UNIVERSITY *of York*

This is a repository copy of *AutoF1: An RNN-Based Approach to Simulating Strategic Decision-Making in Formula 1*.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/id/eprint/229478/

Version: Accepted Version

Proceedings Paper:

Mistry, Vivek and Smith, William Alfred Peter orcid.org/0000-0002-6047-0413 (Accepted: 2025) AutoF1: An RNN-Based Approach to Simulating Strategic Decision-Making in Formula 1. In: The IEEE International Conference on Intelligent Transportation Systems (ITSC) 2025. IEEE Intelligent Transportation Systems Society Conference 2025, 18-21 Nov 2025 , AUS (In Press)

Reuse
This article is distributed under the terms of the Creative Commons Attribution (CC BY) licence. This licence allows you to distribute, remix, tweak, and build upon the work, even commercially, as long as you credit the authors for the original work. More information and the full terms of the licence here: https://creativecommons.org/licenses/

Takedown
If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

White Rose
university consortium
Universities of Leeds, Sheffield & York

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk/

AutoF1: An RNN-Based Approach to Simulating Strategic Decision-Making in Formula 1

Vivek Mistry
Department of Computer Science
University of York
York, UK
https://orcid.org/0009-0005-2243-7961

William A. P. Smith
Department of Computer Science
University of York
York, UK
william.smith@york.ac.uk

***Abstract*—This paper addresses the challenges of providing real-time decision support for Formula One strategy engineers concerning car setup and tyre changes through the development of an AI-driven simulation tool. We investigate this by applying machine learning techniques, specifically recurrent neural networks, to historical race data obtained via the FastF1 Python library. This research demonstrates the process of structuring detailed race data for neural network training to build a simulation model capable of predicting race outcomes and aiding strategic decisions. The evaluation showed the simulation model's performance against historical races and illustrated its application through case studies simulating alternative strategies, which show efficient simulation capabilities. Finally, we explore future directions for enhancing the model's granularity and exploring advanced architectures were discussed.**

***Index Terms*—Machine Learning, Decision Support Systems, Time Series Analysis, Predictive Modelling, Simulator**

I. Introduction

Formula 1 (F1) is a complex, interdisciplinary sport where strategy engineers are key to decision-making throughout a race weekend. Their primary responsibility is to plot the best path to a Grand Prix finish by optimising car setup and pit stop timing [19]. The optimal pit window minimises time loss relative to competitors. Teams can access different tyre compounds for dry and wet conditions [13]. Qualifying sees soft tyres used for fastest laps, while hard tyres offer longevity, and mediums balance performance and durability. Optimising car setup involves balancing straight-line speed and cornering performance for each circuit [16].

Most F1 weekends include three free practice sessions for setup adjustments, focusing on long runs for race pace and short runs for qualifying simulation [17]. Teams use in-house strategy models, updated weekly [22], but these are proprietary. This presents an opportunity to research how to use publicly accessible data for community tools.

Pre-race, engineers analyse conditions to plan strategies, dynamically re-evaluated per lap. Aggressive strategies offer high rewards but increased risk [19]. A mandatory compound change rule requires using at least two different tyre types unless it rains. Strategic factors influencing pit stops include the undercut, overcut, DRS train, Virtual Safety Car (VSC), Safety Car (SC), and Red Flag [7].

Sprint Weekends have featured modified formats over recent seasons. Initially, the Sprint race determined the starting grid for the main Grand Prix, with limited practice time available [6]. From 2023 onwards, a revised format was introduced: a single practice session is followed by traditional qualifying (setting the Grand Prix grid), while a separate Sprint Shootout determines the Sprint race grid resulting in less practice data.

F1 is a data-driven sport where metrics offer a competitive edge. The FastF1 Python library provides accessible race data. Combining this with Machine Learning (ML) allows efficient insight extraction. Successful implementation could yield reputational and financial benefits. In 2023, a 3-point gap cost Mercedes an estimated £6 million in prize money [18]. ML could help teams understand and close such margins.

II. Related Work

While machine learning literature in F1 strategy is limited, game theory and simulation offer modelling approaches.

Aguad and Thraves [1] model a race as a two-player zero-sum feedback Stackelberg game using dynamic programming, focusing on pit strategy with three dry tyre choices and lifespan constraints. The Lemke-Howson algorithm determines an initial Nash equilibrium. Lap time is a function of a ghost lap, an interaction function, and pit stop time, with linear fuel loss and stochastic modelling for yellow flags. The objective is to maximise the time gap or winning odds. Limitations include the two-player simplification (e.g. Hamilton's double overtake at Silverstone 2022 double overtake [11]), the assumption of teams seeking equilibrium, and the formulaic approach lacking adaptability (e.g., change in DRS rule change in 2024 [9]). Machine learning could offer a more adaptive, end-to-end learning solution.

Heilmeier et al. [14] developed simulation software categorising racing strategy, requiring manual inputs such as pit stop timing and track status updates. Driver style and energy consumption are incorporated directly into the performance equations. Unlike combinatorial game theory approaches, the simulation prioritises computational efficiency by first calculating a base race time without competitor interaction. This is based on equations for base lap time, tyre degradation (modelled linearly or logarithmically), fuel mass effects, car and driver characteristics, and pit stop durations. The model accounts for mass sensitivity (the performance gain from fuel weight reduction) and includes a handicap system (adjusting for starting position disadvantage). Competitor interaction and DRS effects are later introduced through simplified rules. Although efficient, the use of manual equations may limit the model's ability to capture dynamic race conditions, suggesting that neural networks could improve prediction accuracy by learning complex patterns directly from data.

Garcia [12] uses regression on tyre wear and formulates a Mixed-Integer Quadratic Programming (MIQP) problem to predict lap times. Linear regression identifies statistically significant variables. Outlier analysis is performed, though the removal of outliers caused by yellow flags limits the model's ability to handle such events. The MIQP optimises remaining race time based on hypothetical future laps with soft tyres, using priors from previous years' races, which may be irrelevant due to changing conditions (e.g., weather). MIQP requires predefined objective functions and constraints, whereas machine learning autonomously identifies trends.

NNs have applications in sports for betting, fans, commentators, and teams. F1's complexity distinguishes it from simpler multi-class classification tasks (e.g. football).

Choo [3] uses NN's for NASCAR pit stops, modelling the race with outputs and caution periods, evaluating pit crew performance. Analysis links lower stop times to leading drivers. The model uses the "average previous rate of change in rank," assuming continued overtaking, which may not hold in F1 due to tyre degradation and car performance differences. NASCAR's repeatable data and single tyre compound simplify modelling compared to F1.

Heilmeier et al. [15] applied ANN's for F1 strategy, noting the necessity of modelling driver interaction. Two NN's were built: one for pitting decisions and one for tyre choice. Issues with realistic race events, like removing late pit stops and only including dry tyres, were noted. A feed-forward NN struggled with gradual race changes, prompting the introduction of an RNN and a hybrid FFNN-LSTM model. However, alternative regression approaches or loss functions were not explored.

This review raises several questions:

- Is tyre degradation linear in our dataset?
- How do track degradation levels in our data compare to existing categorisations [15]?
- How can pit stops be effectively modelled using our data, and are existing equations necessary [14]?
- Can the claimed 15% increase in winning odds from strategic racing [1] be validated with our dataset?

III. Methodology

This section covers the dataset, model layers, and training process. Existing research does not address multiple strategy areas (e.g., car setup and pit stops) under all yellow flag and weather conditions. To fill this gap, we developed an RNN-based simulator for use during races and in post-race analysis.

A. Dataset

The dataset was sourced using the FastF1 Python package [8]. Two additional variables were introduced to the Fastf1-derived dataset: DistanceToDriverBehind and DriverBehind,

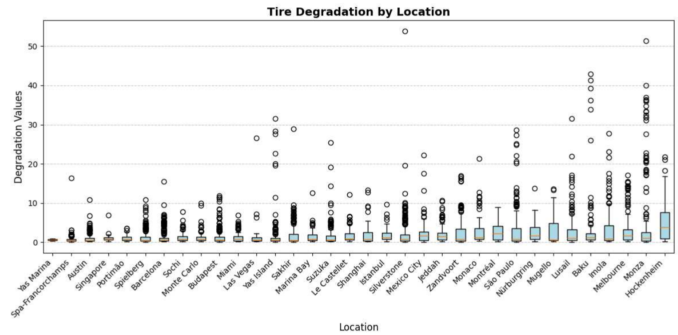

Fig. 1. Box plot showing average tyre degradation between laps across different race locations

derived from existing forward-distance data to support undercut and overcut decisions by representing the gap and identity of the following driver; and MandatoryPitStopMade, a boolean indicating whether a tyre compound change had occurred since race start, defaulting to true for non-race sessions or when wet-weather tyres were used. To ensure the most recent data, we selected the last telemetry record for each lap. Each record represents a single lap from one session. The script was run for the 2018-2024 seasons, as FastF1 began supporting this data format in 2018 [5], with 2024 being the last complete season at the time of starting implementation.

B. Data Exploration

We first examined if the track categories from the 'Virtual Strategy Engineer' paper [15] were evident in our dataset by analysing the mean absolute lap time differences within stints longer than two laps (Figure 1).

Their categories, based on tyre stress with hard compounds, did not align with our findings. For instance, Hockenheim showed the largest degradation IQR in our data, but it is a Group 2 track in the VSE classification. This suggests that predefined track categories might be too simplistic, and individual track encoding is more appropriate for our model.

Next, correlation analysis between all variables and lap time (Table I) revealed weak linear relationships (values in range +/- 0.15). Spearman's analysis showed stronger monotonic relationships, particularly negative correlations between speed and lap time (-0.43 to -0.25) and stronger positive correlations for variables like air pressure.

Analysing the correlation between pit stops and other variables (Table II) showed weak positive Spearman correlations with distance to surrounding drivers and weather factors, and mild negative correlations with lap time and track status. Pearson's analysis showed similar but weaker trends, with Country and DRS appearing more influential. TyreLife showed a stronger negative linear correlation with pit stops.

To assess the claim about strategic vs. myopic driving increasing winning odds by over 15% [1], we analyzed the impact of undercuts and overcuts by plotting the distance to surrounding drivers against positions gained (Figures 2 / 3). This analysis, limited by the qualitative nature of "strategic" and "myopic," showed no strong correlations between

Variable	Pearson	Spearman
Driver	-0.001846	-0.005538
SpeedI1	-0.349522	-0.439104
SpeedI2	-0.280270	-0.254116
SpeedFL	-0.354418	-0.305291
SpeedST	-0.372980	-0.344212
Compound	0.056331	0.062558
TyreLife	-0.182849	-0.229598
FreshTyre	0.001424	0.006129
Team	0.000532	0.004891
TrackStatus	0.033097	0.192804
Brake	0.082295	0.098653
DRS	0.076152	0.087627
Status	0.154935	0.195175
DistanceToDriverAhead	0.104102	0.078061
DistanceToDriverBehind	-0.003691	-0.085292
AirTemp	0.013709	0.043271
Humidity	0.066608	0.076483
Pressure	0.141165	0.188019
Rainfall	0.009263	0.015955
TrackTemp	-0.092103	-0.160489
WindDirection	0.069916	0.098773
WindSpeed	0.034785	0.067747
Country	0.101551	0.132520
Location	-0.079837	-0.136594

TABLE I
CORRELATION COEFFICIENTS FOR LAP TIME

Variable	Pearson	Spearman
LapTime	-0.079714	-0.047071
Driver	0.000162	0.001420
SpeedI1	0.021321	0.020408
SpeedI2	0.028720	0.021574
SpeedFL	-0.009778	-0.013145
SpeedST	0.036932	0.020987
Compound	0.007473	0.010813
TyreLife	-0.028400	-0.028177
FreshTyre	0.018117	0.018761
Team	0.019477	0.019407
TrackStatus	-0.011317	-0.022550
Brake	0.003698	0.005687
DRS	0.039837	0.016848
Status	-0.006905	-0.013746
DistanceToDriverAhead	0.013273	0.024737
DistanceToDriverBehind	0.008767	0.028064
AirTemp	-0.008202	-0.003584
Humidity	-0.004397	-0.016209
Pressure	-0.006094	-0.001354
Rainfall	0.004103	-0.009794
TrackTemp	-0.001314	0.003343
WindDirection	0.006259	0.007083
WindSpeed	0.015743	0.023324
Country	-0.002428	-0.007803
Location	-0.000886	0.000600

TABLE II
CORRELATION COEFFICIENTS FOR PIT STOPS

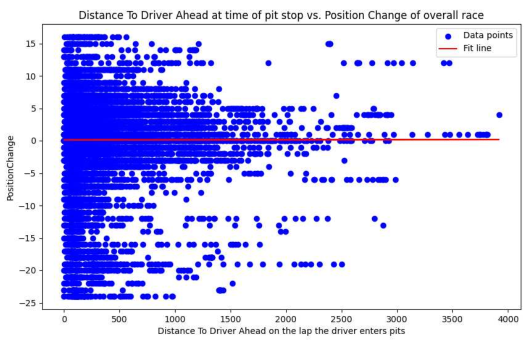

Fig. 2. Distance to driver ahead variable at the time of pit stop against position change at the end of the race.

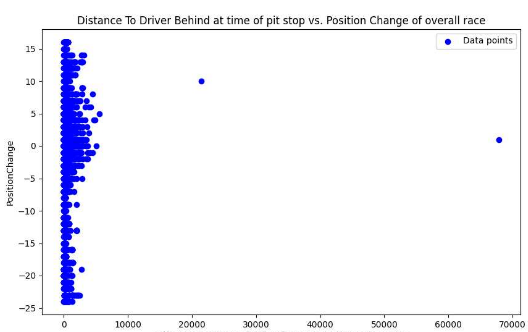

Fig. 3. Distance to driver behind variable at the time of pit stop plotted against position change at the end of the race.

immediate positional changes after pit stops and overall race outcomes. The inability of NumPy to fit a linear model to 'DistanceToDriverBehind' further highlights the complexity. This suggests that race strategy involves more than just reactive decisions and that the stated 15% increase is difficult to quantify based on immediate pit stop outcomes.

The overall analysis indicates that simple linear or monotonic relationships are insufficient to model the data, supporting the use of deep learning.

C. Model Design

Our model design choices are as follows: categorical features such as Team, Status, Driver, and Compound are encoded into learned vectors to capture relationships and enhance input representation. The model predicts dynamic next-lap data using the full current lap dataset, allowing evaluation of theoretical sequences. We compare LSTM and GRU time-series models, followed by a fully connected layer, with evaluation provided in the next section.

The choice between LSTM and GRU arises from findings that GRU's excel on less complex sequences, while LSTM's handle more intricate ones [2]. This suggests the optimal model depends on the sequence complexity, making it pertinent to evaluate if the added complexity of an LSTM offers a significant advantage in this application.

The core architecture, based on standard Recurrent Neural Network components such as LSTM or GRU layers, was specifically adapted to address the inherent complexities of Formula 1 race data. This adaptation primarily focused on enabling a multi-task learning structure capable of concurrently predicting various race metrics. Furthermore, the design prioritised real-time inference capabilities, ensuring the model's practical applicability for on-the-fly strategic decision-making during a race.

D. Data Transformations and Filtering

The models will be trained on four subsets of the local dataset, defined as follows:

1) No excluded data.
2) Exclude drivers who did not finish (DNF).
3) Exclude drivers who did not score any points.
4) Exclude drivers who lost positions during the race.

This approach, inspired by Choo's NASCAR model [3], uses a custom dataset class that is inherited from PyTorch's dataset class. The custom class overrides specific methods and includes an integer parameter to select the desired subset based on the numbers above.

As the most recent data available to teams, sessions before the race should be aggregated using statistical methods to enhance predictions, including aggregating practice sessions. We also exclude sprint race data, which may contain outliers from yellow flag events, and focusing on practice and qualifying runs, the analysis remains centred on the most competitive and relevant aspects of a race weekend. This approach ensures a consistent basis for comparing driver performance across different weekends and formats while avoiding the biases and inconsistencies introduced by sprint races, which have a fundamentally different structure and strategy. If a Sprint Shootout occurred, it is used over Qualifying due to better tyre availability, likely reflecting a more optimal lap.

The ClassifiedPosition field was used to identify DNFs, and the Points column filtered out non-scoring drivers in some dataset variants. Under previous F1 regulations, multiple soft compounds like "ULTRASOFT" and "SUPERSOFT" were available, but for this dataset, they have been consolidated into a single "SOFT" category to simplify analysis while preserving their general performance characteristics. Races with missing tyre data were excluded. Lap times were converted to seconds. Telemetry features (e.g., speed, throttle, gear) were imputed using the nearest available value first, falling back to the mean by location if unavailable. Environmental variables (e.g., temperature, rainfall) were handled similarly, using nearest values followed by mean imputation. Team names were standardised across seasons (e.g. name change between seasons). Status codes were consolidated into related groups—such as finished, finished with laps down, mechanical issues, human error, and illness—to reduce label sparsity and improve model interpretability by avoiding fragmentation across 70 distinct statuses.

Tyre usage analysis of the training data revealed that 95% of laps occurred in dry conditions, and 5% in wet conditions.

E. Representation of Pit Stop

The significant class imbalance in pit stop decisions (approximately 1:30 no-pit to pit) necessitates careful representation. We test binary classification (pit/no-pit) using weighted binary cross-entropy and Focal Loss to address the sparsity. An alternative is a continuous "laps till pit" representation, which we discretise into intervals. For this, we compare standard regression with mean squared error to ordinal regression to capture the sequential nature of pit stops.

F. Handling Gear and Compound Imbalance

To address the imbalance in the multi-class classification of gear and tyre compound, we evaluate the impact of using class weights with cross-entropy loss. These weights will emphasise underrepresented classes to improve prediction accuracy for minority categories.

G. Multi-Task Learning

Balancing different loss functions in the multi-task learning framework is crucial. We explore dynamic loss weighting, where weights adapt during training to ensure equitable improvement across all predicted variables. Additionally, we investigate PCGrad [21], a gradient manipulation technique that projects conflicting gradients to promote more aligned learning across the different prediction tasks.

IV. Model Development

This section explains the results found from the decisions explained in Sections III-E, III-F and III-G. A full implementation is available in our Github repository [20].

A. Pit Representation

The AUC score is used to evaluate the methods described in Section III-E because it measures a model's ability to distinguish between classes without being biased by class distribution, unlike accuracy. It focuses on ranking predictions and performs well across various decision thresholds, making it a reliable metric in this context.

Only weighted binary cross-entropy achieved an AUC above 50%, indicating it predicted both classes. Other methods were biased toward always predicting the "no pit" class. Focal Loss is not appropriate in this case as it's difficult to distinguish between "hard" and "easy" examples—each race contains only 1–3 pits, making all instances similarly challenging. Regression-based methods, such as using Mean Squared Error and Ordinal Regression, may be unsuitable for this task as they predict continuous values rather than addressing the binary classification problem, and may struggle with class imbalance, leading to poor performance.

B. Handling Gear and Compound Inbalance

We evaluated performance using the F1 score, which better reflects precision and recall in imbalanced tasks, and analysed confusion matrices to assess false positives and negatives. Experiments without class weights showed the model defaulting to common classes (medium or hard tyres, gears 1–2). After normalising the weights (making sure the weights sum to 1), the model began predicting less frequent classes, such as intermediate tyres and lower gears (2/3).

Using unnormalised weights (values greater than 1) made training unstable by amplifying loss from rare classes, which hindered optimisation. Normalising weights helped balance class importance without destabilising learning, resulting in more accurate predictions.

C. Multi-Task Learning

We experimented with automatic loss weight selection. While compound predictions and most discrete variables improved, PitNow and Gear remained flat and underperformed compared to earlier experiments in Sections IV-A and IV-B. This may be because the approach was either not suited to the current hyperparameters or not suitable for our model. Next, we tested the Gradient Surgery method without the PitNow variable. Results showed reasonable performance for compound and distance ahead, but continuous variables remained unstable. This could be due to favourable gradient alignment for compound and distance, while gear and others lacked this benefit. As a result, the final model uses manual weighting.

Final weights were determined through a focused manual search, incrementally adjusting values until no further improvement was observed, thereby ensuring certain metrics did not disproportionately influence the overall loss.

V. Evaluation

A. Decision Tree

To establish a baseline for visualising the improvements from using Neural Networks, a decision tree model was trained using the scikit-learn library. The results are shown in Table III.

B. Against Ground Truth

The loss functions, as previously discussed, have been applied, with Huber Loss chosen for all continuous variables due to its combination of the benefits of Mean Absolute Error and Mean Squared Error. A ReduceLROnPlateau scheduler was used, but the learning rate was never reduced, suggesting

TABLE III
DECISION TREE PERFORMANCE ON THE VALIDATION DATASET

Metric	Dataset 1	Dataset 2	Dataset 3	Dataset 4
Accuracy (Compound)	0.5316	0.7132	0.7766	0.7359
Accuracy (Gear)	0.4099	0.3879	0.3541	0.3616
AUC Score (Pit Decision)	0.5000	0.5000	0.5000	0.5000
MSE (Lap Time)	0.0123	0.0116	0.0108	0.0115
MSE (Speed I1)	0.0482	0.0472	0.0490	0.0454
MSE (Speed I2)	0.0160	0.0159	0.0144	0.0155
MSE (Speed FL)	0.0314	0.0310	0.0298	0.0254
MSE (Speed ST)	0.0041	0.0036	0.0034	0.0037
MSE (Speed Telemetry)	0.0224	0.0223	0.0217	0.0218
MSE (RPM)	0.0791	0.0737	0.0745	0.0040
MSE (Throttle)	0.0116	0.0091	0.0091	0.0100
MSE (Distance Ahead)	0.0000	0.0000	0.0000	0.0000
MSE (Distance Behind)	0.0050	0.0051	0.0051	0.0045
Runtime (s)	7345.01	7345.01	7281.93	3372.97

TABLE IV
GRU PERFORMANCE ON THE VALIDATION DATASET

Metric	Dataset 1	Dataset 2	Dataset 3	Dataset 4
Accuracy (Compound)	0.9742	0.9757	0.9650	0.9675
Accuracy (Gear)	0.8534	0.8242	0.7974	0.8103
AUC Score (Pit Decision)	0.6580	0.6619	0.6352	0.6374
MSE (Lap Time)	0.0001	0.0000	0.0001	0.0001
MSE (Speed I1)	0.0161	0.0170	0.0162	0.0212
MSE (Speed I2)	0.0134	0.0152	0.0140	0.0153
MSE (Speed FL)	0.0110	0.0092	0.0100	0.0082
MSE (Speed ST)	0.0093	0.0157	0.0131	0.0140
MSE (Speed Telemetry)	0.0144	0.0209	0.0163	0.0159
MSE (RPM)	0.0106	0.0106	0.0108	0.0132
MSE (Throttle)	0.0203	0.0264	0.0206	0.0223
MSE (Distance Ahead)	0.0643	0.0610	0.0608	0.0645
MSE (Distance Behind)	0.0001	0.0001	0.0001	0.0001
Runtime (s)	7345.01	7345.01	7281.93	3372.97

the model continued improving on the validation set. However, gear and compound outputs began to overfit with extended training, indicating they had been learned earlier. Gradient norms from each fully connected output layer were used to derive initial loss weights, normalised to sum to 1.0. This approach helps balance the learning signal across layers, potentially promoting more stable and steady improvements during training by preventing any single output layer from dominating the loss. This gave a more balanced starting point and led to better convergence, then arbitrarily scaling underperforming outputs.

To evaluate generalisation, we used an external holdout approach, training on seasons 2018–2023 and testing on 2024 data. The results of training can be seen in Tables IV and V, where all combinations produce similar results. To evaluate whether the model can theoretically make better strategies, we can take the first input from each tensor in the testing set and run it auto-regressively. This involves using the generated output and replacing it with the actual data. If the test were to be rerun, by only using a random split once, and using the same split across all datasets. Runtimes vary due to shared GPU cluster use; reported values are approximate and may be affected by external job load and resource contention.

C. Autoregressive Model

The autoregressive runs (results in Tables VI and VII) indicate that the two architectures respond differently to the datasets. For the GRU model, dataset 4 performs best, followed by dataset 1. Since dataset 4 includes only races where the driver gained positions, this suggests that the GRU struggles to distinguish between good and bad strategies. Its strong performance on dataset 4 implies that it learns best from consistently positive outcomes rather than assessing strategic quality. In contrast, the LSTM performs best on dataset 2 (races where the driver did not DNF), with dataset 1 following closely. This suggests that the LSTM is better at differentiating between good and bad strategies. Its two-cell structure likely enables it to retain long-term dependencies more effectively than the GRU, which may explain why it improves strategies even in cases where drivers previously crashed.

TABLE V
LSTM PERFORMANCE ON THE VALIDATION DATASET

Metric	Dataset 1	Dataset 2	Dataset 3	Dataset 4
Accuracy (Compound)	0.9675	0.9727	0.9525	0.9374
Accuracy (Gear)	0.8103	0.8041	0.8146	0.8428
AUC Score (Pit Decision)	0.6580	0.6875	0.6311	0.6021
MSE (Lap Time)	0.0001	0.0001	0.0003	0.0006
MSE (Speed I1)	0.0212	0.0313	0.0312	0.0296
MSE (Speed I2)	0.0153	0.0141	0.0229	0.0158
MSE (Speed FL)	0.0082	0.0157	0.0185	0.0077
MSE (Speed ST)	0.0140	0.0131	0.0238	0.0122
MSE (Speed Telemetry)	0.0159	0.0161	0.0170	0.0193
MSE (RPM)	0.0132	0.0113	0.0160	0.0113
MSE (Throttle)	0.0223	0.0292	0.0265	0.0247
MSE (Distance Ahead)	0.0645	0.0529	0.0675	0.0624
MSE (Distance Behind)	0.0001	0.0001	0.0006	0.0005
Runtime (s)	3702.83	6960.69	3997.56	3538.13

TABLE VI
GRU PERFORMANCE ON TESTING DATASET IN AUTOREGRESSIVE MODE

Metric	Dataset 1	Dataset 2	Dataset 3	Dataset 4
Total Simulations	158	139	77	67
% Faster Simulations	65.19%	48.20%	23.38%	95.52%
Mode of Pits	66	65	65	68
Mean of Pits	57.80	59.22	59.14	59.55
IQR of Pits	13.0	13.5	12.0	14.5
Mode of Compound Switches	0	0	1	0
Mean of Compound Switches	0.66	0.50	0.61	0.48
IQR of Compound Switches	1.0	1.0	1.0	1.0
% Wet Races with Wet Tyre Usage	57.14%	0.00%	0.00%	0.00%
(Sample Size) Wet Races with Wet Tyre Usage	21	13	6	7
% Simulations Leading to Disqualification	46.84%	50.36%	33.77%	46.27%
% Pit Stop Input leads to Compound Change	2.03	0.80	2.16	1.54
(Sample Size) Pit Stop Input leads to Compound Change	296	250	139	130

TABLE VII
LSTM PERFORMANCE ON TESTING DATASET IN AUTOREGRESSIVE MODE

Metric	Dataset 1	Dataset 2	Dataset 3	Dataset 4
Total Simulations	158	139	77	67
% Faster Simulations	94.30%	96.40%	15.58%	17.91%
Mode of Pits	55	67	52	51
Mean of Pits	58.00	58.51	56.48	59.10
IQR of Pits	14.0	15.0	16.0	16.0
Mode of Compound Switches	1	1	0	0
Mean of Compound Switches	1.93	0.99	0.58	0.61
IQR of Compound Switches	2.0	1.0	1.0	1.0
% Wet Races with Wet Tyre Usage	29.41%	35.71%	0.00%	0.00%
(Sample Size) Wet Races with Wet Tyre Usage	17	14	10	9
% Simulations Leading to Disqualification	24.68%	34.53%	42.86%	35.82%
% Pit Stop Input leads to Compound Change	5.17	4.08	0.0	0.0
(Sample Size) Pit Stop Input leads to Compound Change	290	245	134	118

Across all models, the mode, mean, and IQR of pit decisions remain high, indicating room for improvement, either through model optimisation or architectural changes. However, the lower mean, mode, and IQR of compound switches suggest that pit stops do not always correspond directly to tyre changes.

The LSTM structure results in the fewest disqualifications and shows a higher mode for compound switches compared to the GRU. However, the lowest percentage recorded is

24.68%, suggesting that the 'Mandatory Pit Stop' variable was insufficient to ensure a pit stop occurred in roughly a quarter of cases.

Finally, the GRU on dataset 1 has the highest percentage of simulations using wet tyres for wet races. Since tyre changes in wet conditions are typically short-term decisions made immediately after detecting rainfall, the LSTM's ability to retain more long-term information is not necessarily an advantage in this scenario.

VI. Case Study

From this point onwards, we assume that the trained model represents an accurate simulator of the real world. If real data is provided, the model should produce results that closely match reality, and it should respond appropriately to counterexamples. We analyse two races as case studies, adjusting input data to explore how the model responds to changes in specific factors. The data used is not from the model's training dataset, as it was not available during development.

A. Yuki Tsunoda - Australia 2025

Here, we use the GRU model trained on Dataset 1, as Table VI indicates that this combination yields the most realistic results for wet race scenarios. Notably, this model had the highest percentage of wet tyre usage during wet conditions. This race was chosen because Tsunoda stated, "I did my best, but the weather didn't work in my favour—the timing and everything just didn't go my way" [10]. Simulating this race tests how the GRU model reacts to similar conditions, revealing whether it might have recommended a better strategy and avoided real-world mistakes.

The Australia 2025 data contains missing Lap Time values, even though "LapTime" is equal to the sum of "Sector1Time", "Sector2Time", and "Sector3Time". To address this, missing values were filled using this formula. However, this raises concerns about whether additional data was available during training, as any races with missing "LapTime" values were excluded from the process.

To ensure realistic comparisons, we initialised the model by running it up to the actual pit stop lap from Yuki's race, allowing the model to learn a representative hidden state. From that point, we began an autoregressive simulation. Predicted lap times were aggregated and subsequently scaled by a multiplier, derived from simulating real-world race conditions and aligning with Yuki's actual total race time, to ensure robust real-world performance alignment. Excluding DNFs in this simulation leads to scenarios that might not be achievable in real life, particularly given weather as a limiting factor. This simulation was rerun with pit stops at alternative laps for comparison, with the tyre variable fixed to 'intermediate' in each case.

This simulation (shown in Figure 4) reveals that pitting on lap 46 would have been too early, as rain only began on lap 48. Since intermediate tyres are slower than dry tyres in dry conditions, an early switch would have made overtaking easier for competitors while making it harder to regain positions later.

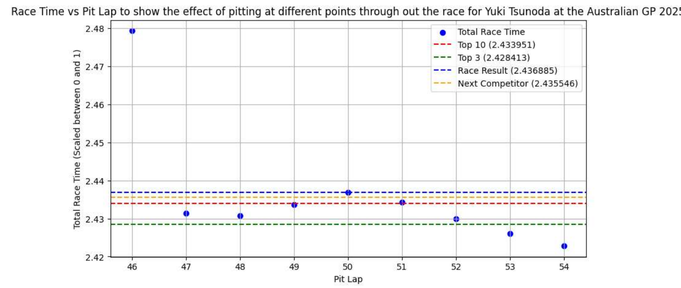

Fig. 4. Effect of changing Yuki Tsunoda's pit stop timing at the 2025 Australian GP, using scaled autoregressive simulations with the tyre type fixed to intermediate.

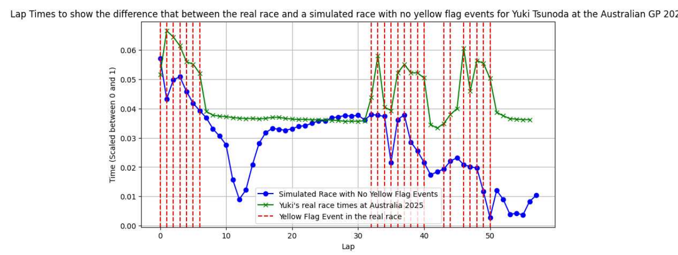

Fig. 5. Simulated race for Yuki Tsunoda at the 2025 Austrian Grand Prix, without yellow flag interruptions, using the same pit strategy and tyre changes as the actual event.

Interestingly, the results also suggest that staying out on dry tyres longer before pitting could have been a viable strategy. However, the risk of crashing due to reduced grip would have been a major concern. This distinction in the 'status' field—separating drivers who finished from those who crashed—is particularly relevant to the hypothesis that the inclusion of this field might still be problematic, either by providing future information to the model or by influencing how the model learns (or fails to learn) from crash events, even though DNFs were present in the dataset.

As this was a difficult race to create a strategy for, as there were 3 SC's, we simulated the race in autoregressive mode without any yellow flag events with the same pit decisions and tyre changes (shown in Figure 5).

Overall, the data reveals several key trends in Yuki's simulated race performance. In the opening laps, which were under a Safety Car in real life, lap times are noticeably faster due to not being speed-limited. Around lap 11, a significant drop in lap times suggests that he had clean air (i.e., when a car is not closely following another and thus benefits from optimal aerodynamic performance), meaning no drivers were ahead to slow him down. This is followed by a plateau, likely indicating the tyres were nearing the end of their optimal performance as degradation set in. Finally, during where there were yellow flag events in reality, Yuki's lap times improved, possibly due to reduced on-track battles and a clearer racing line.

B. Alex Albon - China 2025

Albon was selected for the next case study as in his post-race interview, he said "it was a strange race, I were so com-

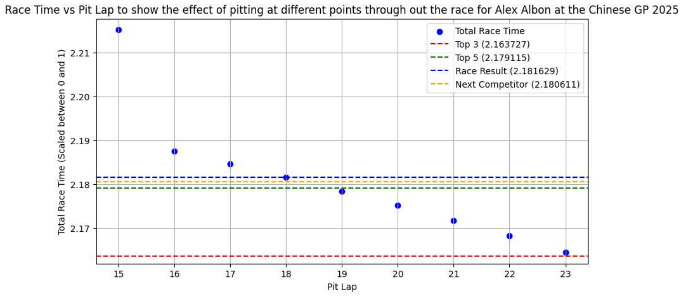

Fig. 6. Effect of changing Alex Albon's pit stop timing at the 2025 Chinese GP, using scaled autoregressive simulations with the tyre type fixed to hard.

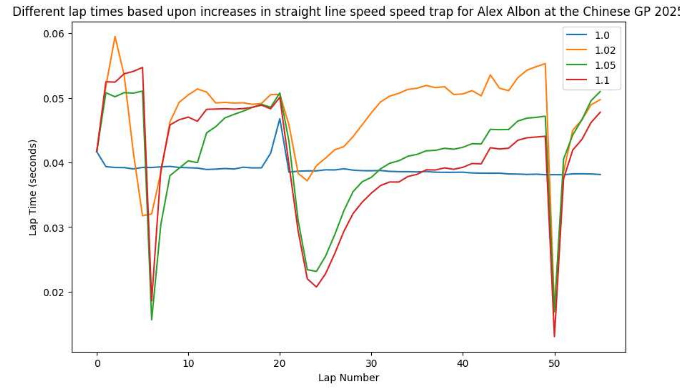

Fig. 7. Case study demonstrating the impact of adjusting Albon's straight-line speed during the 2025 Chinese Grand Prix, using autoregressive simulations with speed scaled by various multipliers.

fortable on the first stint that we kinda cornered ourselves, the hard tyre outperformed everyone's expectations but ultimately we stayed out too long" [4]. Hence, there are two parts to the investigation: changing speed simulations and verifying pit stop claims.

We followed the same method as for Yuki's example to generate the effect of pitting at different times, whilst fixing the transition to the hard tyre (seen in Figure 6). The figure contradicts Albon's hypothesis about staying out too long on the medium tyre, as the model suggests that in previous races, staying on the medium tyre for that long has generally been suitable. This is because overall lap times tend to decrease as the laps before the pit stop increase, likely due to tire degradation stabilising and the driver adapting to the conditions. However, Albon, as the driver, has direct feedback on the car's performance—something this model does not capture. Additionally, it does not account for the real-time strategies of competing drivers. This raises the question of whether the race has been analysed at a sufficient level of granularity, and whether incorporating driver feedback as an input could improve the model's effectiveness.

To further explore this simulation, Albon mentions his struggles with overtaking during the race. While speed is difficult to change in the short term due to the need for extensive research and development, a team might still want to simulate the effects of a faster car to assess whether investing time and money into improving that attribute is worthwhile. For this experiment, we increased the car's straight-line speed—an area where drivers typically have more overtaking opportunities—and analysed the effect on race time. To ensure proper comparison with real-life data, scaling has been applied to map the simulated results to the actual race conditions.

This simulation (shown in Figure 7) shows higher spikes in lap times across all speed increases, while the original speed has a lower range of results. This could indicate that the increased speed is causing faster tire degradation. Another possible explanation for these spikes is that, as Albon's speed increases, he encounters more competitors, which could result in lower lap times as he attempts to overtake them. Overall, while this analysis is based on a single race, it demonstrates that the model has potential. In a real team setting, decisions would ideally be informed by data from multiple races, given that each track presents unique characteristics and is only visited once per season.

VII. Conclusions

We have shown that an RNN trained on historic time series race data is capable of supporting motorsports strategy, from validating driver claims to simulating race scenarios. While initial results are promising, further improvements in model design, data completeness, and evaluation could significantly enhance performance and real-time decision-making capabilities. K-fold CV could be explored in future for additional robustness, and a more robust hyperparameter tuning process could be used. A software approach could also be taken to enforce hard constraints to ensure rule compliance to strengthen AutoF1's direct applicability in real-time strategic support. From an architectural perspective, Transformers could be a better fit due to their ability to use positional encoding, potentially recognising patterns like pit stops and yellow flags using the attention mechanism. Despite experimenting with alternative loss functions for pit decisions, none of them improved performance. The Virtual Strategy Engineer [15] used a single loss function without testing alternatives. Given this, implementing their hybrid model could improve performance for pit decisions. Further optimisation, including hyperparameter tuning, architectural changes, or adjusting granularity, could enhance the model and make it a more powerful tool for decision-making in motorsports strategy. Finally, input granularity could be improved by incorporating data from all competitors, enhancing performance by directly modelling interactions while optimising a single driver's race, rather than relying solely on historical data.

References

[1] Felipe Aguad and Charles Thraves. "Optimizing pit stop strategies in Formula 1 with dynamic programming and game theory". In: *European Journal of Operational Research* 319.3 (Dec. 2024), pp. 908–919.

[2] Roberto Cahuantzi, Xinye Chen, and Stefan Güttel. "A comparison of LSTM and GRU networks for learning symbolic sequences". In: vol. 739. arXiv:2107.02248 [cs]. 2023, pp. 771–785.

[3] Christopher Ledesma Weisen Choo. "Real-time decision making in motorsports : analytics for improving professional car race strategy". Thesis. Massachusetts Institute of Technology, 2015.

[4] *Drivers React After The Race — 2025 Chinese Grand Prix - YouTube*. URL: https://www.youtube.com/watch?v=PrwK2IHtJ-E (visited on 04/06/2025).

[5] *Error loading laps data for pre-2018 races · theOehrly/Fast-F1 · Discussion #550*. en. URL: https://github.com/theOehrly/Fast-F1/discussions/550 (visited on 01/21/2025).

[6] *Everything you need to know about the 2024 F1 Sprint format*. en. URL: https://www.formula1.com/en/latest/article/2024-f1-sprint-rules-format-explained.5pQvaAY52nnX9vZYAYgf3O (visited on 10/10/2024).

[7] *F1 strategy explained: What's an undercut, overcut, a DRS train and more*. en. June 2023. URL: https://www.autosport.com/f1/news/f1-strategy-explained-whats-an-undercut-overcut-a-drs-train-and-more/10488979/ (visited on 10/07/2024).

[8] *FastF1 3.4.3*. URL: https://docs.fastf1.dev/ (visited on 11/17/2024).

[9] *Five F1 Regulation Changes To Watch Out For in 2024*. en-US. URL: https://www.mercedesamgf1.com/news/five-f1-regulation-changes-to-watch-out-for-in-2024 (visited on 11/16/2024).

[10] FORMULA 1. *Drivers React After The Race — 2025 Australian Grand Prix*. Mar. 2025. URL: https://www.youtube.com/watch?v=AIG_Fm7cd68 (visited on 04/06/2025).

[11] FORMULA 1. *THROUGH GOES HAMILTON*. July 2024. URL: https://www.youtube.com/watch?v=FM7wyRivEJY (visited on 11/16/2024).

[12] Quiroga García and Mariana Paz. "Optimization of pit stop strategies in Formula 1 racing: a data-driven approach". MA thesis. Universidad de Chile, 2024.

[13] GPfans.com. *Everything you need to know about F1 tyres in 2024*. en. July 2024. URL: https://www.gpfans.com/en/f1-news/100347/f1-tyres-changes-compounds-rules-colours-blankets/ (visited on 11/16/2024).

[14] Alexander Heilmeier, Michael Graf, and Markus Lienkamp. "A Race Simulation for Strategy Decisions in Circuit Motorsports". In: *21st International Conference on Intelligent Transportation Systems (ITSC)*. Nov. 2018, pp. 2986–2993.

[15] Alexander Heilmeier et al. "Virtual Strategy Engineer: Using Artificial Neural Networks for Making Race Strategy Decisions in Circuit Motorsport". en. In: *Applied Sciences* 10.21 (Jan. 2020), p. 7805.

[16] *How Do You Set Up a Formula One Car?* en-US. URL: https://www.mercedesamgf1.com/news/how-do-you-set-up-a-formula-one-car (visited on 04/04/2025).

[17] *Insider's guide: Who does what in an F1 team?* en. Feb. 2022. URL: https://www.motorsport.com/f1/news/insiders-guide-f1-team-who-does-what/8025042/ (visited on 01/10/2025).

[18] Cambridge Kisby. *F1 prize money: How much do GP teams and drivers really make?* en-GB. Aug. 2024. URL: https://www.motorsportmagazine.com/articles/single-seaters/f1/f1-prize-money-how-much-do-gp-teams-and-drivers-really-make/ (visited on 01/10/2025).

[19] Tom McCullough. *So, you want to be an F1 strategy engineer?* en-GB. URL: https://www.astonmartinf1.com/en-GB/news/feature/so-you-want-to-be-an-f1-strategy-engineer (visited on 10/07/2024).

[20] Vivek Mistry. *mistryvivek/AutoF1-RNN-Race-Simulator*. original-date: 2024-10-07T12:18:06Z. Apr. 2025. URL: https://github.com/mistryvivek/AutoF1-RNN-Race-Simulator (visited on 04/24/2025).

[21] Wei-Cheng Tseng. *WeiChengTseng/Pytorch-PCGrad*. original-date: 2020-08-25T05:51:45Z. Mar. 2025. URL: https://github.com/WeiChengTseng/Pytorch-PCGrad (visited on 04/05/2025).

[22] *What is DRS and why does the device raise criticism? - The Athletic*. URL: https://www.nytimes.com/athletic/4266856/2024/02/23/formula-one-f1-drs-explained/ (visited on 10/10/2024).