VECTOR 1 F1 DECISION SUPPORT SYSTEM

Project Reference No.: *47S_BE_5398*

College ***:*** *City Engineering College, Bengaluru*
Branch ***:*** *Department Of Artificial Intelligence and Machine Learning*
Guide(s) ***:*** *Prof. Vagdevi S.*
Student(S) ***:*** *Mr. Kedar Joshi*
Ms. Kokila K. R.
Mr. Suhas Gajanana

Keywords:

Formula 1, Pitstops, Tyre Compound, Real-time, Grand Prix

Introduction

In the high-stakes world of Formula One racing, where the difference between victory and defeat is measured in milliseconds, optimizing race strategy is paramount. Traditional approaches to making crucial decisions, such as when to pit-stop and which tyres to choose, rely heavily on historical and real-time telemetry data, alongside the intuition of expert strategists. However, despite the availability of vast amounts of data, these strategy calls have not fully harnessed the power of data science and machine learning, leaving room for significant improvements.

This project aims to revolutionize race strategy by integrating machine learning models with deep human expertise to make faster, data-backed strategic decisions. By leveraging historical race data and real-time telemetry from sources such as Ergast and FastF1, the initiative focuses on optimizing pit-stop timings and tyre selections. The ultimate goal is to create a dynamic system that combines the precision of machine learning with the nuanced intuition of seasoned racing professionals, thereby enhancing overall race performance.

For instance, consider a scenario involving driver Charles Leclerc on lap 35 of 58, currently in P4 with a gap of +1.8 seconds to the car ahead and -12.2 seconds to the car behind. With current lap times slowing down and tyre degradation evident from high temperatures and dropping pressures, the model suggests an optimal pit-stop on lap 41 with a switch to soft tyres. This decision is based on the expected performance gain and the strategic timing to maintain and potentially improve track position.

Through this innovative approach, the project aims to set a new standard in race strategy, blending the best of technology and human insight to drive competitive advantage in Formula One.

Objectives

The key objectives of the "Vector One" project are multifaceted and designed to address the needs.

- Enhance Pit-Stop Decisions: Develop predictive models to determine the optimal timing for pit-stops, minimizing time lost and maximizing performance gains.
- Optimize Tyre Selection: Utilize data analytics to recommend the best tyre compounds based on real-time conditions and historical performance data, ensuring optimal grip and durability.
- Improve Overall Race Performance: Combine the precision of machine learning with the nuanced intuition of racing professionals to enhance strategic decision-making, leading to better race outcomes.
- Develop Real-Time Analytics Tools: Create tools that can process and analyze telemetry data in real-time, providing immediate strategic insights during races.
- Leverage Historical Race Data: Use extensive historical race data from sources like Ergast to inform and train machine learning models, ensuring robust and accurate predictions.
- Seamless Integration with Human Expertise: Design the system to work alongside seasoned racing professionals, allowing for a harmonious blend of automated insights and expert intuition.
- Dynamic Strategy Adjustment: Enable the system to adjust strategies dynamically based on real-time race developments, ensuring flexibility and responsiveness to changing conditions.

Methodology

1. Data Collection using Open F1 and Fast F1 APIs:

- Begin by thoroughly understanding the data available through the Open F1 and Fast F1 APIs, including their endpoints, parameters, and rate limits.
- Implement a data retrieval mechanism that efficiently queries the APIs to collect relevant real-time telemetry data and historical race archives.
- Design data structures to store the retrieved data in a structured format for further processing and analysis.

2. Data Preprocessing:

- Clean the collected data to handle missing values, outliers, and inconsistencies using techniques such as data imputation, outlier detection, and data validation.
- Normalize or scale features as necessary to ensure consistency and facilitate model convergence, considering the different scales and distributions of the data.
- Perform feature engineering to extract relevant insights and create meaningful features for subsequent analysis, such as deriving new features from existing ones or encoding categorical variables.

3. Data Streaming with Kafka:

- Set up a Kafka cluster, considering factors such as cluster size, replication factor, and topic partitioning to ensure scalability and fault tolerance.
- Configure Kafka topics for data ingestion, defining appropriate schemas and message formats to ensure compatibility and interoperability.
- Develop a Kafka producer application to publish the preprocessed data to designated Kafka topics, implementing batching and compression techniques for efficient data transfer.
- Implement a Kafka consumer application to subscribe to the Kafka topics and consume the streamed data in real-time, processing and analyzing it for further downstream tasks.

4. Model Development:

- Algorithm Selection: Choosing Random Forest and XGBoost classifiers due to their superior performance in handling complex datasets and their robustness against overfitting.

- Random Forest Classifier: Choose RFC because Capable of handling large datasets with high dimensionality. Provides a measure of feature importance, which is useful for understanding the influence of different factors on pit stop decisions. Handles both regression and classification problems, making it versatile.
- XGBoost Classifier: Known for its high efficiency and predictive accuracy, especially with structured data. Incorporates regularization techniques to prevent overfitting. Provides built-in handling of missing data and robust feature importance measures.

5. Visualizations

- Utilize libraries such as Matplotlib, Seaborn, or Plotly to create a variety of visualizations and graphs depicting various race strategy insights, ensuring clarity, accuracy, and aesthetics.
- Generate visualizations showcasing key metrics such as optimal pit stop windows, tire wear trends, fuel consumption patterns, and competitor positions, selecting appropriate chart types and color schemes for effective communication.
- Design visually appealing and intuitive visualizations that convey insights effectively to race strategists, considering factors such as readability, interactivity,
- and accessibility for different types of users.

6. Web Application Development:

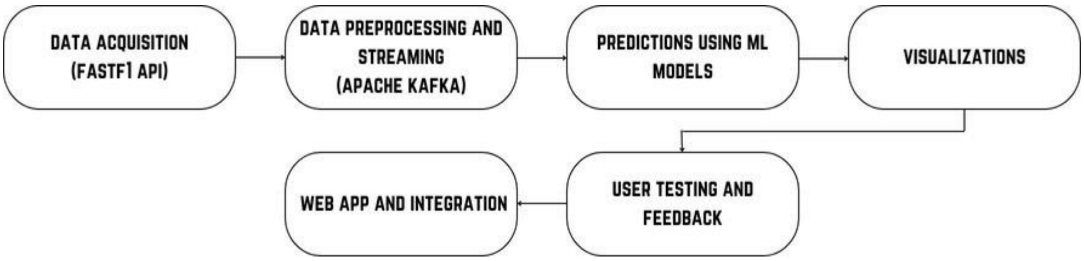
- Choose a suitable web development framework such as Flask,to develop a web application that will serve as the dashboard for accessing and analyzing the generated visualizations.
- Implement the necessary backend functionality to integrate the visualizations into the web application, including data retrieval, processing, and visualization rendering.

7. Integration and Deployment:

- Integrate the components of the system, including Kafka producers and consumers, machine learning models, visualization scripts, and the web application, into a cohesive system architecture.
- Conduct thorough testing of the deployed system to identify and address any issues or inconsistencies, ensuring that all components function as intended and meet the specified requirements.

8. User Testing and Feedback:

- Collect and analyze feedback from users regarding their experience with



the system, including likes, dislikes, suggestions for improvements, and any issues encountered during usage.
- Iterate on the feedback received, incorporating necessary changes and enhancements to refine the system further and improve user satisfaction, prioritizing features and improvements based on user feedback and business requirements.

Results and Conclusions

The machine learning models for optimizing race strategy were evaluated using Accuracy and F1-Score. The Random Forest model achieved 98% accuracy and a 0.50 F1-Score, indicating high predictive accuracy but some limitations in precision and recall. XGBoost showed 97% accuracy with a higher F1-Score of 0.61, suggesting a better balance between precision and recall. Despite the slight accuracy edge of Random Forest, XGBoost's higher F1-Score makes it more reliable for minimizing false positives and negatives. These models, integrated into a real-time system, provide precise, data-driven pit-stop recommendations, enhancing decision-making and offering a competitive edge in Formula One.

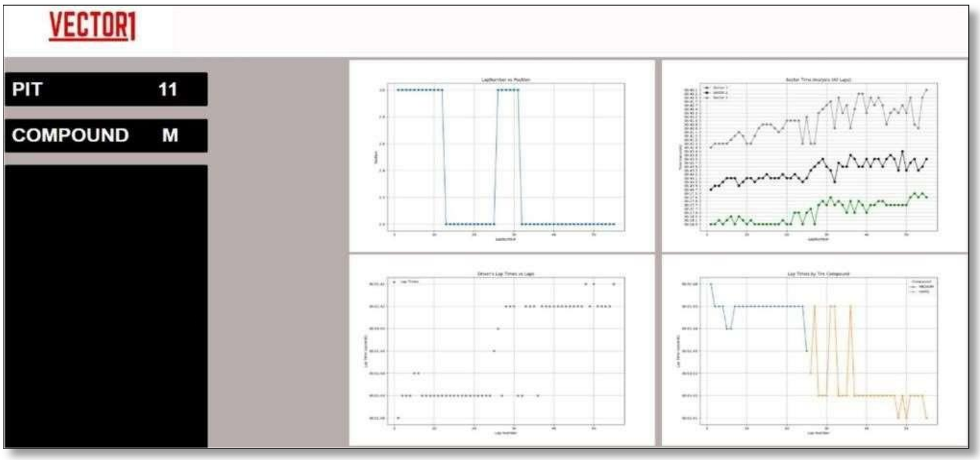


Figure1Dashboard

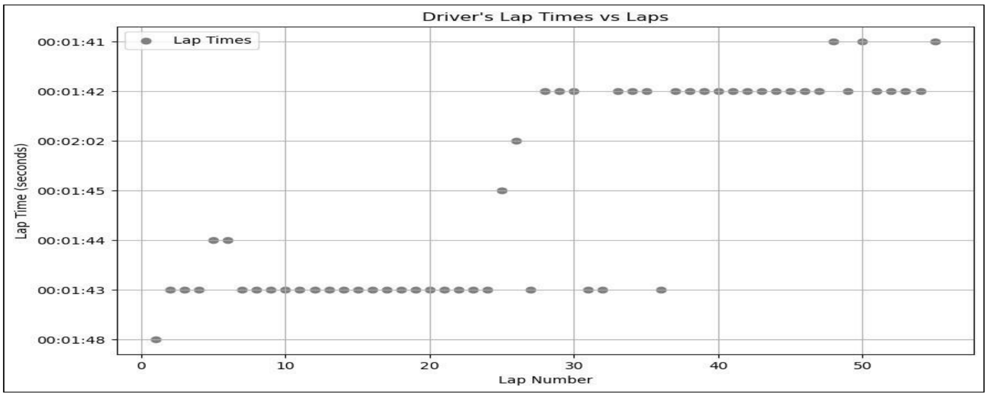


Figure 2 **Graph Showing Drivers Lap Times Vs Laps**

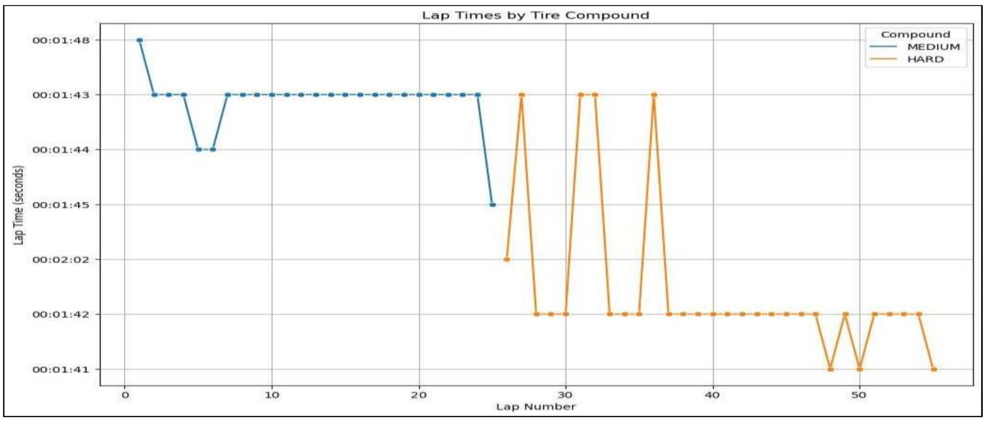


Figure 3 Graph Showing Drivers Lap Times Vs Laps

Innovation in the project

This project represents a significant innovation in Formula One race strategy by integrating advanced machine learning models with expert human intuition. By leveraging historical race data and real-time telemetry, the project optimizes critical decisions like pit-stop timings and tyre selections. Utilizing models such as Random Forest and XGBoost, it provides highly accurate and balanced predictive capabilities. This innovative approach surpasses traditional intuition-based methods, offering precise, data-driven recommendations. The real-time adaptability of the system ensures it can respond dynamically to changing race conditions, revolutionizing how teams strategize and enhancing their competitive edge.

Scope for future work

Future enhancements for the project include the development of advanced simulation techniques to run various race scenarios in real-time, allowing teams to virtually test different strategies before implementation. These simulations will refine strategy recommendations and prepare for unexpected race conditions. Adaptive learning techniques will continuously update models with new data from each race, ensuring they remain accurate and relevant amidst evolving race dynamics, track changes, and new car technologies.

Generative Adversarial Networks (GANs) will be used to generate simulated full race scenarios as training data, capturing complex race dynamics to enhance model training and robustness. Reinforcement Learning (RL) techniques, framed within a Markov Decision Process (MDP), will be applied to learn optimal policies, focusing on long-term success rather than immediate gains.