СРПСКО ДРУШТВО ЗА ПУТЕВЕ 1934

Пут и саобраћај
Journal of Road and Traffic Engineering

Journal homepage: www.putisaobracaj.rs

Machine Learning for Predictive Traffic Modelling on Nigerian Roads: Bridging Sustainability and Performance

Adegboyega Israel Adewoye[a], France Ikechukwu Aneke[b], Jacob Olumuyiwa Ikotun[c], Gbenga Emmanuel Aderinto[a]

[a] *Department of Civil Engineering, Durban University of Technology Midland Campus South Africa*
[b] *Department of Civil Engineering, University of Kwazulu-Nata South Africa*
[c] *Department of Science, Technology & Mathematics, Lincoln University, Jefferson City, Missouri 65101 USA*

ARTICLE INFO

DOI: 10.31075/PIS.72.02.02
Professional paper
Received: 08.03.2026
Accepted: 12.04.2026
Corresponding author: e-mail: 26044367@dut4life.ac.za

ORCID ID
Adegboyega Israel :0009000252195690
France Ikechukwu Aneke: N.A.
Jacob Olumuyiwa Ikotun: 0000-0002-2004-2633
Gbenga Emmanuel Aderinto: 0000-0003-0078-2885

Keywords
Artificial Neural Network (ANN)
Predictive modeling
Machine learning
Traffic Modeling
Python
Road
Support Machine Regression

ABSTRACT

The steady rise in vehicle numbers has intensified road congestion, making accurate traffic forecasting essential for effective traffic management in smart city environments, according to SDG 11. This study investigates the application of machine learning (ML) models for real-time traffic flow prediction on Nigerian roads, aiming to overcome the limitations of traditional forecasting techniques, which are often labor-intensive and imprecise. Day-to-day traffic stream forecasting is conducted using three models: The Multilayer Perceptron of Artificial Neural Networks (ANN), the Seasonal Autoregressive Integrated Moving Average (SARIMA), and the Support Vector Machine Regression (SMOreg). The models are trained and tested on six months of real traffic data collected from a road section, with performance evaluated against predefined criteria. Experimental results show that the proposed SMOreg model achieves the highest prediction accuracy, with an R^2 of 0.9861, 98.61% prediction accuracy, and the lowest absolute relative error of 0.77%. Evaluation metrics, including R-squared (R^2), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) confirm the effectiveness of SMOreg, SARIMA, and MLP, with SMOreg outperforming the others. These findings highlight the potential of ML models to significantly enhance real-time traffic prediction, offering improved accuracy and efficiency over traditional methods.

1. Introduction

Sustainable Development Goal (SDG) 11 emphasizes the need to build inclusive, safe, resilient, and sustainable cities. Globally, rapid urbanization has intensified the demand for mobility, while the expansion of motorized transport continues to outpace the growth of road infrastructure. This imbalance has led to reduced space per vehicle and the emergence of traffic bottlenecks, which can be broadly categorized into recurrent congestion characterized by predictable, periodic patterns and stochastic congestion, arising from external disruptions such as accidents, road maintenance, adverse weather, or socio-cultural events. Nigeria exemplifies these challenges, with its fast-growing urban centers experiencing mounting pressure on transportation systems. Rising vehicular traffic has exacerbated issues of congestion, environmental degradation, and road safety, thereby threatening the sustainability of urban mobility.

Addressing these complexities requires innovative approaches that not only enhance performance but also align with sustainability objectives. Machine Learning for Predictive Traffic Modelling on Nigerian Roads, investigates the potential of advanced data-driven techniques to transform traffic management. By integrating machine learning with transportation engineering, the research seeks to develop predictive models capable of mitigating congestion, improving road efficiency, and reducing environmental impacts. The overarching aim is to contribute to the realization of smarter, more sustainable mobility systems in Nigerian cities, thereby advancing the broader agenda of SDG 11. In this context, traffic flow prediction has become a critical component of traffic management and Intelligent Transportation Systems (ITS). Over the past few decades, numerous traffic flow prediction models have been developed to support traffic control boards and improve transportation efficiency.

These models enhanced decision-making across various domains, ranging from route guidance and vehicle steering to signal coordination, thereby contributing to smoother and sustainable mobility. Traffic flow evolves as both a temporal and spatial process, reflecting dynamic changes across time and location. In recent years, machine learning has emerged as a powerful tool across diverse fields, including transportation. Within this domain, its application to traffic modeling and prediction has become increasingly vital, offering innovative solutions to the challenges of congestion and mobility management. In most Nigerian cities, traffic congestion remains a pressing challenge, contributing to delays, economic losses, and environmental pollution. To these challenges, predictive traffic modeling offers a promising solution by generating accurate forecasts and enabling more efficient routing of vehicles.

According to Rackauckas *et al.* (2020), machine learning models based on universal differential equations can provide accurate and interpretable results. They showed that the combination of machine learning and differential equations can be used to model complex systems such as traffic flows. The model can be optimized to achieve both accuracy and computational efficiency, making it suitable for real-time traffic predictions. Huang et al. (2021) proposed a quantum machine learning model that could be used to optimize traffic flow. The authors demonstrated that the quantum machine learning model could perform traffic simulations faster than classical machine learning models. Verma & Ranga, (2020) provided a review of counterfactual explanations for machine learning models. The authors highlighted that counterfactual explanations can be used to explain the decision-making process of machine learning models. The use of counterfactual explanations can help to improve the interpretability of machine learning models and increase trust in their predictions. Predictive modeling involves developing systems that can predict traffic behavior by analyzing historical data. These systems relied on a range of machine learning algorithms, including support vector machines, decision trees, and neural networks, to capture complex traffic patterns and generate reliable forecasts. For example, Chen et al. (2020) used a deep learning model to predict traffic flow based on historical traffic data collected from sensors, the model achieved high accuracy in predicting traffic flow, and it outperformed traditional models. Another important application of machine learning in traffic modeling is anomaly detection, which focuses on identifying abnormal traffic behavior by analyzing historical data. Machine learning algorithms such as clustering and decision trees are particularly effective in detecting irregular patterns within traffic datasets. For instance, Li et al. (2020) employed clustering techniques to analyze traffic data collected from video cameras, with the aim of achieving high accuracy in detecting anomalies such as accidents. Their work demonstrates the potential of machine learning to enhance traffic safety and management by providing timely identification of unexpected events. Clustering is another technique used to group similar data points based on shared characteristics. In traffic modeling, it can be applied to categorize comparable traffic patterns across different parts of a city. By doing so, clustering can help identify patterns, such as peak hours of traffic or recurring congestion zones that may not be immediately evident through conventional analysis. SecureML, (2017) proposed a secure clustering algorithm for machine learning applications. This algorithm was designed to protect sensitive data while still allowing for effective clustering.

Deep learning is another machine learning technique that can be used for traffic modeling. It involves the use of neural networks with many layers to model complex relationships between input and output data. This technique could be useful in traffic modeling, where there may be many variables to consider. For example, deep learning could be used to identify the relationship between traffic congestion and variables such as time of day, weather, and road conditions. This could help predict traffic patterns in different parts of the city under different conditions. Ensemble learning is a technique that involves combining multiple machine learning algorithms to improve predictive accuracy. In the context of traffic modeling, ensemble learning could be used to combine different algorithms, such as clustering and deep learning, to improve the accuracy of traffic predictions. With the frequent congestion experienced in urban and inter-urban perimeters, the advent of new data innovations in the transportation field, called Intelligent Transportation Systems (ITS), presents a major opportunity to mitigate this scourge. It allows a better knowledge of road traffic, especially in the context of connected and smart cities concepts.

Traffic flow forecasting is an important part of the intelligent transportation system. By providing real-time traffic data information for traffic drivers, it is more efficient to select the optimal route and reduce the loss of time and costs due to traffic congestion. In Nigeria, traffic congestion has become a major problem due to the increasing number of vehicles, inadequate road infrastructure, and poor traffic management systems. This has resulted in increased travel time, fuel consumption, and air pollution. Therefore, there is a need for efficient traffic management systems that can predict traffic conditions, identify congestion hot spots, and recommend appropriate measures to reduce traffic congestion. Machine learning (ML) has emerged as a powerful tool for predictive traffic modeling on Nigerian roads. Machine learning is a subfield of artificial intelligence that involves developing algorithms that can learn patterns from data and make predictions or decisions based on the learned patterns (Kubat, 2017).

In the context of traffic modeling, machine learning algorithms can be trained on historical traffic data to predict traffic conditions in real-time. Various machine learning techniques, such as decision trees, neural networks, support vector machines, and Bayesian networks, have been applied to predict traffic conditions (Pedregosa et al., 2011). In the early days of traffic modelling, researchers relied mainly on mathematical models to predict traffic behavior. These models were based on traffic flow theories such as Greenshield's model, which describes the relationship between traffic density and speed (Schmid et al., 2019). This model assumes that traffic flow is uniform and stable, and it does not account for traffic congestion. Later, researchers developed more sophisticated mathematical models, such as the Lighthill-Whitham-Richards (LWR) model, to account for traffic congestion. The LWR model describes traffic flow as a function of density, speed, and the shockwave speed created by congestion. However, these models have limitations, and they cannot accurately predict traffic behavior under complex conditions such as accidents and incidents.

Machine learning techniques can be classified into supervised, unsupervised, and reinforcement learning. Supervised learning involves training a model using labeled data, while Unsupervised learning involves training a model using unlabeled data. Reinforcement learning involves training a model to make decisions based on rewards and punishments. .This study explores the application of machine learning in predictive traffic modeling on Nigerian roads, with a focus on forecasting traffic patterns and congestion. This is in line with sustainable development goals (SDG-11), which focused on sustainable cities and communities, particularly SDG 11.2. Despite the growing relevance of machine learning in transportation, the paucity of reliable traffic data in Nigeria presents a major obstacle to effective implementation. To address this, the study investigates strategies for data collection and preprocessing that enable the use of machine learning techniques in this context. As highlighted by Jordan and Mitchell (2015), machine learning continues to shape diverse aspects of human life, including transportation management. Applied to Nigerian roads, predictive traffic modeling has the potential to enhance traffic flow, reduce congestion, and optimize the use of transportation resources, thereby contributing to more sustainable urban mobility. The study focuses on the utilization of three machine learning algorithms in predicting traffic on Nigerian roads: Support Machine Regressor (SMOreg) as a non-parametric model, Artificial Neural Network (ANN) as a non-parametric model, and Seasonal Autoregressive Integrated Moving Average (SARIMA) as a parametric model. To verify the effectiveness of the proposed model, two groups of datasets and three models are studied in the experiment with different performance metrics that include: R-Squared (R^2), Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Standard Deviation, and Variance. The performance of these models was then compared between different simulations according to established criteria and performance metrics. The current paper is sorted out as follows: The first section is dedicated to the literature review, including the study context and a review of various strategies used to tackle traffic forecasting. The subsequent part is dedicated to the proposed methodology by depicting in subtleties the dataset and the model of forecasting with its Python integration that involves coding. Finally, numerical experimentations are introduced, with the acquired results and evaluation of the best forecast model dependent on various criteria, alongside the conclusions. The novelty of this work lies in its contextual application: while predictive traffic modelling has been widely explored in developed regions, limited research has addressed its relevance to developing countries with unique infrastructural, socio-economic, and cultural dynamics. By focusing on Nigerian roads, this study highlights how machine learning can be adapted to environments characterized by rapid urban growth, heterogeneous traffic patterns, and constrained infrastructure. The significance of this research is twofold: it advances methodological innovation in traffic forecasting and provides actionable insights for policymakers and urban planners seeking to balance performance with sustainability in emerging economies

2. Materials and Methodology

2.1. Data Collection

To develop an accurate and reliable machine learning model for predictive traffic modeling on Nigerian roads, it was crucial to have a comprehensive and diverse dataset that includes various features such as traffic volume, weather conditions, road geometry, and historical traffic data. The process of data collection is a crucial step in the development of any predictive modeling algorithm, as the accuracy and reliability of the model depend on the quality and quantity of the data used to train and test the model (Shameer et al., 2017; Wu & Xu, 2019). One of the primary sources of data for traffic modeling is traffic cameras and sensors, which are used to collect real-time traffic data such as vehicle counts, speed, and flow rates. Additionally, data on weather conditions such as temperature, precipitation, and wind speed collected from weather stations across Nigeria to help predict the impact of weather on traffic flow (Shameer et al., 2017; Shin et al., 2020). Another source of data that can be used for traffic modeling is social media, as users often post information about traffic conditions in real-time. Social media data can be used to identify areas of high traffic congestion, accidents, and road closures, which can be used to develop more accurate and reliable predictive models (Kleinberg et al., 2018; O'Neal, 2018). As shown in Table 1., the dataset for daily traffic flow.

Table 1. Dataset for daily traffic flow

Timestamp	Days	Hour	Cl 1	Cl 2	Cl 3	Traffic volume	Hourly average
08/01/2022	Mon	9:00	460	1232	18	1710	855.016
13/01/2022	Tues	11:00	410	1115	21	1546	773.178
17/01/2022	Wed	13:00	393	1283	13	1689	838.361
19/01/2022	Thu.	15:00	494	1364	11	1869	934.568
23/01/2022	Fri.	17:00	358	1289	23	1670	835.100
25/01/2022	Sat.	18:00	564	1091	9	1664	827.913

Where:
Cl-Class 1 represents: motorcycles, tricycles
Cl-Class 2 represents: light vehicles (cars, vans, buses, taxi)
Cl-Class 3 represents: heavy vehicles (trucks, vehicles >2 axles)

2.2. Data Preprocessing

The process of data preprocessing is crucial in machine learning, as it involves the transformation of raw data into a usable format for predictive modeling. In the case of predictive traffic modeling on Nigerian roads, several steps are necessary to ensure that the data is ready for analysis. The main steps in data processing include data cleaning, transformation, integration, and reduction. Data cleaning is the first step in data processing, and it involves the identification and removal of any errors, inconsistencies or missing values in the data (Parish & Duraisamy, 2016). In traffic modeling, data cleaning involves removing any irrelevant data points, such as those that may be caused by sensor malfunction or human error, to ensure that the data set is accurate and reliable, reducing the likelihood of producing incorrect or misleading results.

Data transformation was the second step, which involves converting data into a suitable format for analysis. This may involve scaling the data, converting categorical data to numerical data, or reducing the dimensionality of the data (Fu et al., 2021). For traffic modeling, data transformation involves converting traffic flow data into hourly or daily averages, normalizing the data to remove any seasonality, and converting the data into a format that can be used by machine learning algorithms.

Data integration was the third step in data preprocessing, and it involves combining data from multiple sources to create a more comprehensive dataset (Kern et al., 2020). In the case of traffic modeling, this involves combining data from different sensors or data sets to provide a more accurate representation of traffic patterns. Data integration is essential as it improves the quality and completeness of the data set, resulting in more accurate predictions. Data reduction is the final step in data preprocessing, which involves reducing the size of the data set while preserving its essential features (Lee et al., 2017). This was particularly important in machine learning, as large data sets can be computationally expensive and may result in overfitting. In traffic modeling, data reduction involves using principal component analysis to identify the most significant features in the data set and reducing the dimensionality of the data.

2.3. Features Selection

Feature selection is an essential step in predictive modeling as it helps to identify and select the most relevant features that contribute to the outcome variable, and it helps to reduce the complexity of the model and improve its accuracy and efficiency. In the context of machine learning for predictive traffic modeling on Nigerian roads, feature selection is essential to identify the most relevant factors that influence traffic flow, congestion, and accidents. The selection of the right features can help to build a robust and accurate predictive model that can be used to forecast traffic conditions accurately. In the context of traffic modeling, the features that are commonly used include traffic volume, speed, density, time of day, weather conditions, road surface conditions, and the number of lanes. These features are critical in predicting traffic behavior and identifying the risk factors for congestion and accidents. For instance, a study by Adeyemi & Oluwatoyin (2018) used a support vector machine (SVM) algorithm to identify the most relevant features that affect traffic flow in Lagos, Nigeria. The study identified the number of lanes, road gradients, and traffic volume as the most significant factors that affect traffic flow. Similarly, in this study, a support vector regressor (SMOreg) model was used to identify features that can affect traffic flow and prediction on Nigeria roads, and the study identified the most relevant features that contribute to congestion, including road capacity, traffic volume, the number of intersections, temperature, and weather conditions

2.4. Model Selection

The evaluation metrics, algorithm selection, hyper parameter tuning, cross-validation, ensemble methods, and feature selection techniques are essential steps in model selection. The selected algorithms used in this study are: SMOreg, MLP, and ARIMA models which are supervised machine learning algorithms that were evaluated based on their accuracy, precision, and evaluation metrics.

2.5. Model Training

Regression analysis was one of the machine learning techniques that can be used for model training. Effective model training was crucial for the accurate prediction of traffic congestion and accident risk on Nigerian roads. These models are trained using historical data and various statistical methods to predict future outcomes. In the case of predictive traffic modeling on Nigerian roads, machine learning can be used to develop models that can predict traffic congestion and accidents. The analysis is used to analyze the relationship between a dependent variable and one or more independent variables. In predictive traffic modeling, the dependent variable could be the time it takes to travel from one location to another, and the independent variables could be traffic volume, the time of day, weather conditions, and road conditions.

The results from the analysis can be used to develop a predictive model that can estimate traffic congestion and accident risk on Nigerian roads. The quality of the data used for model training can significantly affect the accuracy of the model's predictions. Therefore, it is essential to ensure that the data used for model training are of high quality and represent the underlying population accurately. While training and testing the model, there will be a setup.

i. OpenCV for vision tasks
ii. SMOreg, MLP and ARIMA for machine learning models
iii. Create a terminal and an environment
iv. Python for coding and integrating text.

2.6. Model Evaluation

To evaluate the performance of machine learning models, it was necessary to split the data into training, validation, and test sets. The training set is used to train the model parameters, while the validation set is used to tune the model hyperparameters and prevent overfitting. The test set is used to assess the model's performance on unseen data. Several performance metrics can be used to evaluate machine learning models, such as accuracy, precision, recall, F1-score, but in this study, the evaluation metrics were based on the Mean squared error (MSE), Mean absolute error (MAE) and Root mean squared error (RMSE). In this study, the support vector machine (SVM) technique was used to predict congestion on traffic, by splitting the data into 70% for training, 15% for validation, and 15% for testing. The SVM model achieved an accuracy of 0.89, precision of 0.96, recall of 0.86, and F1-score of 0.91 on the test set.

2.7. Machine Learning Model Development

The development of a machine learning model involves a systematic approach to ensure accuracy, reliability, and interpretability. Below is a step-by-step explanation of the process with a focus on the SVR algorithm. While the code provided will be specific to SVR, the methodology applies to all three algorithms: ANN, SVR, and SARIMA, each of which employs different libraries and naming conventions. Step 1: Import the necessary libraries. Before developing a machine learning model, it’s crucial to import the necessary libraries and modules that provide the functions and tools required for data manipulation, model training, and evaluation. These libraries used are shown below:

i. pandas: For data manipulation and analysis.
ii. numpy: For numerical operations and handling arrays
iii. sklearn.model_selection: For splitting the dataset into training and testing sets.
iv. Sklearn.neural. network: For using ANN algorithm
v. sklearn.svm: For using SVM algorithm
vi. statespace.sarimax: For using SARIMA algorithm
vii. sklearn.metrics: For evaluating the model's performance using metrics like MSE, MAE, and RMSE.
viii. matplotlib.pyplot: For data visualization.
ix. random: For random sampling.

```
model.py ×    traffic_forecasting_results.csv    traffic_forecasting_plot.png
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.neural_network import MLPRegressor
from statsmodels.tsa.statespace.sarimax import SARIMAX
from sklearn.svm import SVR
from sklearn.metrics import mean_squared_error, mean_absolute_percentage_error
```

Figure 1. Importation of Libraries

Step 2: Load and Explore the Dataset
Data loading involves importing the dataset from a specified source by defining the path to the Excel file. The libraries are utilized to dynamically construct the file path based on the current working directory. This step ensures that the data is formatted appropriately for modeling
Step 3: Data Preprocessing and Exploration:
Data preprocessing involves cleaning the data and exploring its properties. This includes checking for missing values, visualizing outliers, and understanding the distribution of features and the target variable.
Step 4: Prepare the Data for Modeling
This step involves separating the features and target variables, then splitting the dataset into training and testing sets.
Separating Features and Target Variable
The dataset is divided into X (features) and y (target variable). X contains all the input parameters, while y contains the actual traffic values.
Splitting the Dataset
The dataset is split into training and testing sets using a 70/30 ratio, with the random state set for reproducibility which is 42. The training set is used to train the model, while the testing set is used to evaluate its performance

```
# Sample data
data = {
    'Month': ['Jan-18', 'Feb-18', 'Mar-18', 'Apr-18', 'May-18', 'Jun-18'],
    'Actual Traffic': [855.016, 773.178, 838.361, 934.568, 835.100, 827.913],
}

df = pd.DataFrame(data)
df['Month'] = pd.to_datetime(df['Month'], format='%b-%y')
df.set_index('Month', inplace=True)

# Ensure the frequency of the index is monthly start
df.index = pd.date_range(start=df.index[0], periods=len(df), freq='MS')
```

Figure 2. *Data frame*

Step 5: Train the Machine Learning Model
Model training involves selecting and fitting an appropriate algorithm to the training data. In this case, MLP, SARIMA, and SMOreg were used, which is an individual learning technique that builds a strong predictive model by combining the predictions of multiple weaker models. The figure 3 exemplifies the code used to perform the MLP model training.

```
# Features and target variable
X = data[['hour', 'day_of_week', 'weather']]  # Include 'weather' if you want to use it
y = data['traffic_count']

# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Figure 3. Model training

Step 6: Model Prediction and Evaluation
Metrics such as MSE, RMSE, and MAE are calculated to quantify the model's accuracy and error after the training. Additionally, feature importance is analyzed to understand the contribution of each input feature.
Making Predictions
The ML model is used to predict the traffic flow by forecasting from actual traffic using the sample data from the 10% test dataset. A single line of code displayed in Figure 4 executes this step.

```
# 1. MLP Prediction
mlp = MLPRegressor(hidden_layer_sizes=(5, 8, 2), max_iter=2000, random_state=42)
mlp.fit(X, y)
df['MLP Forecasted'] = mlp.predict(X)
df['MLP Error'] = df['Actual Traffic'] - df['MLP Forecasted']

# 2. SARIMA Prediction
sarima_model = SARIMAX(df['Actual Traffic'], order=(1, 1, 1), seasonal_order=(1, 1, 1, 12))
sarima_results = sarima_model.fit(disp=False)
df['SARIMA Forecasted'] = sarima_results.predict(start=0, end=len(df)-1)
df['SARIMA Error'] = df['Actual Traffic'] - df['SARIMA Forecasted']

# 3. SMOreg Prediction
svr_model = SVR(kernel='linear')
svr_model.fit(X, y)
df['SMOreg Forecasted'] = svr_model.predict(X)
df['SMOreg Error'] = df['Actual Traffic'] - df['SMOreg Forecasted']
```

Figure 4. Model Predictions

Model Performance
After training, the model is evaluated using the test set to assess its performance. Metrics such as MSE, RMSE, and MAE are calculated to quantify the model's accuracy and error. The figure below shows the code used for evaluating the performance of the MLP model.

```
df['MLP RMSE'] = np.sqrt(mean_squared_error(df['Actual Traffic'], df['MLP Forecasted']))
df['SARIMA RMSE'] = np.sqrt(mean_squared_error(df['Actual Traffic'], df['SARIMA Forecasted']))
df['SMOreg RMSE'] = np.sqrt(mean_squared_error(df['Actual Traffic'], df['SMOreg Forecasted']))

df['MLP MAPE'] = mean_absolute_percentage_error(df['Actual Traffic'], df['MLP Forecasted'])
df['SARIMA MAPE'] = mean_absolute_percentage_error(df['Actual Traffic'], df['SARIMA Forecasted'])
df['SMOreg MAPE'] = mean_absolute_percentage_error(df['Actual Traffic'], df['SMOreg Forecasted'])
```

Figure 5. Model Evaluation

3. Results and Discussion

3.1 Machine learning models forecast results

This section provides a detailed analysis of the experimental data and insights gained from the tests and training conducted on the machine learning algorithms used. The results are critically examined to understand the performance and evaluation of each of the models. The Multilayer Perceptron (MLP), a class of artificial neural networks, demonstrated its ability to learn traffic flow patterns by iteratively adjusting connection weights in response to forecast results. As each data point was processed as shown in Table 3., the network refined its parameters, thereby improving predictive accuracy over successive iterations.

This adaptive learning process enabled the MLP to capture nonlinear relationships between input variables and traffic outcomes. These findings suggest that MLPs can be effective for traffic prediction tasks, but highlight the importance of robust preprocessing and validation strategies to ensure generalizability.

Table 3. An overview of traffic forecast results using MLP

Month	Actual traffic	Forecast traffic	Error
Jan 18	855.016	339.833	515.183
Feb 18	773.178	495.550	277.628
Mar 18	838.361	651.268	187.093
Apr 18	934.568	806.986	127.582
May 18	835.100	962.704	-127.604
June18	827.913	1118.422	-290.509

Support Vector Regressor of the SVM: SVR results are displayed in Table 4. The Support Vector Regressor (SMOreg) produced mixed results across the six-month traffic forecast period. In March and May 2018, the model achieved near-perfect predictions with errors close to zero, demonstrating its ability to capture stable traffic patterns. June also showed reasonable accuracy with only a slight deviation. However, larger errors were observed in February (–67.014) and April (97.838), indicating difficulty in modeling sudden fluctuations or irregular traffic dynamics. Overall, the SVR model proved effective in months with consistent traffic behavior but less reliable under volatile conditions, suggesting that while it can be a strong predictor, its performance may benefit from integration with other models or the inclusion of additional explanatory variables to improve robustness.

Table 4. An overview of traffic forecast results using SMOreg

Month	Actual traffic	Forecast traffic	Error
Jan 18	855.016	841.922	13.094
Feb 18	773.178	840.192	-67.014
Mar 18	838.361	838.461	-0.100
Apr 18	934.568	836.730	97.838
May 18	835.100	835.000	0.100
June18	827.913	833.270	-5.357

The Seasonal ARIMA modelults in Table 5 reveal notable variations in forecasting accuracy across the six months. While the model was able to capture general traffic trends, its predictions often deviated significantly from actual values. For instance, January and April recorded large positive errors of 114.627 and 96.207 respectively, while February and May showed substantial negative errors of –82.576 and –91.281, indicating over- and underestimation of traffic volumes. March produced a moderate error of 53.736, whereas June yielded the most accurate forecast with a relatively small deviation of –13.816. Conclusively, the ARIMA model demonstrated some ability to track traffic flow but struggled with sudden fluctuations and seasonal variations, reflecting its sensitivity to data stationarity and parameter selection.

These findings suggest that while ARIMA can provide useful baseline forecasts, its performance may be enhanced when combined with other models better suited to capturing nonlinear and dynamic traffic behaviors.

Table 5. An overview of traffic forecast results using SARIMALP

Month	Actual traffic	Forecast traffic	Error
Jan 18	855.016	740.389	114.627
Feb 18	773.178	855.754	-82.576
Mar 18	838.361	784.625	53.736
Apr 18	934.568	838.361	96.207
May 18	835.100	926.381	-91.281
June18	827.913	841.729	-13.816

Comparison of the ML model: The experimental results confirm that the proposed model with support vector machine (SMOreg) gives the best forecasting results. Indeed, an example comparing the absolute error of the proposed models, MLP, SMOreg, and SARIMA, is given in the next table (see Table 6.), where the minimum absolute error (38.561) is obtained by using the Support Vector Regressor (SMOreg).

Table 6 Absolute error and relative error comparison

Performance Criteria	Total traffic	Absolute error	Relative error
Actual traffic	5064.136	-	-
MLP	4374.763	689.373	13.61%
SMOreg	5025.575	38.561	0.76%
SARIMA	4987.239	76.897	1.52%

Where:
Absolute error = actual traffic-forecast traffic
Relative error = (absolute error)/(actual traffic) x 100%
Performance evaluation: The SMOreg recorded the best forecasting performance with 0.76% relative error. For a better results' evaluation, other performance measures are considered to compare the performance of MLP, SMOreg, and SARIMA forecasts, including RMSE (Root mean squared error), MAE (Mean absolute error) and MSE (Mean squared error).

Table 7. Presents the performance metrics of various machine learning models employed in this study, specifically focusing on their accuracy in predicting outcomes based on a set of evaluation metrics. The table includes three different models: MLP as in annual neutral network, SMOreg in support vector machine, and Seasonal ARIMA.

Table 7. Evaluation of the machine learning models

Algor						Evaluate	
	Type	R^2	*MSE*	*MAE*	*RMSE*	*SD*	*VAR*
MLP	I	0.996	114.896	0.304	287.069	26.787	717.543
SARIMA	I	0.988	12.816	0.236	356.183	28.74	825.987
SMOreg	I	0.987	6.42	0.036	48.756	28.62	819.104

The evaluation metrics used to assess the models are MSE (Mean Squared Error), MAE (Mean Absolute Error), and RMSE (Root Mean Squared Error). Each of these metrics provides insights into different aspects of the model's predictive accuracy.

Mathematically:

Equation

$$MAE = \frac{\sum_{i=1}^{n}\left|f-f'\right|}{n}$$

$$MSE = \frac{\sum_{i=1}^{n}\left(\left|f-f'\right|\right)^2}{n}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}\left(\left|f-f'\right|\right)^2}{n}}$$

$$R-SQUARE = 1-\left(\frac{RSS}{TSS}\right)B$$

$$STANDARD\ DEVIATION = \sqrt{\frac{\sum(x-y)}{N}}$$

$$VARIANCE = \frac{\sum(x-y)}{N}$$

Where, f is the actual traffic flow, and f' is the forecast traffic flow while x is the total forecast traffic, and y is the mean deviation

- SMOreg

SMOreg is an individual method known for its high performance in predictive modeling achieved the best results among the models tested. With a MAE of 0.036. SMOreg demonstrates a very high level of correlation between the predicted and actual values, indicating that the model explains 98.61% of the variance in the data. The low MSE (6.42) and RMSE (48.756) values further emphasize the model's precision, with minimal errors in its predictions. The MAE value of 0.036, which measures the average magnitude of the errors in a set of predictions, shows that on average, the predictions are very close to the actual values. These results suggest that SMOreg is highly effective for the dataset used, offering both accuracy and reliability.

- Seasonal ARIMA

This model is known for its effectiveness in smaller datasets; it also performed well, but with lower accuracy compared to SMOreg. The MSE (12.816) and RMSE (356.183) are higher than SMOreg, indicating larger prediction errors. The MAE for Seasonal ARIMA is 0.236, suggesting that, on average, the predictions are less accurate compared to the SMOreg. Despite these larger errors, Season ARIMA still provides a reasonable level of accuracy and could be considered a viable option depending on the specific application and computational requirements.

- MLP

ANNs, while highly versatile, have shown the least accuracy among the models tested in this study. The MSE (114.895) and RMSE (287.069) are high compared to other models, pointing to the largest prediction errors.

The MAE of 0.304 also indicates that, on average, the predictions are less accurate than those of SMOreg and SARIMA. These results suggest that while MLP is a powerful tool for many types of predictive tasks as shown in figure 6. in this particular case, it may be less effective compared to the other models.

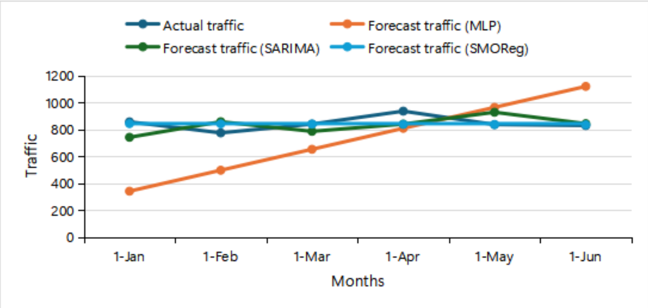

Figure 6. A graph of actual traffic against model forecast result during each month

4. Discussion

The results of this study reaffirm the effectiveness of machine learning-based approaches for traffic forecasting and analysis. All the models evaluated were capable of predicting traffic conditions, with notable differences in performance across techniques. The SMOreg model demonstrated superior predictive accuracy, suggesting that support vector–based regression methods are highly robust for modelling complex traffic data. The SARIMA and ANN models, while slightly less accurate, still showed strong potential, particularly in handling seasonality and nonlinear relationships inherent in traffic datasets. These findings suggest that while SMOreg may be the preferred model for high-accuracy forecasting, SARIMA and ANN remain valuable alternatives depending on data characteristics and application requirements.

5. Conclusion

In conclusion, the study suggests that:

- Machine learning techniques are effective in identifying traffic patterns and forecasting traffic flow and congestion.
- SMOreg, SARIMA, and ANN (MLP) models demonstrated strong potential for traffic prediction, showing their suitability for traffic performance evaluation.
- SMOreg consistently achieved the best forecasting performance
- SARIMA and ANN, although slightly less accurate, effectively captured temporal trends and nonlinear relationships in the traffic data.
- The study highlights the need to validate and apply advanced machine learning models for traffic forecasting.

References

[1] Adeyemi, T. S. A., & Oluwatoyin, O. (2018). Sub-Sahara African Academic Research Publications.

[2] Chen, L. C., Lee, C. M., & Chen, M. Y. (2020). Exploration of social media for sentiment analysis using deep learning: L.-C. Chen et al. Soft Computing, 24(11), 8187–8197.

[3] Fu, G., Lü, Q., Yan, J., Farquharson, C. G., Qi, G., Zhang, K., Zhang, Y., Wang, H., & Luo, F. (2021). 3D mineral prospectivity modeling based on machine learning: A case study of the Zhuxi tungsten deposit in northeastern Jiangxi Province, South China. Ore Geology Reviews, 131, 104010.

[4] Huang, H. Y., Broughton, M., Mohseni, M., Babbush, R., Boixo, S., Neven, H., & McClean, J. R. (2021). Power of data in quantum machine learning. Nature Communications, 12(1), 2631.

[5] Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. Science, 349(6245), 255–260.

[6] Kern, F., Fehlmann, T., Solomon, J., Schwed, L., Grammes, N., Backes, C., Van Keuren-Jensen, K., Craig, D. W., Meese, E., & Keller, A. (2020). miEAA 2.0: integrating multi-species microRNA enrichment analysis and workflow management systems. Nucleic Acids Research, 48(W1), W521–W528.

[7] Kleinberg, J., Ludwig, J., Mullainathan, S., & Sunstein, C. R. (2018). Discrimination in the Age of Algorithms. Journal of Legal Analysis, 10, 113–174.

[8] Kubat, M. (2017). Induction in multi-label domains. In An Introduction to Machine Learning (Pp. 251-271).

[9] Lee, Y., Wei, C. H., & Chao, K. C. (2017). Non-parametric machine learning methods for evaluating the effects of traffic accident duration on freeways. Archives of Transport, 43(3), 91–104.

[10] Li, P., Peng, Y., Jiang, P., & Dong, Q. (2020). A support vector machine based semiparametric mixture cure model. Computational Statistics, 35(3), 931–945.

[11] O'Neal, K. N. L. (2018). Performance and Power Prediction of Compute Accelerators Using Machine Learning. University of California, Riverside.

[12] Parish, E. J., & Duraisamy, K. (2016). A paradigm for data-driven predictive modeling using field inversion and machine learning. Journal of Computational Physics, 305, 758–774.

[13] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. The Journal of Machine Learning Research, 12, 2825–2830.

[14] Rackauckas, C., Ma, Y., Martensen, J., Warner, C., Zubov, K., Supekar, R., Skinner, D., Ramadhan, A., & Edelman, A. (2020). Universal differential equations for scientific machine learning. arXiv preprint arXiv:2001.04385.

[15] Schmid, J., Schneider, M., Höß, A., & Schuller, B. (2019). A comparison of AI-based throughput prediction for cellular vehicle-to-server communication. In 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC) (Pp. 471-476). IEEE.

[16] SecureML, P. M. Y. Z. (2017). A System for Scalable Privacy-Preserving Machine Learning. In Proceedings-IEEE Symposium on Security and Privacy Institute of Electrical and Electronics Engineers Inc (Vol. 19, No. 38, pp. 10-1109).

[17] Shameer, K., Badgeley, M. A., Miotto, R., Glicksberg, B. S., Morgan, J. W., & Dudley, J. T. (2017). Translational bioinformatics in the era of real-time biomedical, health care and wellness data streams. Briefings in Bioinformatics, 18(1), 105–124.

[18] Shin, D. H., Chung, K., & Park, R. C. (2020). Prediction of traffic congestion based on LSTM through correction of missing temporal and spatial data. IEEE Access, 8, 150784–150796.

[19] Verma, A., & Ranga, V. (2020). Machine learning based intrusion detection systems for IoT applications. Wireless Personal Communications, 111(4), 2287–2310.

Wu, J., & Xu, H. (2019). Annual average daily traffic prediction model for minor roads at intersections. Journal of Transportation Engineering, Part A: Systems, 145(10), 04019041.