Proceedings of the 4th International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/47/20241191

Deep Neural Network-based lap time forecasting of Formula 1 Racing

Zhixuan Zhao

School of Automation, Nanjing University of Information Science & Technology, Nanjing, Jiangsu, 210044, China

202083240033@nuist.edu.cn

Abstract. Making comparisons and analyzing players in the sporting world is extremely valuable. The media, coaching staff, and players all rely on this data to assess performance, develop strategies, and make critical decisions. Therefore, neural networks can be employed to create a practical system that uses previous years' data to predict future performance. This paper uses a Deep Neural Network (DNN) to predict the fastest lap time in qualifying for Formula 1 (F1) races. The network categorizes data to learn each driver's performance at each circuit and provides separate predictions. By doing so, it considers the unique characteristics of each driver and track, enabling more accurate predictions. The paper demonstrates that neural networks tend to have better performance and adaptability in such complex environments compared to traditional mathematical methods like linear regression. Neural networks can learn from the data and detect patterns that are difficult to capture with traditional methods. As a result, they can achieve a relatively precise prediction, providing valuable insights and decision-making support for coaches, drivers, and fans.

Keywords: Deep Learning, Formula 1 Racing, Neural Network.

1. Introduction

The prediction of race results is significant in comparing and analyzing different players in almost all sports events [1]. Particularly in Formula One races, it is beneficial to the press, the driver, and the rival if a prediction related to an incoming race is given, for a foreseeable prediction can be of vital importance to race analysis, strategy making, and self-assessment. Nowadays, predictions are mostly given by statistic or manual way. It is constructive and necessary to introduce a new method to give precise predictions based on machine learning.

Formula 1 (F1), represents the highest level of motorsport and can reach 60 billion views every year. It is meaningful to provide a prediction of how fast a driver can reach on track, considering the pole position mostly depends on who gets the fastest lap time in qualifying and can involve the final result [2].

In Formula One races, the fastest lap of a particular driver strongly correlates to the race car's pace and the track surface's condition. Meanwhile, such an abstract relationship is hard for common mathematic methods like polynomial regression to characterize and fit because of efficiency and universality issues [3]. Consequently, considering that Deep Neural Network (DNN) has outstanding performance in circumstances that require non-linear regressions, it can be utilized to seek hidden functional relationships of recent years' lap times of each driver. Therefore, a credible prediction for next year can be given [4,5]. Fully Connected Neural Network (FCNN), a kind of Deep Neural Network (DNN), is broadly utilized in regression prediction for its excellent adaptiveness and deep performance [6].

In the context that is claimed above, the data that the network needs to deal with are usually discrete, as shown in Figure 1. Therefore, compared to traditional statistic systems such as polynomial regression, a neural network has a stronger capability to seek hidden functional relationships. Therefore, using a neural network is a proper approach to give prediction.

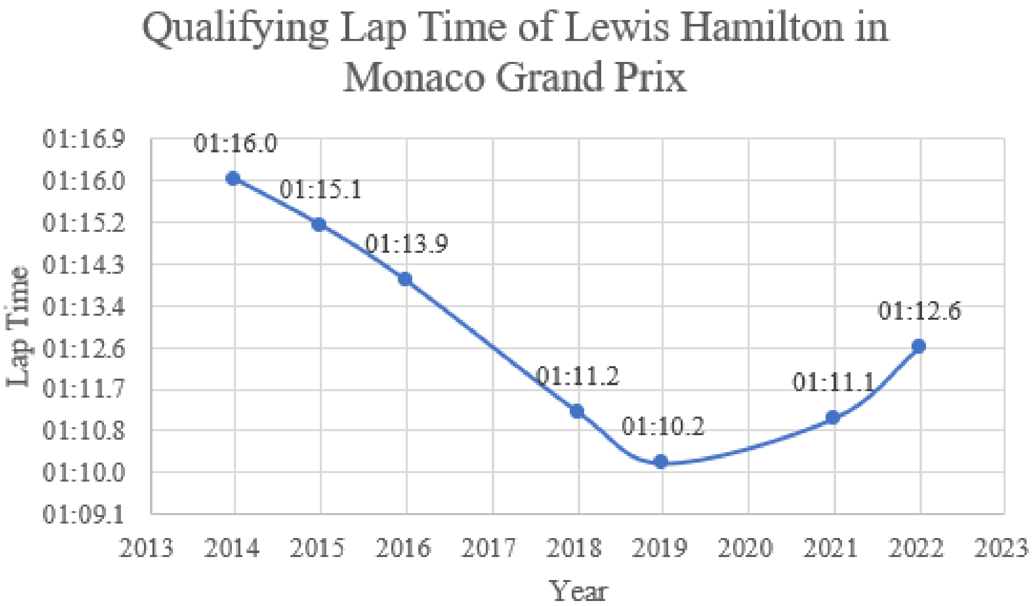

Figure 1. Qualifying Lap Time of Lewis Hamilton in Monaco Grand Prix (Figure credit: Original).

2. Method

2.1. Dataset

Following the principle of authenticity, trustworthiness, convenience, and feasibility in data collection, most of the data in this paper are from the Formula 1 World Championship (1950 - 2023) public dataset in Kaggle, which is used to train the neural network, and the rest of the data are collected independently and used for comparison and assessment of the prediction result. In addition, the data from Kaggle only include the race results until 2022. That is to say, the ongoing 2023 season is not included in the public dataset, so this part of data is collected from the F1 official website, which are used to compare with the prediction given by the neural network to evaluate the performance and accuracy [7].

2.2. Data Processing

The period of the data included in the dataset is over 30 years, and the circuits and drivers can change a lot between decades. Besides, the regulations differ every year, which causes the performance of race cars slightly changes over time. For that reason, the period of the data cannot be too long to prevent the decrease of data dependency.

In Formula 1 races, the technical regulation often revises a little if a major regulation change is not made. As a result, the training data for the network are all from 2014 onwards, which is considered the beginning of the "Hybrid Era" of Formula 1 [8]. Moreover, only the qualifying fastest lap times are included because the result of a qualifying session is only related to the fastest lap which every driver can achieve. In contrast to ranking the finishing position in the final race result, qualifying sessions can mostly show the absolute speed of each driver with the least interference, which can reach relative fairness as the standard to measure the pace of each driver.

As for the processing method of the lap time result in all three qualifying sessions in one race, Q1, Q2, and Q3, the minimum value method is taken. The minimum time is regarded as the input data.

After actual inspection, the lap time of each driver in each race track should be separately input and trained. Numbering the circuits and drivers and inputting them all into the neural network would not be a good approach. When dealing with some special tracks that are shorter than many of the others and need less time to finish, the network may regard these time data as outliers and neglect them. Additionally, whether to the performance or the training difficulty, a Single Input Single Output (SISO) system is more beneficial.

2.3. Network Structure

As what has been shown in Figure 1, the goal is to achieve univariate non-linear regression. Therefore, it is suitable to choose an FCNN which has an input layer, an output layer, and two hidden layers with 9 neurons in total. The specific network structure is displayed below in Figure 2.

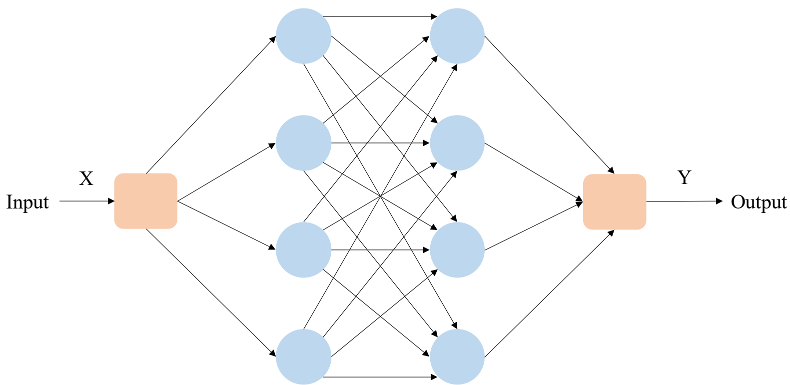

Figure 2. Network structure (Figure credit: Original).

The input, which is a year value like 2023, will be fed to this network. After going through 2 hidden layers, the original input data will be transferred and finally output a new value, which is a prediction lap time.

The model must be trained first before practical use. A bunch of input and output data that has a specific parallel relationship will be fed to the network so that the network can learn the correspondence and pattern between them, which is non-linear in this circumstance. For this reason, when dealing with problems the model has never met, such as how fast a driver can go in the year 2023, it can answer the pattern that it has learned before.

A neural network with enough depth can almost fit any continuous function [9], so DNN is a proper solution to this non-linear regression problem.

2.3.1. Forward Propagation. The activation function between hidden layers is Parametric Rectified Linear Unit (PReLU) and the activation function before the output layer is Hyperbolic Tangent Function (Tanh). This setting can ensure the efficiency of the training part, and at the same time, avoid the unawakened neuron issue that sometimes occurs when using Rectified Linear Unit (ReLU) as the activation function. The mathematic expressions of the activation functions are listed below:

$$\mathrm{Tanh}(x) = \frac{e^x + e^x}{e^x + e^{-x}} \tag{1}$$

DOI: 10.54254/2755-2721/47/20241191

$$PReLu(y_i) = \begin{cases} y_i, & y_i > 0 \\ a_i y_i, & y_i \le 0 \end{cases} \tag{2}$$

The input of the activation function is x or y and the output is a non-linear mapping of the input. An activation function creates a non-linear relationship between input and output, making the network able to represent a non-linear relationship. Especially, a_i is a self-learning parameter that controls the gradient when y is less than 0 in PReLU.

2.3.2. Backward Propagation. When dealing with the parameter update in backward propagation, SmoothL1Loss is used as the loss function. The reason why the SmoothL1Loss function is adopted is this function has a smooth curve when the absolute difference is less and can have more tolerance toward the data points near the fitting curve than the frequently-used L1Loss function, which can decrease the possibility of undulation near the optimal point. The mathematic expression of the loss function is listed below:

$$SmoothL1 = \begin{cases} 0.5x^2, & |x| < 1 \\ |x| - 0.5, & |x| \ge 1 \end{cases} \tag{3}$$

$$x = f(x_i) - y_i \tag{4}$$

Here x is the difference between the prediction value and ground truth. SommthL1Loss ensures that the gradient is small enough when x is small and also not too large when x is big. So, the convergence speed is fast enough at the beginning meanwhile preventing vibration near the bottom.

The optimizer utilizes the loss function to update the parameters in the neural network [10]. In this paper, Adaptive Moment Estimation (Adam) is adopted as the optimization algorithm. A variable learning rate enables a broader parameter setting scope and better adaptability, which is of great significance to this system. Adaptability and adjustment are the very qualities that this network needs, which makes it possible to train multiple times and get all the drivers' lap time predictions in a single network. The mathematic expression of the optimizer is listed below:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{5}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{6}$$

$$\widehat{m}_t = \frac{m_t}{1-\beta_1^t} \tag{7}$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2^t} \tag{8}$$

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} \widehat{m}_t \tag{9}$$

wherein g_t is the gradient of parameter; β_1, β_2 are the attenuation coefficients of 2 indices' weighted average; $\widehat{m}_t$, $\hat{v}_t$ are the moving averages of gradients after deviation correction; θ_{t+1} is the parameter updated; η is learning rate; ϵ is a tiny constant to keep denominator greater than 0. Adam can adjust the learning rate itself so that it has excellent adaptability.

3. Result

3.1. Training Details

During the training procedure, the network will be fed multiple times. Only one driver's data on one circuit will be fed at a time. All the output will be gathered in the form of a data frame and finally output a Comma-Separated Values (SCV) file.

Here are the hyperparameters that settled in the practical training part. The training epoch is 500, the learning rate is 0.01, and the weight decay is 0.

DOI: 10.54254/2755-2721/47/20241191

3.2. Performance Comparison

In most circumstances, the ratio of loss value to scale of the data could be controlled below 5%, which means the deviation of the result is substantially below 5%.

Taking the lap times of Lewis Hamilton who is mentioned above as an example, the comparison between the predictions given by the neural network and the actual race results by July 2023 is shown in Table 1 below.

Although the ratio of relatively accurate predictions is more than half, some of the predictions have considerable deviations.

Table 1. The comparison between prediction and actual result.

Circuit Name	Prediction	Result	Deviation
Albert Park Grand Prix Circuit	82.23	77.14	6.6%
Bahrain International Circuit	89.21	90.38	1.3%
Circuit de Barcelona-Catalunya	78.83	72.82	8.3%
Circuit de Monaco	73.06	71.73	1.9%
Circuit Gilles Villeneuve	72.84	87.63	16.9%
Silverstone Circuit	86.51	77.21	12.1%
Hungaroring	77.01	76.61	0.5%
Circuit de Spa-Francorchamps	105.43	107.09	1.6%
Red Bull Ring	64.74	64.82	0.1%

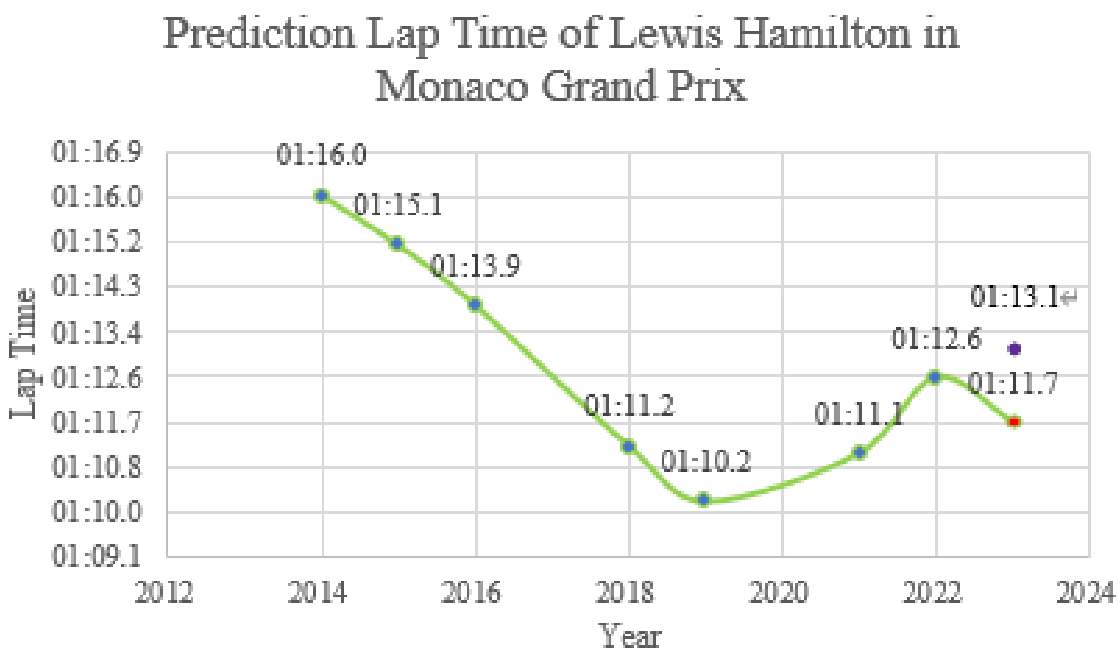

Figure 3. Predicted lap time of Lewis Hamilton in Monaco Grand Prix (Figure credit: Original).

As Figure 3 above shows, if the prediction is given by the trend of previous years' lap times, the result ought to be located at 1:13.1 (purple point), which is exactly what the network provides. But the fact is that the lap time decreases to 1:11.7 (orange point) due to the increase in the performance of race cars in 2023. The limitation of a SISO system is that a simple regression system can not consider abstract and non-digital elements. Generally speaking, the consequence of prediction is fairly reasonable and credible.

DOI: 10.54254/2755-2721/47/20241191

4. Conclusion

This paper highlights that the neural network system demonstrates competent performance and accuracy when limited to regression-based predictions. The regression model serves as a convenient and efficient method for predicting players' competition results. However, a limitation of this approach is that it cannot take into account realistic factors, such as weather conditions and driver status. The optimization of the model involves several aspects. For instance, considering the performance of other drivers at a specific track can provide additional information about how a particular driver might perform on that circuit. The tendencies of the teammate's performance can reflect the tendencies of the car's performance. Additionally, incorporating data from previous races and seasons can offer valuable insights into a driver's form and predict their likely performance in the future.

In the future, this type of prediction model could not only be used for predicting a player's competition result but could also be a part of a player performance assessment system. The predictions could be presented as expectations or targets and could be checked and analyzed after and during races. This system could become a crucial component of training and assessment processes in sports. It could also provide valuable insights to coaches, team managers, and sponsors, enabling them to make more informed decisions and improve overall performance.

References

[1] Lalwani, A., Saraiya, A., Singh, A., Jain, A., & Dash, T. (2022). Machine Learning in Sports: A Case Study on Using Explainable Models for Predicting Outcomes of Volleyball Matches. arXiv preprint arXiv:2206.09258.

[2] Wesselbaum, D., & Owen, P. D. (2021). The value of pole position in Formula 1 history. Australian Economic Review, 54(1), 164-173.

[3] Fooladpanjeh, S., Dadrasi, A., Gharahbagh, A. A., & Parvaneh, V. (2021). Fuzzy neural network and coupled gene expression programming/multivariate non-linear regression approach on mechanical features of hydroxyapatite/graphene oxide/epoxy: Empirical and optimization study. Journal of Mechanical Engineering Science, 235(23), 7169-7179.

[4] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.

[5] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., et al. (2018). Recent advances in convolutional neural networks. Pattern recognition, 77, 354-377.

[6] Koch, B., Sainburg, T., Geraldo, P., Jiang, S., Sun, Y., & Foster, J. G. (2021). Deep learning of potential outcomes. arXiv preprint arXiv:2110.04442.

[7] F1-The Official Home of Formula 1 Racing. URL: www.formula1.com. Last Accessed 2023/09/05

[8] van Kesteren, E. J., & Bergkamp, T. (2023). Bayesian analysis of Formula One race results: disentangling driver skill and constructor advantage. Journal of Quantitative Analysis in Sports, 0, 1-8.

[9] Winkler, D. A., & Le, T. C. (2017). Performance of deep and shallow neural networks, the universal approximation theorem, activity cliffs, and QSAR. Molecular informatics, 36(1-2), 1600118.

[10] Boob, D., & Lan, G. (2017). Theoretical properties of the global optimizer of two layer neural network. arXiv preprint arXiv:1710.11241.