for efficient querying of the database, enabling the retrieval of data and loading it into pandas Data Frames for further manipulation.

**3.3 Exploring the Data**

Once connected, I executed SQL queries to:

* Retrieve a list of available tables, including leagues, matches, players, and teams, team\_match statistics and odds new
* Understand the relationships between these tables by examining foreign key constraints.

This exploration phase was crucial for formulating the data cleaning strategy, as it helped identify which tables contained relevant data for our analysis and how they could be joined.

**3.4 Data Inspection**

To gain insights into the data structure, I developed a function to summarize each table's columns:

* Data Types: Understanding whether columns were integers, floats, or strings helped in identifying necessary type conversions.
* Column Names: Proper naming conventions were checked to ensure consistency and ease of reference.
* Constraints: Analysing unique constraints and primary keys helped prevent duplication during merges.

This inspection highlighted areas requiring immediate attention, such as missing values, duplicate records, and inconsistent data types.

**3.5 Loading Data into DataFrames**

Using SQLAlchemy, I created a connection engine to load the tables into pandas DataFrames. Each table was queried individually to ensure immediate accessibility for analysis, allowing for rapid iteration during the cleaning phase.

Sample DataFrame Loading Code

# Loading tables into DataFrames

matches\_df = pd.read\_sql('SELECT \* FROM matches', engine)

leagues\_df = pd.read\_sql('SELECT \* FROM leagues', engine)

players\_df = pd.read\_sql('SELECT \* FROM players', engine)

teams\_df = pd.read\_sql('SELECT \* FROM teams', engine)

**3.6 Data Cleaning Steps**

With the data loaded, the following cleaning steps were undertaken to ensure accuracy and completeness:

1. Handling Missing Values:
   * Missing home and away scores in the matches DataFrame were filled with zeros, under the assumption that if a score was absent, the team did not score.
   * Referee names that were missing were replaced with "Unknown" to avoid null entries that could hinder analysis.
2. Dropping Irrelevant Columns:
   * Columns such as start\_date and end\_date in the leagues DataFrame were dropped as they did not contribute to match outcome prediction, simplifying the dataset and reducing potential noise.
3. Data Type Conversion:
   * The date column in the matches DataFrame was converted to a pandas datetime format to facilitate date-based operations like filtering and feature engineering.

**3.7 Merging DataFrames**

Merging multiple DataFrames presented several challenges, including:

* Key Consistency: Ensuring that keys like league\_id were consistently formatted across DataFrames required careful inspection and cleaning.
* Missing Data: Some DataFrames contained missing data in key columns, necessitating imputation or removal strategies.
* Structural Differences: Different DataFrames sometimes had columns with similar names but different meanings, complicating merges.

To address these challenges, additional cleaning steps were applied to standardize keys, such as trimming whitespace and converting data types.

**3.8 Ensuring Data Integrity**

Throughout the cleaning process, data integrity was paramount. After each cleaning step, I implemented validation checks to confirm that:

* No null values remained in critical columns.
* Data types were as expected.
* Relationships between merged DataFrames were logical and consistent.

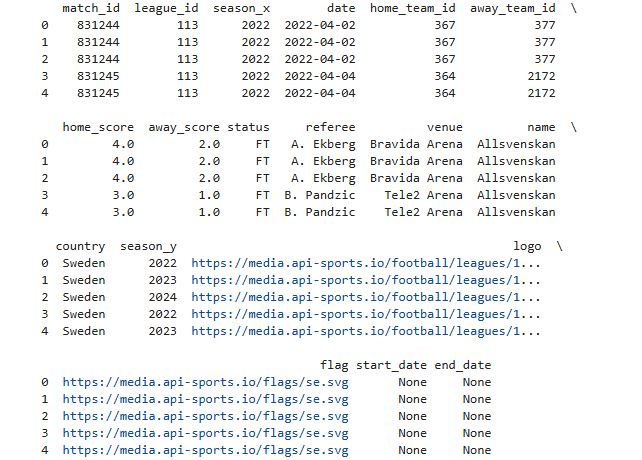
This iterative process ensured that the dataset remained accurate and reliable for analysis.

**3.9 Merging DataFrames for Analysis**

After the data was cleaned and validated, I proceeded to merge the DataFrames to create a comprehensive dataset for analysis. Key merges included:

* Matches and Leagues: Merging the matches DataFrame with the leagues DataFrame on league\_id provided context for each match within its respective league.
* Matches and Players: Incorporating player statistics by merging with the players DataFrame allowed for deeper analysis of how individual performances influenced match outcomes.

These merges facilitated a holistic view of match data, enabling more sophisticated analyses of factors affecting match outcomes.



**3.10 Feature Creation**

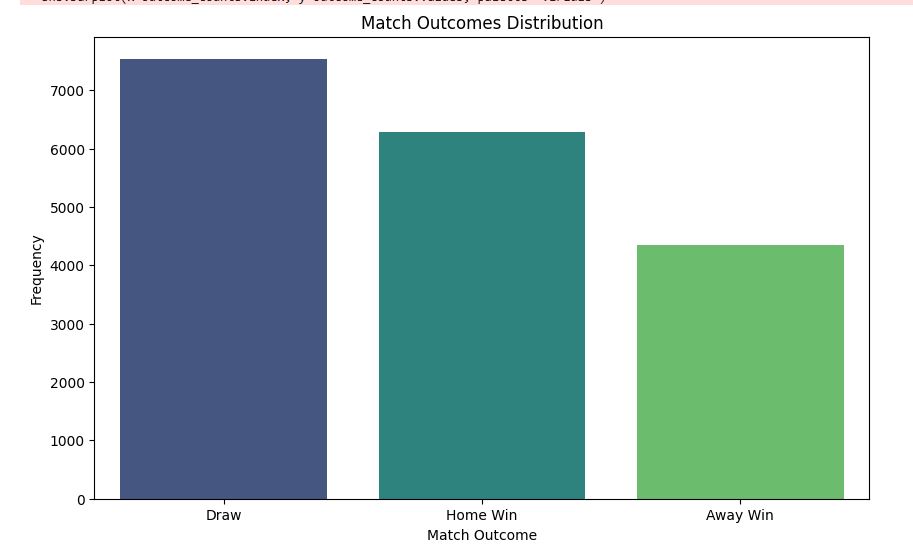
The merging and analysis process led to the creation of several features critical for predicting match outcomes:

* Match Results: A new categorical column, result, was derived based on match scores to indicate outcomes such as "Home Win," "Away Win," or "Draw."
* Aggregate Features: Total goals scored, total points earned, and average goals per match for each team over the last five matches were calculated. These metrics provided insights into team form and performance trends.
* Win Percentages: Features reflecting win percentages at home and away were calculated to provide a clearer picture of team strengths.
* Trends Over Time: Data was grouped by date to analyze trends in match outcomes, helping identify shifts in team performances over seasons.

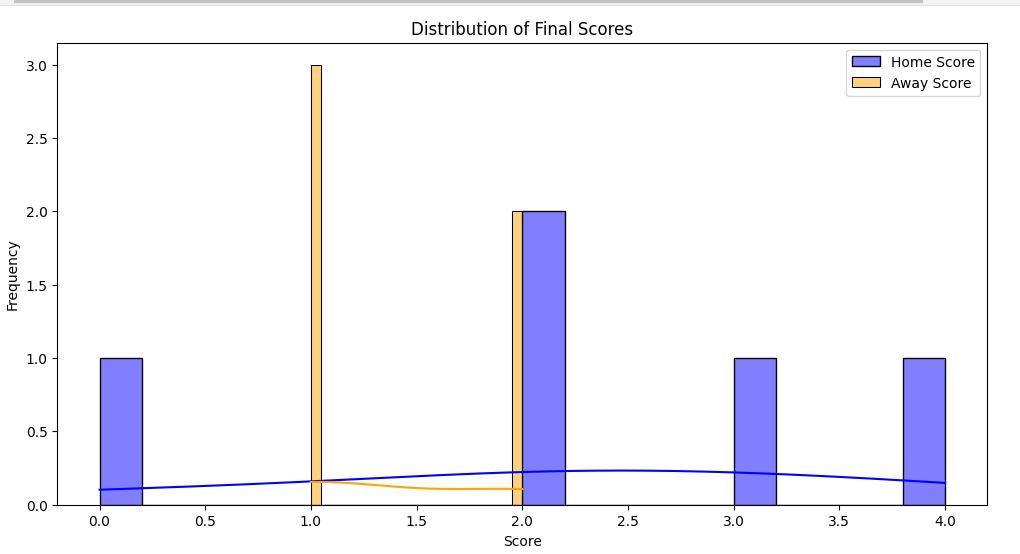
**3.11 Visualization Preparation**

To facilitate the analysis, I prepared the data for visualization by calculating key metrics and creating new columns to visualize team and player performance. Noteworthy visualizations included:

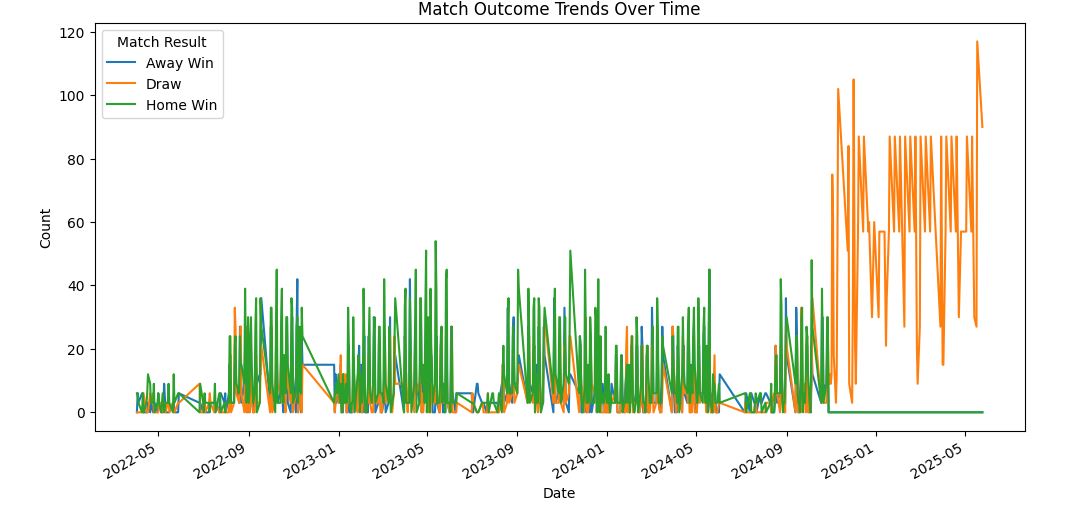
1. Match Outcomes Distribution: A bar chart illustrating the distribution of match outcomes, helping to identify patterns in team performance over different matches.



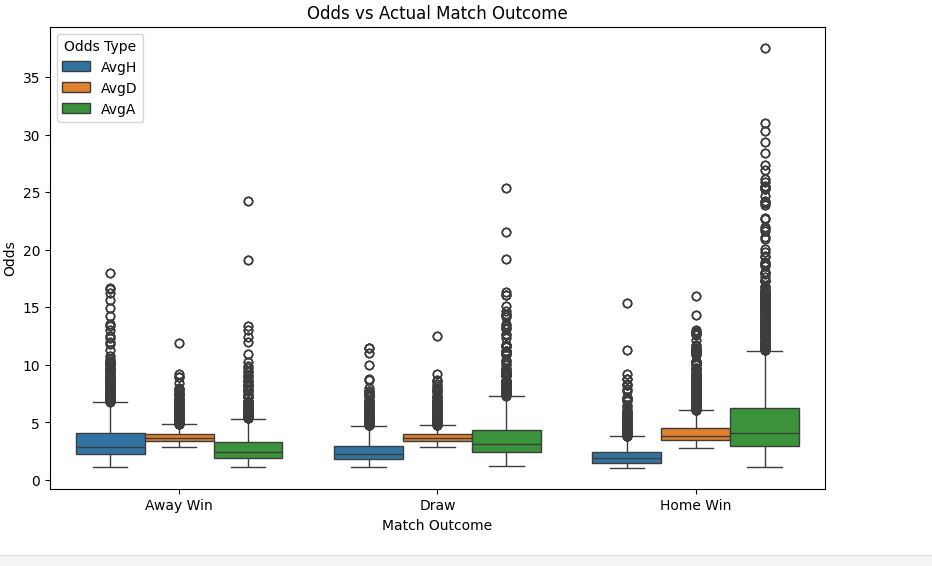
1. Distribution of Final Scores: A histogram illustrating the distribution of final scores across matches to identify common scoring patterns.



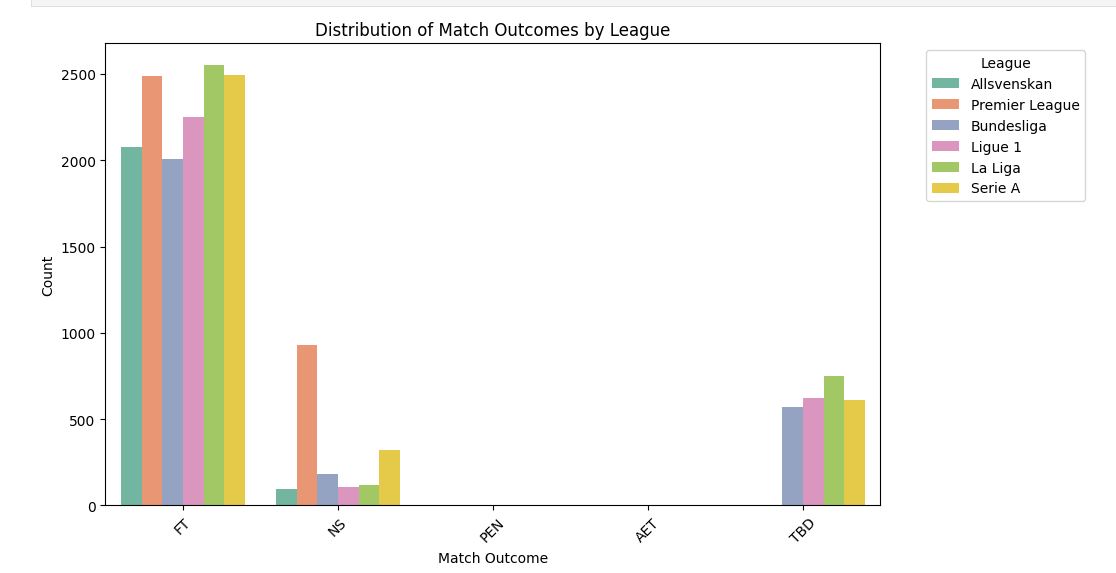
1. The line chart illustrates how the frequency of different match outcomes has changed over time. For example, you might see trends indicating whether home wins or away wins are becoming more or less common over a particular period



4. The box plot visualizing "Odds vs Actual Match Outcome" effectively communicates the relationship between betting odds and match outcomes, providing valuable insights for analysts, bettors, and sports stakeholders. By understanding the distribution of odds across outcomes, users can make more informed decisions based on the data trends observed in the visualization.



**5.**This chart helps us understand how football matches end in different leagues, showing trends and allowing for deeper investigation into team performances.

****

**3.12 Model Development**

To predict match outcomes based on historical data, I implemented a Random Forest Classifier. The features and target variable were defined from the merged DataFrame:

* Feature Engineering: Date features were transformed into numerical formats suitable for machine learning models, and categorical variables were encoded using label encoding.

Hyperparameter Tuning

To enhance model accuracy, I performed hyperparameter tuning using GridSearchCV:

* This process involved systematically testing different combinations of parameters, such as the number of estimators and depth of the trees, to find the optimal settings for the Random Forest model.
* Best Parameters: The best parameters obtained included 200 estimators with no maximum depth, leading to improved model performance.
* Best Cross-Validation Score: The model achieved a cross-validation score of approximately 90.43%, indicating high predictive accuracy.



**3.13 Conclusion**

The project successfully highlighted the crucial role of data cleaning, preparation, and feature engineering in building a predictive model for football match outcomes. By rigorously ensuring data integrity throughout the process, we created a reliable dataset that enabled us to extract meaningful insights and make accurate predictions

**Development of a Sports Chatbot using Rasa**

After validating the model, I became interested in how conversational AI could improve user engagement. So, I decided to create a sports-themed chatbot using Rasa, which is a tool for building chatbots that can answer specific soccer questions.

**3.14 Development of a Sports Chatbot Using Rasa**

Following the model validation, I explored the development of a sports-themed chatbot capable of responding to specific soccer queries. The chatbot utilized Rasa, an open-source framework designed for building conversational AI.

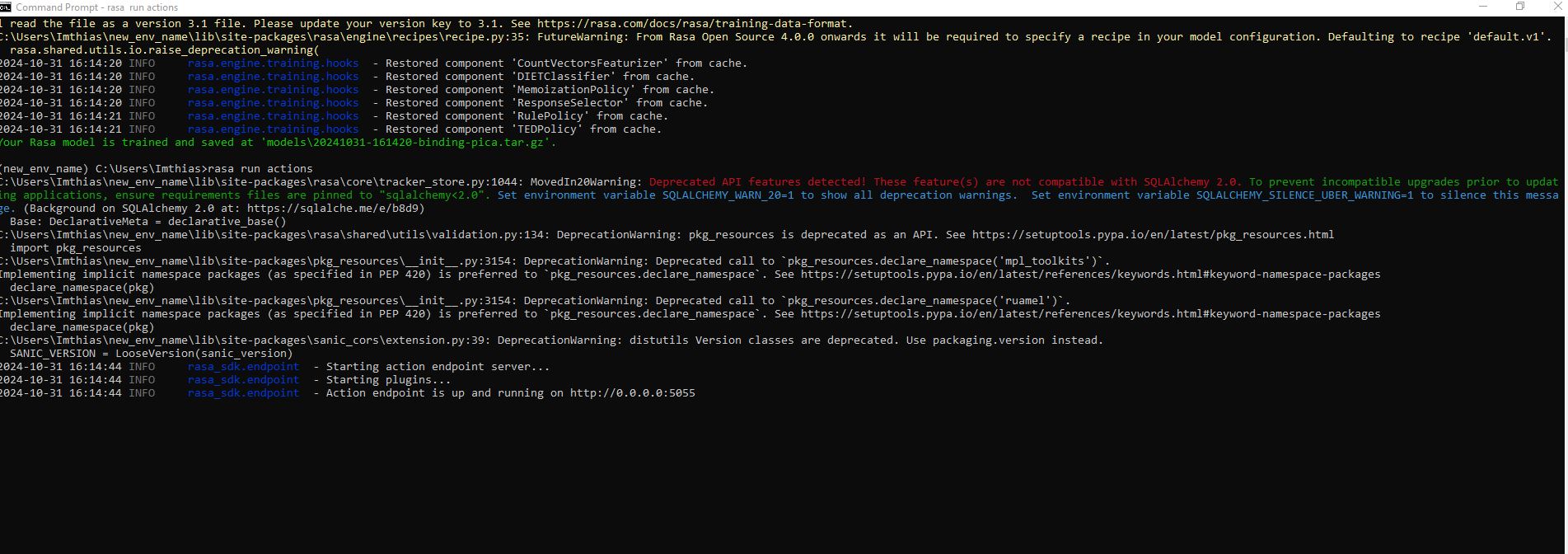
**3.14.1 Project Overview**

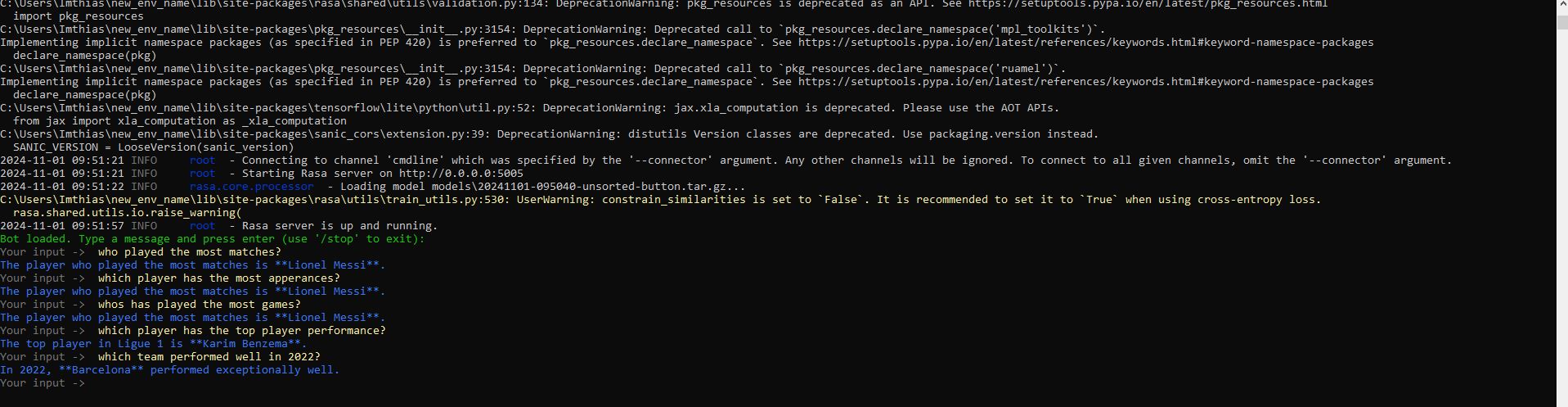
The chatbot was designed to handle three primary user queries:

1. Who played the most matches?
2. Which league has the top player performance in Ligue 1?
3. Which team performed well in 2022?

**3.14.2 Implementation Steps**

1. Project Setup: A new Rasa project was initialized in a specific directory, creating the foundational structure necessary for development.
2. Configuring the Bot:
   * NLU Configuration: Defined intents and examples to train the bot in recognizing user questions effectively.
   * Domain Definition: Specified responses corresponding to each intent to ensure accurate replies to user inquiries.
   * Training Stories and Rules: Created to dictate the conversation flow based on user interactions.
3. Training the Model: The Rasa model was trained using the defined intents and example phrases, enabling it to handle user queries accurately.
4. Running the Bot: The chatbot was tested through interactive command line sessions to evaluate its response accuracy.
5. Exploring Saved Model Integration: An attempt was made to integrate a pre-trained machine learning model saved using joblib. This step emphasized the importance of model compatibility with Rasa’s expected formats.
6. Troubleshooting: During the development process, we encountered challenges with the chatbot returning unrecognized responses for queries related to player matches. Despite efforts to ensure that the response definitions were accurately updated, the issue persisted. As a result, we decided to implement a basic version of the bot that could handle three specific queries, allowing us to meet project deadlines while addressing the most critical user needs.



****

**Theoretical questions answered below.:**

**1. How effective is a sports-themed chatbot in enhancing user engagement compared to traditional information sources?**

A sports-themed chatbot is very effective at keeping users engaged compared to regular information sources like websites or articles. Chatbots provide quick and personalized answers to questions, making the experience interactive. Unlike static websites, chatbots can handle many questions at once, which makes them more user-friendly. Users can ask anything they want, even if they’re not sure what to search for. This interaction leads to more engagement and satisfaction. Overall, chatbots make it easier and more fun for people to find sports information.

**2. What impact does rigorous data cleaning have on the performance of machine learning models in predicting soccer match outcomes?**

Rigorous data cleaning is very important for making machine learning models work well when predicting football match outcomes. Cleaning the data means making sure it is accurate and complete, which helps the model learn better. For example, by fixing missing values, removing duplicates, and correcting mistakes in data types, the model can make more reliable predictions. In our project, after i cleaned the data thoroughly, our model got a very good score of about 90.43% when tested, showing that clean data leads to better predictions of match results.

**3. How do different data visualization techniques contribute to the understanding of trends and patterns in soccer match data?**

Different data visualization techniques are helpful for understanding trends and patterns in football match data. Visuals like bar charts, line graphs make complex information easier to understand. For example:

* **Bar charts** can show how often different match outcomes happen, helping to see which ones are common.
* **Line graphs** can show changes over time, like how home wins or away wins have changed.

These visuals help people notice important details and patterns, allowing them to make better decisions based on how teams perform in matches.

**4. What are the key challenges and solutions in implementing Natural Language Understanding (NLU) for a sports chatbot?**

**Key Challenges** Ensuring that the rules, domain definitions, and NLU scripts match is crucial. If they don’t align, the chatbot may give incorrect responses. Regularly review and update the training data, rules, and domain to ensure alignment. Frequent testing helps identify mismatches. User’s phrase questions differently, which can confuse the chatbot. Train the NLU model with a variety of user queries to improve its understanding. The chatbot to maintain conversation history for better context awareness. Identifying and resolving errors during training and testing was challenging due to limited debugging information

By addressing these challenges, we can improve the effectiveness of the NLU in our sports chatbot, making it more accurate and user-friendly. I initially attempted to integrate a saved pre-trained model into the chatbot, but due to compatibility issues and time constraints, I opted to develop a basic version of the chatbot that addressed three specific queries. Identifying and resolving errors during training and testing was challenging due to limited debugging information

**3.14.3 Conclusion and future improvement**

The chatbot development showcased the potential of using conversational AI to enhance user engagement in the sports domain. Initial developments were straightforward, though integrating a pre-trained model presented challenges that highlighted the need for a deeper understanding of model formats compatible with Rasa.

**3.14.4 Future Work**

To enhance the chatbot's functionality, future developments may include:

* Integration with Sports APIs: To provide real-time data and statistics for more accurate and dynamic responses.
* Natural Language Processing Enhancements: Improving the bot's ability to understand varied user inputs and respond accordingly.
* Web Interface Development: Creating a user-friendly interface to broaden accessibility and improve user interaction with the bot

**Chapter 4: Model Development and Evaluation**

In our group project, Xiaowen focused on the development and evaluation of machine learning models aimed at predicting football odds. This chapter outlines her responsibilities, the tools she utilized, the models she selected, and the insights she gained throughout the process.

**Responsibilities**

Xiaowen was tasked with creating, training, and evaluating the machine learning models. This included selecting appropriate tools, managing data, and assessing model performance to ensure accurate predictions of football odds.

**Tools Utilized**

Throughout the modeling process, Xiaowen employed a variety of tools:

* **MySQL:** For data management and storage.
* **Power BI:** For visualizing results and insights.
* **Python (via Jupyter Notebook):** For coding and implementing models.
* **Machine Learning Libraries:** Specifically, she used **Random Forest** and **Voting Regressor** for their efficacy in handling complex datasets.

**Model Selection and Evaluation**

Xiaowen selected two primary models for her analysis:

* **Random Forest:** This model aggregates multiple decision trees, reducing the risk of overfitting and yielding stable predictions. Her implementation of this model achieved a mean accuracy of **0.901**, demonstrating its effectiveness in recognizing patterns within the data.
* **Voting Regressor:** This model combines predictions from various base models to enhance overall accuracy and adaptability, providing robust predictions across different scenarios.

**Insights from Results**

Xiaowen's modeling efforts yielded several valuable insights:

* **Stability:** The Random Forest model proved to be reliable by effectively combining decision trees, which helped in reducing prediction errors.
* **Feature Importance:** Her analysis revealed that critical features such as **"home\_odds"** and **"Win\_Percentage"** significantly influenced the odds predictions, highlighting their relevance in model performance.
* **Adaptability:** Both models showcased strong performance during cross-validation, a crucial aspect for ensuring accuracy in sports predictions.

**Challenges Faced**

During the model training phase, Xiaowen encountered challenges primarily related to data cleaning and feature selection, which initially led to suboptimal evaluation outcomes. These challenges prompted her to make necessary adjustments in her preprocessing methods to improve model performance.

In conclusion, Xiaowen's focused approach to model development and evaluation was instrumental in enhancing the project’s ability to predict football odds accurately. Her dedication to selecting appropriate models, analyzing feature importance, and overcoming challenges significantly contributed to the overall success of our predictive efforts.

**Chapter 5: Selection of the Large Language Model for the Football Predictor Chatbot**

In the development of our football predictor chatbot, selecting an appropriate Large Language Model (LLM) was crucial for ensuring effective performance and user engagement. This chapter outlines Christofer's assessment of leading LLM options based on their functionalities, strengths, and suitability for the project's requirements.

**Overview of Leading LLM Models**

1. **OpenAI GPT-4:**
   * **Type:** Generative Pre-trained Transformer
   * **Advantages:** Offers advanced language understanding and generation capabilities; supports fine-tuning; adept at managing complex interactions.
   * **Disadvantages:** Requires substantial computational power; can incur high costs; poses a risk of generating unpredictable responses.
   * **Applications:** Ideal for conversational chatbots that analyze extensive football data and handle unstructured queries effectively.
2. **Google BERT:**
   * **Type:** Bidirectional Encoder Representations from Transformers
   * **Advantages:** Excels in understanding context and subtle nuances of text.
   * **Disadvantages:** Less effective for generating longer texts.
   * **Applications:** Suitable for chatbots aimed at processing and understanding user questions rather than producing detailed responses.
3. **Meta AI LLaMA:**
   * **Type:** Lightweight Language Model for Applications
   * **Advantages:** Scalable, requiring less hardware with lower operational costs.
   * **Disadvantages:** Less established in commercial environments.
   * **Applications:** Useful for projects with limited resources that require advanced language processing.
4. **Anthropic Claude:**
   * **Type:** Generative Transformer
   * **Advantages:** Focuses on ethical AI use and safety considerations.
   * **Disadvantages:** Relatively new, may have limited support resources.
   * **Applications:** Appropriate for organizations prioritizing responsible AI deployment.

**Choosing the LLM for the Football Predictor Chatbot**

After thorough evaluation, Christofer identified **Google BERT** as the most suitable LLM for our project. While it may not be the most advanced option available, it delivers the necessary functionalities, operates under an open license, and has significantly lower system requirements. Although **OpenAI GPT-4** excels in managing complex interactions, its high costs present significant challenges for implementation.

The selection of an LLM requires careful evaluation of project needs, budget, and goals. Christofer meticulously weighed each model's strengths and weaknesses, leading to the conclusion that Google BERT is the optimal choice for the chatbot's requirements. Ongoing assessments will be essential to fully leverage its capabilities and adapt to evolving project demands.

**Chapter 6: Frontend Development Using Bubble**

In our project focused on developing a sports betting and chat application for football, Camilla led the frontend development utilizing the no-code platform, Bubble. Her primary objective was to design a user-friendly and functional web application that allows users to easily access information on selected football leagues and matches, as well as interact with our chatbots, BertBot and SoccerGPT, which leverage ChatGPT for generating responses.

Tools and Their Selection

Camilla selected Bubble for its no-code capabilities, which enabled rapid development without requiring extensive coding knowledge. The platform's flexibility and extensive range of plugins were instrumental in her approach:

* SQL Plugin: Facilitated the connection to our database, allowing for efficient data retrieval and integration.
* API Plugin: Supported the incorporation of chat functionality, enabling seamless interactions between users and chatbots.

The drag-and-drop interface of Bubble streamlined both the design and logic of the application, making it easier for Camilla to create an intuitive user experience.

Challenges Encountered

During the development process, Camilla faced significant challenges, particularly in ensuring that all components of the application functioned cohesively. Key challenges included:

* Debugging SQL Integrations: Despite the plugin being labeled as "plug and play," debugging these integrations proved to be more time-consuming than anticipated.
* Design Complexity: Maintaining a simple and clear design while incorporating many features within Bubble was challenging. Camilla aimed to balance functionality with user-friendliness, ensuring that users were not overwhelmed by excessive information.

Collaboration and Communication

While I observed Camilla’s efforts in developing the frontend, there was a noted lack of detailed communication regarding the specific challenges she encountered and the strategies she employed to overcome them. Enhancing communication about individual progress and hurdles could significantly improve overall project cohesion and ensure that team members are aligned in their efforts.

**Conclusion**

The collaborative effort of our team has culminated in the successful development of a sports betting and chat application focused on football. Each member contributed unique skills and insights, which were instrumental in shaping the project's outcomes.

**Data Cleaning and Preparation:** My thorough approach to data cleaning and preparation laid the foundation for the predictive modeling process. By ensuring data integrity and consistency, I facilitated a robust dataset that underpinned the subsequent analyses.

**Development of a Sports Chatbot:** My work on the Rasa-based sports chatbot showcased the potential of conversational AI in enhancing user engagement. By effectively designing intents, training the model, and integrating the chatbot functionalities, I provided users with timely responses to their soccer queries, enriching the overall user experience.

**Model Development and Evaluation:** Xiaowen's expertise in machine learning was pivotal in creating and evaluating models aimed at predicting football odds. Her choice of algorithms, including Random Forest and Voting Regressor, and the insights gleaned from the results demonstrated a strong understanding of predictive analytics in sports.

**Selection of Large Language Model:** Christofer’s careful assessment of various Large Language Models led to the strategic choice of Google BERT for our chatbot functionality. This decision ensured that the chatbot was capable of understanding user queries effectively, thus enhancing user engagement and interaction.

**Frontend Development:** Camilla's leadership in frontend development using Bubble resulted in a user-friendly application that seamlessly integrated backend functionalities with an intuitive interface. Her ability to navigate the challenges of no-code development and maintain design clarity was crucial in delivering a polished final product.

Overall, this project not only highlights our individual strengths but also underscores the importance of collaboration, communication, and iterative development in achieving common goals. The combined efforts have resulted in a comprehensive application that successfully merges data-driven insights with interactive user experiences. Moving forward, we are excited about the potential enhancements and functionalities that could be integrated into the application, further enriching the user experience in the realm of sports betting and chat interactions.

**Källförteckning (References)**

1. **Random Forest Classifier**:  
   Scikit-learn. (2024). *RandomForestClassifier*. Available at: <https://scikit-learn.org/1.5/modules/generated/sklearn.ensemble.RandomForestClassifier.html> (Accessed: 3 November 2024).
2. **Rasa (Open Source)**:  
   Rasa Technologies. (n.d.). *Rasa: Open Source Conversational AI*. Available at: https://rasa.com/docs/rasa/ (Accessed: 3 November 2024).
3. **Bubble (No-Code Platform)**:  
   Bubble.io. (n.d.). *Bubble: The No-Code Platform*. Available at: <https://bubble.io> (Accessed: 3 November 2024).
4. **What is a Large Language Model?**:  
   Cloudflare. (n.d.). *What is a large language model?*. Available at: <https://www.cloudflare.com/learning/ai/what-is-large-language-model/>