Samarra Journal of Pure and Applied Science

www.sjpas.com

p ISSN: 2663-7405
e ISSN: 2789-6838

Prediction of Traffic Flow in Vehicular Ad-hoc Networks using Optimized Based-Neural Network

Ahmed Ibrahim Turki[1*]
1- Department of Physics, College of Education, University of Samarra, Iraq

https://doi.org/10.54153/sjpas.2024.v6i2.758

Article Information

Received: 09/11/2023
Revised:22/11/2023
Accepted: 11/12/2023
Published: 30/06/2024

Keywords:

Traffic flow, VANET, Developed Neural Network, Optimization, Prediction.

Corresponding Author

E-mail: ahmed.ibrahim@uosamarra.edu.iq
Mobile: 009647704290098

Abstract

Traffic flow prediction is crucial for managing traffic, reducing pollution, ensuring public safety, and is a vital element of Intelligent Transportation Systems. Furthermore, the variables that influence traffic patterns (such as public events, road closures, and accidents) are predominantly unforeseen, rendering the prediction of traffic flow an intricate undertaking. The objective of this research is to enhance the accuracy of traffic flow forecast by employing a novel strategy that utilizes feedforward Neural-Networks and the Quasi-Newton method for optimization. The proposed strategy decreases the error factor based on the Lagrange multiplier and Jacobian vector. This enhancement has resulted in expedited convergence throughout the process of learning. The sample was chosen using datasets provided by the Traffic Monitoring System, specifically from Highway England, Performance Measurement System, and Maryland 511, for the year 2023. In order to assess the proposed model, the research outcomes are juxtaposed with other established prediction methodologies. The study revealed that the research model that was established performed better than previous prediction strategies based on measures of mean absolute error and root mean squared error..

Introduction:

Transportation is a fundamental component of supply chain management [1]. Accurate forecasting of traffic patterns is a crucial aspect of modern transportation systems. It provides a significant advantage for many devices that rely on accurate future traffic statistics [2]. For example, the anticipated traffic patterns serve as a crucial point of consideration when determining vehicle routes, aiding travelers in making more informed decisions on their chosen paths. Anticipating the timing and location of congestion is highly beneficial for transportation management, since it enables experts to allocate resources to roadways during periods of congestion risk, ultimately reducing traffic congestion. As a result of its significant superiority over different devices, traffic flow prediction [3] has emerged as a prominent area of study in recent years. In essence, predicting traffic flow involves making assessments about future conditions based on insights and expertise derived from related historical data. Consequently, the mechanisms used for transmitting, collecting, mining, and storing data have a significant influence on prediction algorithms [4]. Various conventional traffic forecasting methods have been developed to incorporate temporal dependence. For instance, the Kalman filtering model employs a recursive algorithm to estimate and update traffic parameters over time. Gaussian-based models utilize probability distributions to characterize traffic patterns. The Autoregressive integrated moving average model combines autoregression and moving averages to capture temporal trends. Additionally, the Historical Average method relies on past traffic data averages for prediction, among others [5].

Furthermore, within the domain of multi-source data fusion, the mixed Gaussian probability model is used to combine information from many sources. This model employs a probabilistic framework to improve the accuracy of predictions. The tensor factorization approach is used to breakdown traffic data into latent variables, hence enabling a more thorough comprehension of spatial-temporal patterns. Coupled matrix methods are used to effectively include information from several matrices, enabling the analysis of intricate interactions across diverse data sources. However, the variability in traffic patterns and the efficacy of these approaches remain insufficient in reliably predicting outcomes [6]. In recent years, the use of deep learning methods [7] has shown encouraging outcomes in a dynamic prediction model. Deep learning techniques have gained popularity in the field of traffic flow prediction due to their ability to learn different levels of features from spatial-temporal data. Recurrent Neural Networks (RNNs) are particularly effective in capturing temporal variations in traffic patterns by utilizing recurrent connections to retain information from previous time steps [8]. Convolutional Neural Networks (CNNs) excel at extracting spatial features through convolutional operations [9]. Graph Convolutional Networks (GCNs) are capable of modeling complex relationships between different stations by incorporating historical traffic passenger flows and linkages. This integration of historical data enhances the accuracy of traffic flow predictions [10]. These aforementioned methodologies take into account both spatial reliance and temporal dependency. However, they disregard the modeling of linkages concerning dynamic traffic stations using historical traffic passenger flows [11]. Considering both possible links and traffic topological structures across stations based on previous traffic passenger flows will assist in making precise predictions of traffic flows [12].

The primary motivations behind this paper are (1) the intricate nature of anticipating traffic flow in vehicular networks, (2) the widespread issue of determining optimal routes to avoid traffic congestion, and (3) the significant impact of traffic congestion on both national economies and human well-being. This research aimed to accomplish multiple objectives based on these motives. Initially, machine learning algorithms to uncover concealed patterns that could potentially impact traffic. After that, a predictive model is built that can consistently and accurately predict the upcoming volume of irrigated flow, which is not addressed in previous studies as mention in table 1 it provides a summary of the comparative analysis of different machine learning methods used in previous studies. This is anticipated to result in enhanced traffic management decision-making. Furthermore, traffic data is characterized by its significant size, disorderliness, lack of linearity, constant change, lack of stability, unpredictable behaviors, and limited association between its attributes and the aim of prediction. These concerns are also discussed in this paper.

In order to achieve the aforementioned research goals, this study enhances the rate at which typical Multilayer Perceptron-Neural Networks (MLP-NN) converge during training by substituting the Gradient Descent algorithm with the Quasi-Newton approach. Consequently, this work developed a novel prediction model capable of accurately forecasting traffic flow in VANET, while also ensuring efficient computational time.

Table 1. A comparative analysis of recent studies within the field of Machine Learning.

Reference	Prediction Techniques	Dataset	Prediction condition	Comb. or single use	RMSE
[13]	SVM, ANN, LSTM	Highway England (HE)	Traffic Monitoring Unit	Single	10.5, 12.4 8.4
[14]	Reinforcement learning-based method	HE	Highway	Single	2.7
[15]	(ARIMA – MLP), (ARIMA – RNN)	HE	Highway	Comb.	0.84, 0.81
[16]	MGCRN	Performance Measurement System (PeMS)	Expressway	Comb.	4.42
[17]	STGM	PeMS	Freeway	Comb.	4.369
[18]	DDSTGNN	PeMS	Freeway	Comb.	4.30
[19]	ANN	Maryland 511	Highway	Comb.	N/A
[20]	SVM, Random Forest, Etc.	HE	Urban highway	Single and comb	0.179, 0.178

Materials and Methods:

Technical Indicators Generation

The main idea behind a technical indicator refers to the results of mathematical calculation applied to charts for long or short prediction on different time frame configurations [21].

1 Average True Range (ATR):

ATR is a statistic often used for evaluating market volatility. This is accomplished by analyzing the whole spectrum of an asset's price movements over a predetermined period. The ATR is a statistical metric used to assess the dispersion and variability of a certain dataset over a defined time interval [22]. The calculation of ATR can be determined using (1) provided.

$$ATR_n = \frac{1}{n}\sum_{i=1}^{n} TR_i \quad (1)$$

Where:

$$TR_i = Max\{A_n, B_n, C_n\} \quad (2)$$

$$A_n = HighestFlow_n - LowestFlow_n\,,\;\; B_n = |HighestFlow_n - Flow_n|$$

$$C_n = |LowestFlow_n - Flow_n|$$

2 Exponential Moving Average (EMA):

This specific technical indication is well acknowledged and often used. The EMA is a unique method of calculating an average that takes into account past data with varying weights. To accomplish this goal, the EMA approach is used to alleviate the influence of stochastic flow variations by computing the mean flow within a designated temporal interval. The EMA is calculated based on previous data, which makes it a lagging indicator. This statement suggests that EMA may not possess the capacity to predict nascent trends, but it may confirm the direction of an already established trend [23]. The calculation of EMA can be determined using the formula provided in (3).

$$EMA = \frac{d_t + \alpha * d_{t-1} + \cdots \alpha^t * d_0}{1 + \alpha + \cdots \alpha^t} \tag{3}$$

where:

$\alpha = \frac{s-1}{s+1}$ is the weighting term

d_t flow at time t

3 Relative Strength Index (RSI):

The oscillator indicator analysis the comparative movements of recent instances of unobstructed flow and disruption. The indicator demonstrates periodic variation within a numerical interval spanning from 0 to 100. A score around 100 suggests that a significant proportion of traffic flow units within the given time frame are categorized as flow Up, whilst a value nearing 0 indicates that the bulk of traffic flow units are classed as density Down. The computation of flow Up and flow Down in the hourly models included calculating the mean of the previous s hours, where n was set to 1 to reflect the hourly pattern. The determination of upward flow and downward flow was accomplished by the use of a piecewise function. The function was designed to assess if the disparity in flow exceeded zero. In such instances, the assigned value was denoted as flow Up. Conversely, if the disparity in flow did not exceed zero, the assigned value was denoted as flow Down [23]. The calculation of RSI can be determined using the formula provided in (4) below.

$$RSI_n = 100 - \left[\frac{100}{D_n}\right] \tag{4}$$

where:

$$D_n = \left[1 - \frac{\frac{1}{n}\sum_{i=1}^{n} Flowup[flow_i - flow_n]}{\frac{1}{n}\sum_{i=1}^{n} flowdown[flow_i - flow_n]}\right] \tag{5}$$

4 Rate of Change (ROC):

The ROC, also known as the Rate of Change, is a kind of oscillator that has resemblance to the Momentum (MOM) indicator. The measurement quantifies the extent of a variable's change as a proportion rather than a fixed quantity. The ROC is a widely used quantitative measure for evaluating change. It may be used to identify situations of unimpeded flow or substantial interruption, which have traditionally been indicative of an imminent change in a certain pattern. It is important to acknowledge that a positive value of the ROC indicates a prevailing upward trend, while a negative value indicates a prevailing downward trend. The ROC does not provide much predictive efficacy in discerning forthcoming traffic density trends [22]. The calculation of ROC can be determined using the formula provided in (6).

$$ROC = (\text{Current flow / flow of n bars ago}) - 1.0) * 100 \quad (6)$$

Traffic flow prediction techniques

Several familiar ML algorithms could be used to predict a given problem with a set of features.

1 K-Nearest Neighbour (KNN)

The KNN algorithm is a form of supervised learning. This tool is versatile and can be applied to both classification and regression tasks. The KNN method utilizes the complete dataset in order to make predictions. When we have an observation that is not included in the dataset and we wish to make a prediction for it, the algorithm searches for the K cases in the dataset that are most similar to our observation [24]. Subsequently, the algorithm utilizes the associated pictures of these K Neighbours to compute the output value of the observation that we aim to forecast. KNN is an effective method for detecting intrusions, safeguarding privacy, ensuring cluster stability, and determining location [25].

2 Naive Bayes (NB)

NB classification and prediction method is a supervised machine learning algorithm that categorizes a collection of observations based on predefined rules established by the algorithm [26]. NB classifier operates under the assumption that the classes of the training dataset are already known, which makes it a supervised tool. NB method is suitable for addressing several VANET difficulties, including driver behaviour prediction, broadcast storm avoidance, and misbehaver awareness [27]. This method is resilient against irrelevant qualities, straightforward to implement, and has low training requirements. Bayes' theorem offers a method for calculating the posterior probability P(c|x) based on the prior probability P(c), the marginal likelihood P(x), and the likelihood P(x|c). Consider the (7) presented below:

$$P(c|x) = \frac{P(X|C)P(c)}{P(x)} \quad (7)$$

where:

P(c|x) is the posterior probability of class (c, target) given predictor (x, attributes).

P(c) is the prior probability of class.

P(x|c) is the likelihood which is the probability of the predictor given class.

P(x) is the prior probability of the predictor.

3 Multilayer Perceptron (MLP)

The MLP approach consists of neurons that employ differentiable functions [28]. Network consists of one or more concealed layers. These networks have a significant level of connectedness, which is determined by neuronal connection strengths, in order to analyse a substantial portion of the complexity associated with them. This complexity arises from the non-linear behaviour and extensive interconnections among their neurons. Fig. 1 illustrates the presence of three primary layers in the MLPNNs. The input layer, denoted as (1, 2., ..., n), is the first layer responsible for transmitting a signal to the network neurons. The hidden layer, denoted by the indices (1, 2, ..., h), is the middle layer in the neural network. On the other hand, the output layer, denoted by the indices (1, 2, ..., 0), is the final layer. All the computations of the system are performed within the concealed layer. The results for that network are derived from the third layer. Furthermore, the network includes bias nodes denoted as (b_{hidd}) and (b_{out}). The output layer and hidden layer are connected to one another. Furthermore, the network includes synaptic weight values $W_{in,out}$ that are assigned between the input layer and hidden layer, where W_{hi}, out represents the synaptic weight values assigned between the hidden layer and output layer of the network. The Backpropagation (BP) algorithm is commonly employed as a supervised learning technique to train MLP. By modifying the weight values, the network error was effectively managed and decreased [29]. In this study, the error refers to the MSE between the simulated output of the network and the original output. Equation (8) describes the relationship between the input x_t and the output y_t:

$$Y_t = F\left(w_0 + \sum_{a=1}^{h} w_a * F\left[w_{a,0} + \sum_{b=1}^{m} w_{b,a} * x_t\right]\right) \tag{8}$$

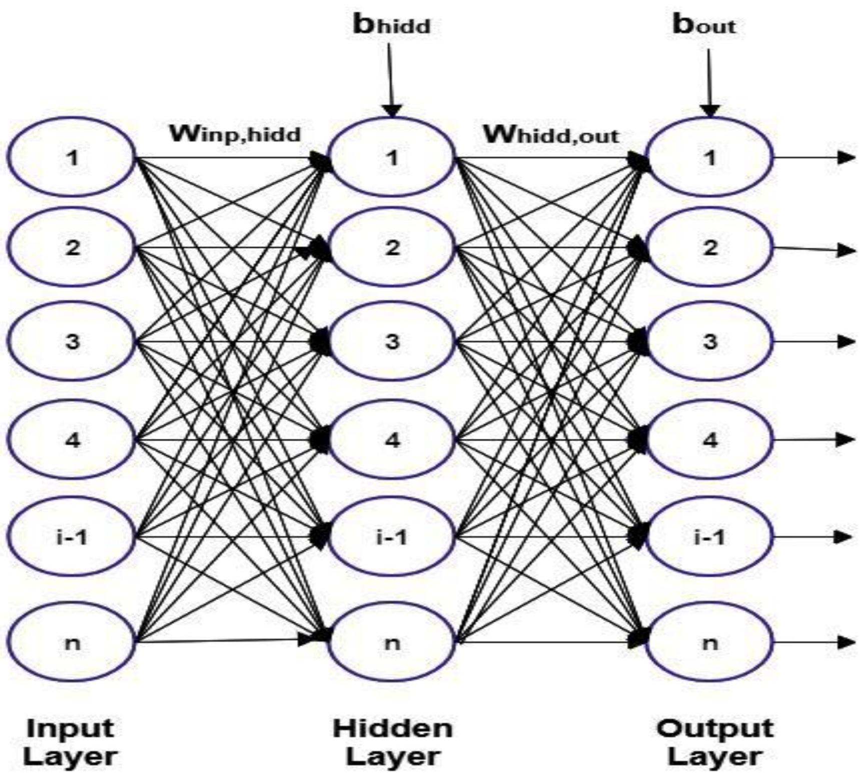

Fig. 1 The MLP-NN [9].

Optimization method

1 Gradient Descent (GD)

The first derivative is used in GD method to find the local optimum solution for a function that is differentiable f: .The approach traces the steepest descent from the current position by recursively moving in the reverse direction of the function-gradient [30]. Consequently, the Gradient Descent (GD) process undergoes iterations as seen in (9).

$$w^{(i+1)} = w^{(i)} - g^{(i)}\eta^{(i)} \tag{9}$$

where i=0,1, 2, and the factor η indicates the learning rate.

2 Newton's Method

It is approach necessitates the explicit storing and calculation of the (Hessian matrix (H)). It aims to locate the point x by solving the equation $f'(x) = 0$. This is accomplished by approximating function (f') as a (linear function (q)) and subsequently finding the root of this function. The root of the q function does not necessarily have to coincide with the root of f' prime, although it could be a reasonable assumption. Newton's technique utilizes the estimation of the 1st derivative f', but it explicitly computes the 2nd derivative f'', therefore necessitating more stringent prerequisites. Conversely, it will enhance the rate of convergence [31]. Consequently, the methodology used by Newton involves a series of iterative steps as show in (10).

$$w^{(i+1)} = w^{(i)} - H^{(i)-1}.g^{(i)} \tag{10}$$

where H indicates to Hessian Approximation

3 Quasi-Newton Methods

The Quasi-Newton approach is based on the utilization of a symmetric positive definite matrix, denoted as B, which serves as an approximation of either (H) matrix or its inverse. Instead of using the Hessian matrix, an alternative approach is employed because to the high computational cost and occasional unavailability of this matrix. Matrix B can manifest irrespective of the matrix employed. The object in question has clear positive symmetry. In each iteration, a proper matrix update is executed, resulting in the same outcome for the search [32]. Hence, the update of symmetric positive matrix is a crucial matter. The variable metric approach, known as the (Broyden Fletcher Goldfarb Shanno) (BFGS) algorithm, is renowned for its utilization of a rank two variation of the classic B. There are different updating techniques, however actual data shows that the BFGS approach is more successful with the imprecise streak search. Therefore, this method is preferred in practice [33]. The sole distinction between the Quasi-Newton approach and the Newton method pertains to the variables H and B. The Quasi-Newton equation has been expressed mathematically in (11).

$$w^{(i+1)} = w^i - \left(B^i * g^i\right)\eta^i \tag{11}$$

Methods of evaluation:

This study used two approaches, namely Root Mean Squared Error and Mean Absolute Percentage Error, to assess the prediction models.

1 Root Mean Squared Error (RMSE)

RMSE is a commonly employed metric for quantifying the disparities between observed values and model-predicted values [22], as depicted in (12).

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{n}(Actual_i - predicted_i)^2} \tag{12}$$

2 Mean Absolute Error (MAE):

The metric MAPE, as denoted in (13), is a machine learning and statistical measure that measures the Mean Absolute Percentage Error between the actual and anticipated values in a given dataset. The assessment quantifies the mean percentage deviation between the observed and projected values, so providing a metric to evaluate the overall precision of forecasts in reference to the actual values [34].

$$MAPE = \frac{1}{N}\sum_{i=1}^{n}\left|\frac{Actual_i - Predicted_i}{Actual_i}\right| \tag{13}$$

The proposed model

Fig. 2 illustrates the arch. of the proposed model for traffic flow prediction, highlighting its simplicity and efficient performance. Comprehensive depiction of the suggested model is presented in the following subsections:

1 Traffic flow datasets

This study is based on an empirical test with three historical data sets for three of the busiest highways in the United Kingdom and the United States. The study period started from 1/1/2023 to 30/1/2023 for all selected highways. Historical data is collected continuously daily.

- *Highway England (HE)* provides information on the speed and amount of traffic during a 15-minute interval. These datasets cover about one-third of all motor vehicle traffic in UK [35].
- *Performance Measurement System (PeMS):* The system offers up-to-date information on traffic volume, as well as a decade's worth of historical data, with a 5-minute frequency. The data is collected from over 39,000 probes located in both suburban and urban parts of California. Additionally, the system gives other relevant information on road conditions, such as incidents and lane closures [36].
- *Maryland 511 (MD):* It offers up-to-date photos of road conditions and traffic flow data at 15-minute intervals. In addition, this system also provides weather conditions [37].

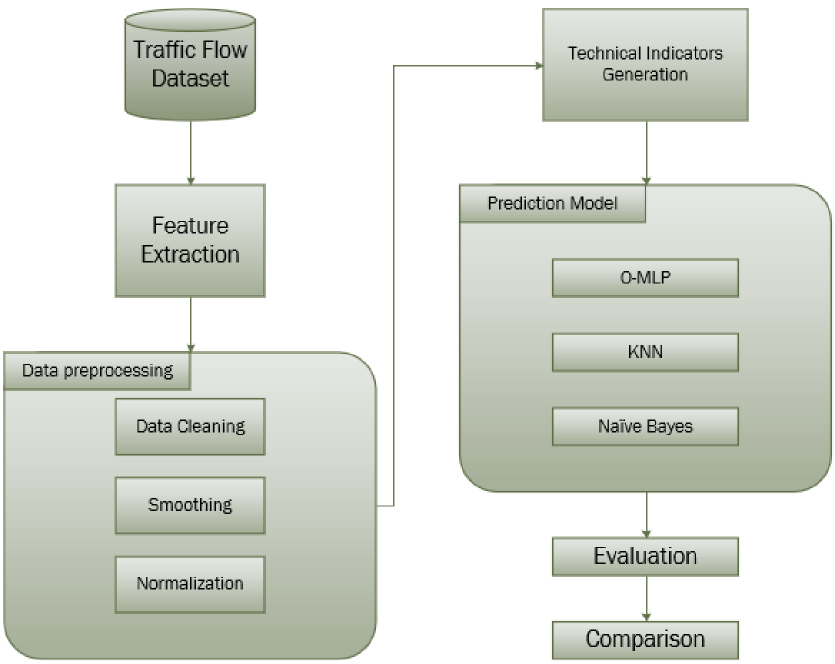

Fig. 2 The proposed system.

2 Feature extraction

The first phase consisted of extracting conventional features from the dataset that included the traffic flow metrics. Included in this pack are the following items:

- *Site Name:* A concise depiction of the website.
- *Local Date:* Date specific to British Summer Time (BST).
- *Local Time:* Time intervals of 15 minutes based on BST.
- *Total Carriageway Flow:* The total count of vehicles observed on any lane during a 15-minute interval.

3 Pre-processing stage

During this stage, the data underwent processing and was made ready for the prediction stage.

- *Data Cleaning:* The missing numbers were estimated using the historical average, taking into account the current nearby flow.
- *Smoothing:* seeks to reduce the impact of noise and variability in datasets. According to [38], it enhances the representation of fundamental patterns and trends in the data, hence aiding the recognition and comprehension of significant patterns by machine learning algorithms. The calculation of the exponentially smoothing statistic for the Y series can be done iteratively, as demonstrated in (14).

$$S_0 = Y_0$$

$$\text{For } t > 0, S_t = \alpha * Y_t + (1 - \alpha) + S_{t-1} \quad (14)$$

Let α represent the smoothing factor, where α is a value between 0 and 1, exclusive.

- Data normalization: The Z-score function can be used to normalize input features. The representation of this function is as (15):

$$v' = \frac{v - \mu_A}{\sigma_A} \quad (15)$$

Let A represent the original dataset, μ denote the mean value of the feature, and σ represent the standard deviation of the feature.

4 Technical Indictors Generation Stage

Technical indicators (TIs) are valuable for forecasting future vehicle flow and can be included into traffic management systems. Consequently, the objective of this step is to improve the precision of forecasting by utilizing the standard features (consisting of four attributes) as input for the models. The results obtained from this stage consist of the four TIs features. Additional information regarding the TIs generating stage is provided in Algorithm 1. The construction of this method is based on Equations (1) to (6).

Algorithm 1: The TIs Generation stage for Datasets
Input: The dataset is represented as an array S of dimensions n×m×r, where n represents the number of data points, m indicates the number of standard technical attributes, and r represents the historical data size.
Output: Array **O1** $(n \times j \times r)$, where j is the number of Technical Indicator features.

Begin
1. Let k represent the period sliding window.
2. Initialize k ← 4.
3. Initialize an empty array O1 of size $(n \times j \times r)$ to store the Technical Indicator features.
4. For each data point t in S do
5. For i ← 0 to m do
6. Execute **ATR**(Density[t], High[i], Low[i], Density, t, i, k) in parallel and save the output in the **ATR(i)**.
7. Execute **EMA** (Density, t, i, r) in parallel and save the output in the **EMA(i)**.
8. Execute **RSI** (Density, t, i, r) in parallel and save the output in the **RSI(i)**.
9. Execute **ROC** (Density, t, i, r) in parallel and save the output in the **ROC(i)**.
10. End For
11. End For
12. Merge array **S** of size (n×m×r) with array **O1** of size (n×j×r) to create Input2 of size (n×|S+O1|×r).
End.

5 Prediction model

For the purpose of comparison with the suggested prediction model, two widely-used prediction methodologies were employed in this study. Initially, the K-Nearest Neighbors algorithm was utilized to forecast forthcoming traffic patterns. Through experimentation, it was determined that a value of five for K yielded the highest level of prediction accuracy in this investigation. Furthermore, the Naive Bayes method was utilized to forecast forthcoming traffic patterns.

The proposed model was built using a well-established prediction model. Input characteristics of the proposed model consist of the technical analysis indicators that were computed in the preceding stage using Algorithm 1. The task was accomplished by utilizing Optimize-Multilayer Perceptron (O-MLP) neural networks, which were trained using the Back Propagation algorithm with one hidden-layer. Training process involved minimizing the specified loss function using the Quasi-Newton method, which is based on projected gradient, instead of the traditional Gradient-Descent approach. Three neurons were identified in the hidden layer, along with a bias. The penalty for exceeding the weight limit is 0.02, while the tolerance value is 1.0E-4. Furthermore, a modified form of the logistic function known as the hyperbolic tangent (tanh) function was employed as the activation-function for the hidden layer in order to enhance the efficiency at execution time. A linear function was employed in the output layer to forecast the traffic flow.

The RMSE and MAE metrics were utilized to assess the performance of the suggested model and to compare its results with those obtained from the Naive Bayes and KNN approaches. The optimization model comprises numerous steps, as seen in algorithm 2. It aids in error reduction by rendering the loss function gentler, hence expediting convergence.

Algorithm 2.
Input: Initial Parameters (Q_{vk}), Converges Factor (CF)
Output: Final Weight Vector
Begin
Define new symbols:
W_k: Current weight vector in the neural network
CF: Converges Factor
k: Iteration counter
Ridge: Regularization parameter (e.g., 0.01)
G_k: Jacobian (gradient) vector at iteration k
B_k: Symmetric positive definite matrix at iteration k
d_k: Search direction
a: Step size for line search
DW_k: Change in weight vector
DG_k: Change in gradient vector
Step 1: Initialization.
- Ridge = 0.01
- k=1
- Set the initial values of variable W_k
- Compute the Jacobian vector G_k by utilizing W_k.
- Initialize a B_k matrix

Step 2: If the norm of G_k tends to 0, then cease.
Step 3: Determine the search direction in the form of:
- $d_k = -B_k^{-1}G_k$

Step 4: Utilize line search to find an appropriate step size α and thereafter modify the weight vector.
$W_{k+1} = W_k + \alpha d_k$
Step 5:
- Determine G_{k+1} by utilizing W_{k+1}:
  $G_{k+1} = G_k + Ridge * W_{k+1}$
- Calculate changes in the weight and gradient vectors:
  $DW_k = W_{k+1} - W_k$
  $DG_k = G_{k+1} - G_k$
- Update the B_k matrix using the BFGS update formula:
  $B_{k+1} = B_k + \frac{DG_k * DG_k^T}{DG_k^T * DW_k} - \frac{B_k * DW_k * DW_k^T * B_k}{DW_k^T * B_k * DW_k}$

Step 6: k = k + 1 and then return to Step 2.
End.

6 Deep Traffic Flow Optimization Model

The Traffic Flow Prediction (TFP) model is structured with four layers: input, statistical, learning weights, and output. It incorporates the diversification principle of the Markowitz model, the local search capability of the Quasi-Newton optimization algorithm, and the adaptability of deep learning. The process involves computing traffic flow predictions, assessing associated risks, utilizing Quasi-Newton for weight optimization, and determining the most effective traffic flow predictions. TFP efficiently tackles computational challenges by employing Quasi-Newton to fine-tune network weights. While originally designed for long-term traffic flow forecasting, an Integrated Traffic Prediction (ITP) system has been introduced. This system combines Deep Learning Regression (DLR) to predict future traffic flow with TFP for optimal weight allocation, adapting it for short-term traffic predictions. ITP significantly improves traffic flow optimization, prediction accuracy, and reduces implementation time, as outlined in Fig. 3.

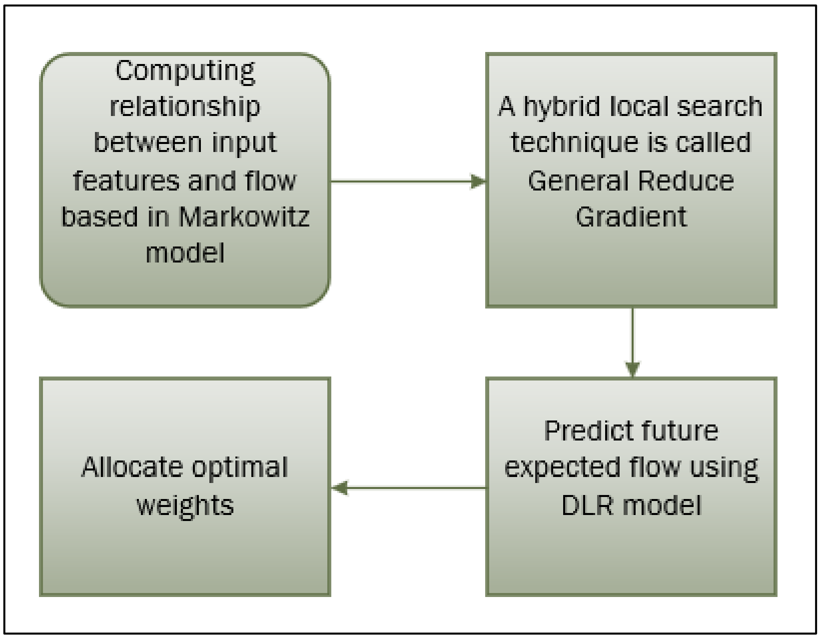

Fig. 3 The proposed methodology for traffic flow optimization.

Results and Analysis

The outcomes for various techniques are condensed in Table 2. Table 2 reveals that the proposed model has an average MAE of 0.84, significantly surpassing the other models. Similarly, according to the data presented in Table 2, it can be demonstrated that the proposed model has a lower RMSE of 0.07. This suggests that the proposed model exhibited superior efficiency.

Table 2: Comparative MAE, RMSE for the 3 datasets.

Model	England Highway		PeMS		Maryland 511	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
O-MLP	1.07	0.10	0.85	0.04	0.61	0.07
KNN	5.91	7.02	4.88	5.45	5.58	8.71
Naïve bayes	7.07	8.95	8.22	8.86	7.53	9.12

While predicting traffic patterns can be challenging due to the constantly changing nature of traffic, the suggested model demonstrates favorable outcomes, suggesting its success in accurately forecasting traffic at this specific timeframe and under these particular circumstances. Upon examining the findings, it is evident that the suggested model has superior performance when all evaluation metrics are applied. The suggested model demonstrates the lowest RMSE of 0.07 for the Maryland 511 dataset, while it exhibits the highest MAE of 0.1 for the HE dataset. The PeMS dataset achieved the lowest RMSE of 5.45 using the KNN approach, while the Maryland 511 dataset had the highest RMSE of 9.12 using the Naive Bayes method. Overall, the outcomes of the suggested model surpassed those of the other two methods. The probable cause for this is the implementation of the optimization method, along with the utilization of technical analysis indicators, which enhances the models' capacity to predict in the near future.

The outcomes of the suggested model were contrasted with conventional prediction models using an identical dataset. Nevertheless, the researcher managed to juxtapose the research findings with prior literature that employed an identical dataset. It is necessary to note that research might differ in terms of duration, technique, and evaluation procedures. The prediction model in this paper is designed to forecast traffic flow in the vehicular network using a regression model, which is an improvement over earlier research.

Conclusion

This study constructed a feed-forward neural network forecaster using past data to anticipate future traffic flow and produce strategies for efficient traffic control. The model employed four impactful input features, namely ATR, EMA, RSI, and ROC, to accurately anticipate short-term traffic flow. To enhance the accuracy of the prediction, the MLP is enhanced with a novel hybrid algorithm that accelerates the convergence towards the ideal solution, ensuring precise prediction outcomes. The proposed model can serve as a valuable tool for transport agencies to make informed decisions on traffic management and the prevention of traffic congestion. The justification for this approach is based on the utilization of previous traffic data to identify any valuable forecasting insights. The suggested model's findings are optimal, as determined through a comparison with other prediction models such as Naive Bayes and KNN. The proposed model achieved an improvement in the RMSE measure for O-MLP compared to KNN of about 98.58%, and compared to Naïve Bayes of about 98.88% for the England Highway data set, with almost similar percentages for the other data sets. Notwithstanding the significant contributions and discoveries made by this research, it is not exempt from constraints. Firstly, this study does not investigate the effects of exogenous factors such as seasonal variations or accidents on traffic. Future efforts should focus on constructing an integrated predictive system by consolidating a comprehensive dataset for various traffic scenarios, in order to overcome these limitations. Long Short-Term Memory (LSTM) can be used to model and predict the impact of seasonal changes on traffic patterns over time, and Convolutional Neural Networks (CNN) can help predict the impact of accidents on traffic flow at different locations.

Abbreviation

ANN: Artificial Neural Network
ARIMA: Autoregressive integrated moving average
DDSTGNN: Decoupled Dynamic Spatial-Temporal Graph Neural Network
LSTM: Long Short-Term Memory
MGCRN: Meta-Graph Convolutional Recurrent Network
RNN: Recurrent Neural Networks
STGM: Spatio-Temporal Graph Mixformer
SVM: Support Vector Machine

Reference

1. Gao, Y., Li, Z., & Kang, D. S. (2018). Supply chain coordination: a review. Journal of Advanced Management Science, 6(4).

2. Turki, A. I., & Hasson, S. T. (2022, November). A Markova-Chain Approach to Model Vehicles Traffic Behavior. In 2022 International Conference of Science and Information Technology in Smart Administration (ICSINTESA) (pp. 117-122). IEEE.

3. Deng, S., Jia, S., & Chen, J. (2019). Exploring spatial–temporal relations via deep convolutional neural networks for traffic flow prediction with incomplete data. Applied Soft Computing, 78, 712-721.

4. Emami, A., Sarvi, M., & Bagloee, S. A. (2020). Short-term traffic flow prediction based on faded memory Kalman Filter fusing data from connected vehicles and Bluetooth sensors. Simulation Modelling Practice and Theory, 102, 102025.

5. Ahmed, A. A., & Pradhan, B. (2019). Vehicular traffic noise prediction and propagation modelling using neural networks and geospatial information system. Environmental monitoring and assessment, 191(3), 190.

6. Chen, Q., Song, Y., & Zhao, J. (2021). Short-term traffic flow prediction based on improved wavelet neural network. Neural Computing and Applications, 33, 8181-8190.

7. Li, W., Wang, J., Fan, R., Zhang, Y., Guo, Q., Siddique, C., & Ban, X. J. (2020). Short-term traffic state prediction from latent structures: Accuracy vs. efficiency. Transportation Research Part C: Emerging Technologies, 111, 72-90.

8. Raj, D. J. S., & Ananthi, J. V. (2019). Recurrent neural networks and nonlinear prediction in support vector machines. Journal of Soft Computing Paradigm, 1(1), 33-40.

9. Du, B., Peng, H., Wang, S., Bhuiyan, M. Z. A., Wang, L., Gong, Q., ... & Li, J. (2019). Deep irregular convolutional residual LSTM for urban traffic passenger flows prediction. IEEE Transactions on Intelligent Transportation Systems, 21(3), 972-985.

10. Zhang, Q., Jin, Q., Chang, J., Xiang, S., & Pan, C. (2018, August). Kernel-weighted graph convolutional network: A deep learning approach for traffic forecasting. In 2018 24th international conference on pattern recognition (ICPR) (pp. 1018-1023). IEEE.

11. Peng, H., Wang, H., Du, B., Bhuiyan, M. Z. A., Ma, H., Liu, J., ... & Philip, S. Y. (2020). Spatial temporal incidence dynamic graph neural networks for traffic flow forecasting. Information Sciences, 521, 277-290.

12. Benocci, R., Molteni, A., Cambiaghi, M., Angelini, F., Roman, H. E., & Zambon, G. (2019). Reliability of Dynamap traffic noise prediction. Applied Acoustics, 156, 142-150.

13. Sun, P., Aljeri, N., & Boukerche, A. (2020). Machine Learning-Based Models for Real-time Traffic Flow Prediction in Vehicular Networks. IEEE Network, 34(3), 178-185.

14. Chen, X., & Chaudhari, P. (2021). MIDAS: Multi-agent Interaction-aware Decision-making with Adaptive Strategies for Urban Autonomous Navigation. In 2021 IEEE International Conference on Robotics and Automation (ICRA) (pp. 7980-7986). Xi'an, China.

15. Rajalakshmi, V. & Ganesh Vaidyanathan, S. (2022). Hybrid Time-Series Forecasting Models for Traffic Flow Prediction. Promet–Traffic and Transportation, 34 (4), 537-549.

16. Jiang, R., Wang, Z., Yong, J., Jeph, P., Chen, Q., Kobayashi, Y., ... & Suzumura, T. (2023, June). Spatio-temporal meta-graph learning for traffic forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 37, No. 7, pp. 8078-8086).

17. Lablack, M., & Shen, Y. (2023). Spatio-temporal graph mixformer for traffic forecasting. Expert Systems with Applications, 228, 120281.

18. Shao, Z., Zhang, Z., Wei, W., Wang, F., Xu, Y., Cao, X., & Jensen, C. S. (2022). Decoupled dynamic spatial-temporal graph neural network for traffic forecasting. arXiv preprint arXiv:2206.09112.

19. Sekuła, P., Marković, N., Vander Laan, Z., & Sadabadi, K. F. (2018). Estimating historical hourly traffic volumes via machine learning and vehicle probe data: A Maryland case study. Transportation Research Part C: Emerging Technologies, 97, 147-158.

20. Rahi, A. A. (2019). Machine Learning Approaches for Traffic Flow Forecasting (Doctoral dissertation). https://doi.org/10.18745/th.22590

21. Wu, S., Li, G., Deng, L., Liu, L., Wu, D., Xie, Y., & Shi, L. (2018). L1 -norm batch normalization for efficient training of deep neural networks. IEEE transactions on neural networks and learning systems, 30(7), 2043-2051.

22. Turki, A. I., & Hasson, S. T. (2023). Study Estimating hourly traffic flow using Artificial Neural Network: A M25 motorway case. Samarra Journal of Pure and Applied Science, 5(1), 47-59.

23. Turki, A. I., & Hasson, S. T. (2023). Using a new algorithm in Machine learning Approaches to estimate level-of-service in hourly traffic flow data in vehicular ad hoc networks. Tikrit Journal of Pure Science, 28(3), 74-83.

24. Cheng, S., Lu, F., Peng, P., & Wu, S. (2018). Short-term traffic forecasting: An adaptive ST-KNN model that considers spatial heterogeneity. Computers, Environment and Urban Systems, 71, 186-198.

25. Luo, X., Li, D., Yang, Y., & Zhang, S. (2019). Spatiotemporal traffic flow prediction with KNN and LSTM. Journal of Advanced Transportation, 2019.

26. Liu, T., Shi, S., & Gu, X. (2020). Naive Bayes classifier-based driving habit prediction scheme for VANET stable clustering. Mobile Networks and Applications, 25, 1708-1714.

27. Guo, X., Chen, Y., Cao, L., Zhang, D., & Jiang, Y. (2020). A receiver-forwarding decision scheme based on Bayesian for NDN-VANET. China Communications, 17(8), 106-120.

28. Hussein, A. A., Yaseen, E. T., & Rashid, A. N. (2023). Learnheuristics in routing and scheduling problems: A review. Samarra Journal of Pure and Applied Science, 5(1), 60-90.

29. Liu, Y., Sun, Y., Xue, B., Zhang, M., Yen, G. G., & Tan, K. C. (2021). A survey on evolutionary neural architecture search. IEEE transactions on neural networks and learning systems.

30. Zhang, C., Yao, M., Chen, W., Zhang, S., Chen, D., & Wu, Y. (2021). Gradient descent optimization in deep learning model training based on multistage and method combination strategy. Security and Communication Networks, 2021, 1-15.

31. Chopra, S., Yadav, D., & Chopra, A. N. (2019). Artificial neural networks based indian stock market price prediction: before and after demonetization. J Swarm Intel Evol Comput, 8(174), 2.

32. Goldfarb, D., Ren, Y., & Bahamou, A. (2020). Practical quasi-newton methods for training deep neural networks. Advances in Neural Information Processing Systems, 33, 2386-2396.

33. Zhao, W. (2021). A Broyden–Fletcher–Goldfarb–Shanno algorithm for reliability-based design optimization. Applied Mathematical Modelling, 92, 447-465.

34. Razzaq, L., Farooq, M., Mujtaba, M. A., Sher, F., Farhan, M., Hassan, M. T., ... & Imran, M. (2020). Modeling viscosity and density of ethanol-diesel-biodiesel ternary blends for sustainable environment. Sustainability, 12(12), 5186.

35. Highways England, Highways England network journey time and traffic flow data, 2015, [Online]. Available: https://data.gov.uk/dataset/highways-englandnetwork-journey-time-and-traffic-flow-data, (Accessed Oct. 2023).

36. Caltrans - State of California, PeMS data source - caltrans - state of California, 2009, [Online]. Available: http://pems.dot.ca.gov, (Accessed Oct. 2023).

37. Maryland Department of Transportation, Maryland 511, 2013, [Online]. Available: https://chart.maryland.gov, (Accessed Oct. 2023).

38. Memisoglu Baykal, T., Colak, H. E., & Kılınc, C. (2022). Forecasting future climate boundary maps (2021–2060) using exponential smoothing method and GIS. Science of The Total Environment, 848, 157633.

IRAQI Academic Scientific Journals
Samarra Journal of Pure and Applied Science
SJPAS
مجلة سامراء للعلوم الصرفة والتطبيقية
www.sjpas.com
p ISSN: 2663-7405
e ISSN: 2789-6838

التنبؤ بتدفق حركة المرور في الشبكات المخصصة للمركبات باستخدام الشبكة العصبية المحسنة

أحمد ابراهيم تركي[1]*
1- قسم الفيزياء، كلية التربية، جامعة سامراء، العراق

معلومات البحث:
تأريخ الاستلام:
تاريخ التعديل :
تأريخ القبـــول:
تاريخ الــنشر:
الكلمات المفتاحية:
تدفق حركة المرور، شبكة المركبات المخصصة، الشبكة العصبية المطورة، التحسين، التنبؤ

معلومات المؤلف
الايميل: ahmed.ibrahim@uosamarra.edu.iq
الموبايل: 009647704290098

الخلاصة:
يعد التنبؤ بتدفق حركة المرور أمرًا بالغ الأهمية لإدارة حركة المرور، وتقليل التلوث، وضمان السلامة العامة، وهو عنصر حيوي في أنظمة النقل الذكية. علاوة على ذلك، فإن المتغيرات التي تؤثر على أنماط حركة المرور (مثل الأحداث العامة وإغلاق الطرق والحوادث) تكون في الغالب غير متوقعة، مما يجعل التنبؤ بتدفق حركة المرور مهمة معقدة. الهدف من هذا البحث هو تعزيز دقة التنبؤ بتدفق حركة المرور من خلال استخدام استراتيجية جديدة تستخدم الشبكات العصبية المغذية وطريقة التحسين شبه نيوتن. تقلل الإستراتيجية المقترحة من عامل الخطأ المعتمد على مضاعف لاغرانج ومتجه جاكوبي. وقد أدى هذا التعزيز إلى التقارب السريع طوال عملية التعلم. تم اختيار العينة باستخدام مجموعات البيانات المقدمة من نظام مراقبة حركة المرور، وتحديداً من Highway England، ونظام قياس الأداء، وMaryland 511، لعام 2023. ومن أجل تقييم النموذج المقترح، يتم وضع نتائج البحث جنبًا إلى جنب مع منهجيات التنبؤ الأخرى المعمول بها. وكشفت الدراسة أن نموذج البحث الذي تم وضعه كان أداؤه أفضل من استراتيجيات التنبؤ السابقة بناءً على مقاييس متوسط الخطأ المطلق وجذر متوسط الخطأ التربيعي.