AUT Journal of Civil Engineering

AUT J. Civil Eng., 6(3) (2022) 369-386
DOI: 10.22060/ajce.2023.22199.5821

The short-term prediction of traffic parameters: a review of parametric and nonparametric approaches

A. Taheri[1], R. Moghadasi[2], A. Rasaizadi[2]

[1] Department of motor vehicles, Technical University of Berlin, Berlin, Germany.
[2]Department of transportation planning engineering, Tarbiat Modares University, Tehran, Iran.

ABSTRACT: Over the past few decades, there has been a lot of interest in short-term traffic prediction which is a crucial part of transportation systems. One of the major considerations that travelers evaluate when making travel plans is information regarding the near future. Transportation planners have investigated a number of methods to produce more reliable travel time predictions in the future. However, there have not been enough in-depth and comprehensive surveys in this area yet. In this paper, a comprehensive review of the literature has been conducted, and various traffic parameter prediction algorithms have been investigated. The methods can be divided into two main groups: parametric, and nonparametric. Parametric models are those that require the specification of some parameters before they can be used to make predictions. In contrast to nonparametric procedures, parametric approaches have a distribution with a defined number of parameters. The parametric approaches are related to statistical methods like time-series, while the nonparametric approaches are related to machine learning methods like neural networks. Predicting flow, volume, speed, density, and occupancy are the main emphasis of the majority of the literature. A detailed methodology is first presented for each of the techniques mentioned in this article, and then the outcomes of numerous articles that have used the technique are discussed.

Review History:

Received: Feb. 13, 2023
Revised: May, 14, 2023
Accepted: May, 31, 2023
Available Online: Jun. 01, 2023

Keywords:

Traffic prediction
parametric models
nonparametric models
statistical methods
machine learning methods

1- Introduction

Short-term traffic prediction is the process of estimating the number of vehicles that will be on a given road segment at a specific time in the near future. This information is crucial for traffic management and planning, as it allows transportation authorities to anticipate congestion and implement measures to alleviate it. Short-term traffic prediction is typically based on historical traffic data, weather conditions, and other relevant factors such as events or road closures. The accuracy of short-term traffic prediction has improved significantly in recent years thanks to advances in technology and data analytics [1].

The benefits of short-term traffic prediction are numerous. It helps reduce travel time for commuters, improves safety by reducing the likelihood of accidents caused by congestion, and reduces fuel consumption and air pollution by minimizing idling and stop-and-go traffic. Short-term traffic prediction is also important for emergency response planning, as it allows first responders to anticipate traffic conditions and adjust their routes accordingly. Overall, short-term traffic prediction is a critical tool for transportation management and planning, helping to improve the efficiency, safety, and sustainability of our transportation systems.

*Corresponding author's email: arash_rasaizadi@modares.ac.ir

Accurate and timely traffic state prediction is a critical research issue in transportation planning and engineering [2]. Due to this, a variety of research projects have recently been carried out that consider different approaches to this issue. By using past or current traffic data, traffic state prediction attempts to predict future traffic parameters [3]. The majority of the prediction models in the literature have focused on the flow/volume, density/occupancy, and speed of traffic [4]. Additionally, short-term predictions are those that are made for a time frame of less than an hour [5].

Parametric or nonparametric methods are often used in traffic state predictions. The parametric approaches are related to statistical methods like time-series, while the nonparametric approaches are related to machine learning methods like neural networks, etc [6]. Both parametric and nonparametric approaches are properly considered in the literature in the sections that follow.

The parametric approach is covered in Section 2. In particular, a description of time series techniques, including seasonal autoregressive integrated moving average (SARIMA), auto-regressive integrated moving average (ARIMA), and auto-regressive integrated moving average with exogenous variable (ARIMAX), is provided. In section 3, neural networks (NNs), support vector machines (SVM), support vector regression (SVR), and k-nearest neighbor regression are thoroughly investigated as non-parametric

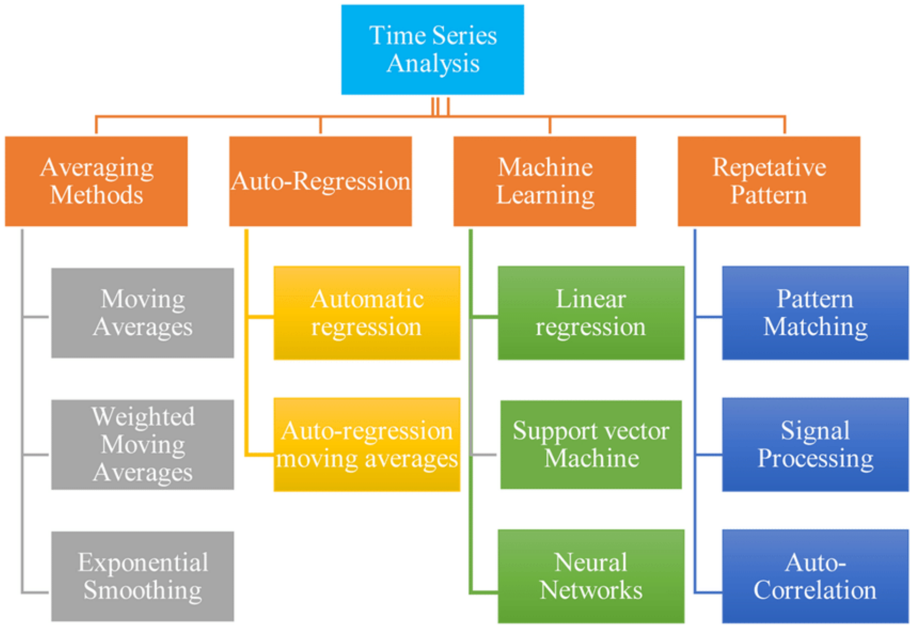

Fig. 1. Summary of predictive methods

models. The neural networks (NNs) used in this study include multilayer feedforward neural networks (MLFNN), time-delay neural networks (TDNN), recurrent neural networks (RNN), convolution neural networks (CNN), deep belief networks (DBN), radial basis function neural networks (RBFNN), wavelet neural networks (WNN), and fuzzy neural networks (FNN). The conclusion of the current investigation is offered in Section 4 as a last step.

Figure 1 shows the summary of predictive methods.

The research gaps are as following:

1. Limited availability of real-time traffic data: Short-term traffic prediction heavily relies on real-time traffic data. However, the availability of such data is limited in many regions, which makes it challenging to develop accurate prediction models.

2. Lack of standardized evaluation metrics: There is no standardized evaluation metric for short-term traffic prediction models. This makes it difficult to compare the performance of different models and to identify the best-performing ones.

3. Inability to account for external factors: Short-term traffic prediction models often fail to consider external factors such as weather conditions, events, and road closures. This limits their accuracy and reliability.

Innovations of this study include the following items:

1. Integration of machine learning techniques: Machine learning techniques such as deep learning and reinforcement learning have shown promising results in short-term traffic prediction. They can learn from historical traffic patterns and adjust their predictions based on real-time data.

2. Use of big data analytics: The availability of big data has enabled the development of more accurate short-term traffic prediction models. Big data analytics can process vast amounts of data from various sources, including social media, GPS, and traffic cameras.

3. Integration of predictive analytics: Predictive analytics can be used to forecast future traffic patterns based on historical data and external factors such as weather conditions and events. This can improve the accuracy and reliability of short-term traffic prediction models.

2- Parametric models

Parametric models are those that require the specification of some parameters before they can be used to make predictions [7]. In the following, parametric models will be examined.

2- 1- Time series

Time series forecasting occurs when you make scientific predictions based on historical time-stamped data. It involves building models through historical analysis and using them to make observations and drive future strategic decision-making [8].

2- 1- 1- Autoregressive Integrated Moving Average (ARIMA)

The autoregressive process of order p is denoted AR(p), and defined by equation (1);

$$X_t = \sum_{(r=1)}^{p} \phi_r X_{(t-r)} + \varepsilon_t \quad (1)$$

Where, $\varnothing_1,\ldots,\varnothing_r$ are fixed constants and $\{\epsilon_t\}$ is a sequence of independent (or uncorrelated) random variables with mean 0 and variance σ^2.

a)Moving average processes

The moving average process of order q is denoted MA(q) and defined by equation (2);

$$X_t = \sum_{s=0}^{q} \theta_s \varepsilon_{t-s} \tag{2}$$

where $\theta_1, \ldots, \theta_q$ are fixed constants, $\theta_0 = 1$, and $\{\epsilon_t\}$ is a sequence of independent (or uncorrelated) random variables with mean 0 and variance σ^2.

b) ARMA processes

The autoregressive moving average process, ARMA(p, q), is defined by equation (3);

$$X_t - \sum_{r=1}^{p} \phi_r X_{t-r} = \sum_{s=0}^{q} \theta_s \varepsilon_{t-s} \tag{3}$$

where again $\{\epsilon_t\}$ is white noise. This process is stationary for appropriate ϕ, θ.

c)ARIMA processes

If the original process [9] is not stationary, we can look at the first order difference process (equation (4));

$$X_t = \nabla Y_t = Y_t - Y_{t-1} \tag{4}$$

or the second order differences (equation (5));

$$X_t = \nabla_2 Y_t = \nabla(\nabla Y)_t = Y_t - 2Y_{t-1} + 2Y_{t+2} \tag{5}$$

and so on. If we ever find that the differenced process is a stationary process we can look for a ARMA model of that. The process [9] is said to be an autoregressive integrated moving average process, ARIMA(p, d, q), if Xt = ∇dYt is an ARMA(p, q) process. AR, MA, ARMA, and ARIMA processes can be used to model many time series.

In ARIMA, each component performs the role of a parameter and uses a common notation. Standard notation for ARIMA models is ARIMA with p, d, and q, where integer values are used in place of the parameters to denote the model's type [10]. The parameters can be defined as:

p: the number of lag observations in the model; also known as the lag order.

d: the number of times that the raw observations are differenced; also known as the degree of differencing.

q: the size of the moving average window; also known as the order of the moving average.

For instance, the quantity and type of terms are taken into account in a linear regression model. A parameter with a value of 0 indicates that the model should not include that specific component. In this approach, the ARIMA model can be built to serve the same purpose as an ARMA model or even just a basic AR, I, or MA model. The basis of ARIMA models is the idea that past values may still have an impact on present or future values [11].

d)The AR in [AR]IMA, Auto Regressive

The autoregressive (AR) regression model is based on the idea of autocorrelation, in which the dependent variable depends on its old values [12]. The equation (6) below is the general equation is:

$$Y_t = \beta_1 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \ldots + \phi_P Y_{t-P} \tag{6}$$

An observation at time t, Yt, is dependent on Yt-1, Yt-2,... Yt-p, as shown. The number of prior lag observations we include in the model is indicated by the p in this equation, which is known as the lag order.

e) The I in AR[I]MA, Integrated

The integrated part of ARIMA makes a forecast on the difference between any two observations rather than directly on the data in an effort to slightly improve the non-stationary of the time-series data [13]. (see equation (7)).

$$\begin{aligned} &Z_t = Y_{t+1} - Y_t \rightarrow d = 1 \\ &Q_t = Z_{t+1} - Z_t \rightarrow d = 2 \\ &\ldots \end{aligned} \tag{7}$$

Depending on the hyper-parameter "d" that is selected during the training of the ARIMA model, the differencing operations (Y →Z and Z →Q) are carried out several times.

f)The MA in ARI[MA], Moving Average

The MA, or Moving Average, is the last component of ARIMA. It applies some kind of aggregation procedure to the prior observations in terms of residual error "ε" in an attempt to reduce the noise in the time series data (see equation (8)).

$$Y_t = \beta_2 + \omega_1 \varepsilon_{t-1} + \omega_2 \varepsilon_{t-2} + \ldots + \omega_q \varepsilon_{t-q} + \varepsilon_t \tag{8}$$

The term "q" here is another hyper-parameter that is the same as "p", and "ε" represents the residual errors from the aggregation function. However, q indicates the time window for the residual error of the moving average rather than the time window "p" for the time series data itself [14].

Several researches investigate the use of ARIMA-based models to forecast traffic conditions. Alghamdi et al. (2019) [13] investigated several variables that have a considerable impact on the rate of traffic congestion using ARIMA-based modeling. In this study, a short-term time series model for non-Gaussian traffic data demonstrates good performance in forecasting traffic conditions. A proposed ARIMA model for predicting traffic flow was made by Dong et al. in 2009 [15]. The ARIMA model is more accurately refined based on the various data training periods. The studies demonstrate that the ARIMA model trained using time-oriented data can outperform the model trained using non-time-oriented data. The ARIMA model was used by Yan et al. (2019) [16] to estimate the subway's short-term traffic flow. The ARIMA model outperformed other models, notably the support vector machine (SVM), in the prediction of short-term traffic flow. To estimate multi-scale high-speed network traffic, Sadek et al. (2004) [17] introduced the Gegenbauer autoregressive moving average (GARMA) model. The GARMA model outperforms the conventional autoregressive (AR) model in terms of prediction performance.

2- 1- 2- Seasonal auto-regressive integrated moving average (SARIMA)

(The general form of seasonal model SARIMA(p, d, q) (P, D, Q)s is given by: (see equation (9))

$$\phi_p(B^s)\phi(B)\nabla_s^D\nabla^d x_t = \Theta_Q(B^s)\theta(B)w_t \qquad (9)$$

where, {wt} is the nonstationary time series, {wt} is the usual Gaussian white noise process. "S" is the period of the time series. The ordinary autoregressive and moving average components are represented by polynomials φ(B) and θ (B) of orders p and q. The seasonal autoregressive and moving average components are $\varnothing_p\left(B^s\right)$ and $\Theta_Q\left(B^s\right)$, where P and Q are their orders. ∇^d and ∇_s^D are ordinary and seasonal difference components. B is the backshift operator. The expressions are shown as follows: (see equation (10-16)

$$\varphi(B)_t = \varphi_1 B^1 - \varphi_2 B^2 - \ldots - \varphi_p B^p \qquad (10)$$

$$\Phi_p(B^s) = 1 - \Phi_1 B^s - \Phi_2 B^{2s} - \ldots - \Phi_p B^{ps} \qquad (11)$$

$$\theta(B) = 1 + \theta_1 B^1 + \theta_2 B^2 + \ldots + \theta_q B^q \qquad (12)$$

$$\Theta_Q(B^s) = 1 + \Theta_1 B^s + \Theta_2 B^{2s} + \ldots + \Theta_Q B^{Qs} \qquad (13)$$

$$\nabla^d = (1-B)^d \qquad (14)$$

$$\nabla_s^d = (1-B^s)^D \qquad (15)$$

$$B^k x_t = x_{(t-k)} \qquad (16)$$

In this study, we concentrate on monthly precipitation time series. If the seasonal period of the series s = 12. It is clear that we may then rewrite as: (see equation (17))

$$\phi_p(B^{12})\varphi(B)\nabla_{12}^D\nabla^d x_t = \Theta_Q(B^{12})\theta(B)w_t \qquad (17)$$

SARIMA models are simply ARIMA models that have a seasonal component. The following are the parameters for these kinds of models according to the formula SARIMA (p, d, q) x (P, D, q, s) [18]:

p and seasonal ***P***: indicate the number of autoregressive terms (lags of the stationarieszed series)

d and seasonal ***D***: indicate differencing that must be done to stationarize series

q and seasonal ***Q***: indicate the number of moving average terms (lags of the forecast errors)

s: indicates the seasonal length in the data

The forecast is seasonal with SARIMA, which stands for Seasonal-ARIMA. There is no doubt about the significance of seasonality, and ARIMA fails to implicitly capture that information. The model's components that are still based on ARIMA are the Autoregressive (AR), Integrated (I), and Moving Average (MA) sections. Seasonality strengthens the SARIMA model by incorporating robustness. The SARIMA is represented as follows, where m is the number of observations per year [19].

To anticipate traffic flow over the short term with only a limited amount of input data, Kumar et al. (2015) [20] proposed a prediction technique using the SARIMA model. In conclusion, the results were positive, and the traffic flow prediction technique suggested in this study could be taken into consideration in circumstances when the database is a major restriction. The SARIMA model was investigated by Ghosh et al. (2007) [21] in their study of short-term traffic flow prediction. The model inference produced a prediction interval that changed over time as opposed to the classical inference, which relied on the normal distribution assumption to produce point forecasts and a constant confidence interval. For the railroad passenger flows examined by Milenkovic et al. (2015) [22], an appropriate modeling approach was suggested. The time series that covers ten years is fitted and forecasted using the SARIMA method. The

SARIMA model could be taken into consideration for estimating monthly passenger flows on trains based on the results, which demonstrated good prediction capabilities. The SARIMA model was used by Giraka et al. (2019) [23] to propose a short-term forecast of intersection turning volume. The findings were favorable, with the majority of the cases having a Mean Absolute Percentage Error (MAPE) of less than 10.

2- 1- 3- ARIMAX

(An ARIMAX model is viewed as a multiple regression model with one or more autoregressive (AR) terms and/or one or more moving average (MA) terms. Autoregressive terms for a dependent variable are merely lagged values of that dependent variable that have a statistically significant relationship with its most recent value. Moving average terms is nothing more than residuals (i.e., lagged errors) resulting from previously made estimates. The general ARIMAX models are as follows: Autoregressive model with exogenous variables (ARX): (see equation (18))

$$yt = \varphi(L)yt + \beta xt + \varepsilon t \tag{18}$$

Moving average model with exogenous variables (MAX): (see equation (19))

$$\varphi(L)yt = \beta xt + \theta(L)\varepsilon \tag{19}$$

Autoregressive Moving Average with exogenous variables model (ARMAX): (see equation (20))

$$\Delta Yt = \varepsilon t + \sum \varphi ipi = 1\Delta Yt - i\sum \theta j\varepsilon t - j + qj = 1\sum \beta mXt - mMm = 1 \tag{20}$$

where xt represents exogenous variables, β their coefficients, φ(L)yt is an AR model (φ1yt – 1 + φ2yt – 2 + φ3yt – 3 + . . . + φpyt – p) and θ(L)εt is the MA model (θ1εt – 1 + θ2εt – 2 + θ3εt – 3 + . . . + θqεt – q).

Both transfer function order for predictors and ARIMA order for the dependent variable are used in multivariate ARIMA models. By using past values, these models are utilized to estimate future values. With or without the use of predictors, the ARIMA model can estimate future values. The ARIMAX model is what is created when independent variables or predictors are added to the ARIMA model [10]. The following is the mathematical function for the ARIMAX model:

$$Z_1 = \sum_{i=1}^{3} W_{1,i}x_i - \mu_1 and Z_2 = \sum_{i=1}^{3} W_{2,i}x_i - \mu_2 \tag{21}$$

where, φ1,…,φp and θ1,…,θq are the parameters; εt, εt-1 are white noise error and β1,…,βm are the parameters of independent variables input Xt and t is the time. The three processes involved in setting up the ARIMAX model are identification, parameter estimation and selection, and diagnostic checks. Identification requires deciding whether the data are stationary, parameter estimation and selection involve choosing the relative weights of AR and MA, and diagnostic check involves confirming that there is no residual autocorrelation [10].

Williams (2001) [10] used the ARIMAX model to estimate the short-term freeway traffic flow. The research found that ARIMAX models outperform univariate forecast models in terms of forecast accuracy. However, it was advised that before broad usage of ARIMAX models, many drawbacks needed to be resolved. According to this study, the ARIMAX model assumes constant transfer function parameters whereas the correlation between upstream and downstream data changes depending on the current traffic situation, particularly the speed of the traffic stream. Yang et al. (2017) [24] attempted to explain why the ARIMAX model is better than the conventional ARIMA method in the context of traffic flow prediction. Finally, by lowering the MAPE value by 2.855 percent, the ARIMAX model demonstrated its superiority over the ARIMA model in capturing significant changes in traffic flow. Table (1) below summarizes all of the statistical models that have been discussed thus far:

3- Nonparametric Models

When referring to distributional models and hypothesis tests, the term “nonparametric” describes the number of parameters that were used to create the distribution. It is crucial to note that “parametric” approaches have a distribution with a fixed number of parameters, but “nonparametric” methods do not. They frequently use milder assumptions like symmetry or continuity [25]. In the following, neural networks (NNs), support vector machines (SVM), support vector regression (SVR), and k-nearest neighbor (KNN) regression are thoroughly investigated as non-parametric models.

3- 1- Neural Networks

Multilayer feedforward neural networks (MLFNN), time-delay neural networks (TDNN), recurrent neural networks (RNN), convolution neural networks (CNN), deep belief networks (DBN), radial basis function neural networks (RBFNN), wavelet neural networks (WNN), and fuzzy neural networks (FNN) are the most common types of NN models used for traffic state prediction in the literature. Each of the NN kinds listed above is discussed in greater depth in this section.

3- 1- 1- Multilayer feed-forward neural network

(input layer and an output layer of perceptrons. Technically, this is referred to as a one-layer feedforward network with two outputs because the output layer is the only layer with an activation calculation.

n this single-layer feedforward neural network, the network's inputs are directly connected to the output layer perceptrons, Z_1 and Z_2.

The output perceptrons use activation functions, $g_{1,}$ and g_2, to produce the outputs Y_1 and Y_2.

Since;

$$Y_1 = g_1(Z_1) = g_1(\sum_{i=1}^{3} w_{1,i} x_i - \mu_1) \tag{22}$$

$$Y_2 = g_2(Z_2) = g\,2(\sum_{i=1}^{3} w_{2,i} x_i - \mu_2) \tag{23}$$

And

$$\phi_i(\| x - c_i \|) = exp(-\frac{\| x - c_i \|^2}{(2\sigma_j^2)}) \tag{24}$$

When the activation functions g_1 and g_2 are identity activation functions, the single-layer neural net is equivalent to a linear regression model. Similarly, if g_1 and g_2 are logistic activation functions, then the single-layer neural net is equivalent to logistic regression. Because of this correspondence between single-layer neural networks and linear and logistic regression, single-layer neural networks are rarely used in place of linear and logistic regression.

MLFNN is a basic feedforward neural network with one input layer, one output layer, and one or more hidden layers. There are several nodes, also known as neurons or units, in each layer. Weighted links connect the nodes, which contribute to the transmission of information from one layer to the next. MLFNN is the most often utilized type of NN so far. With the adaption of nonlinear activation functions and the training of weights in a multilayer framework, MLFNN may approximate nonlinear functions. In the literature, backpropagation is the most frequent technique for training MLFNNs, and these models are commonly referred to as backpropagation neural networks (BPNN) [4, 26].

A three-layer feed-forward neural network model has been constructed to replicate an upgraded version of a well-known higher-order continuum traffic model proposed by Zhang et al. in 1997 [27] for traffic system identification. The neural network model captures the traffic dynamics of this model fairly well, according to simulation data. Polson et al. (2017) [28] proposed using a regularized linear vector autoregressive model to discover spatial-temporal connections in data to perform variable selection. The traffic speed was then predicted using a NN with a stack of hidden layers. The MSE achieved by the traditional MLFNN with one hidden layer was 14 percent higher than the deep model. Elhenawy et al. (2017) [29] proposed a method in which the network was first trained using the backpropagation algorithm with only one hidden layer. The output layer was then preceded by a new hidden layer. The model was then retrained, starting with the trained weights between the input layer and the first hidden layer. This method was continued until the desired number of hidden layers had been achieved. The entire NN was then fine-tuned to complete the training process. In addition, principal component analysis (PCA) was utilized to minimize input dimensionality, and the divide-and-conquer strategy was suggested for dealing with vast amounts of data in this work. When estimating speed and flow, the model with three hidden layers had mean absolute percentage errors (MAPEs) of 2.8 and 8%, respectively.

Because of its capacity to describe nonlinear functions using basic structures, MLFNNs have been widely employed in traffic state prediction. Shallow NN models have been demonstrated to be inferior to MLFNNs with numerous hidden layers. These models, however, may not be sufficient to exploit more complicated correlations in traffic data and achieve higher accuracy [30, 31].

3- 1- 2- Radial basis function neural networks

(RBF network in its simplest form is a three-layer feedforward neural network. The first layer corresponds to the inputs of the network, the second is a hidden layer consisting of several RBF non-linear activation units, and the last one corresponds to the final output of the network. Activation functions in RBFNs are conventionally implemented as Gaussian functions. To illustrate the working flow of the RBFN, suppose we have a data set *D* which has *N* patterns of (xp,yp) where xp is the input of the data set and yp is the actual output. The output of the *i*th activation function ϕi in the hidden layer of the network can be calculated using Eq. (…) based on the distance between the input pattern *x* and the center *i*.

$$Pxh^1 ... h^l = \prod_{k=0}^{l-2} Ph^k h^{k+1} Ph^{l-1)} h^l \tag{25}$$

Here, $\|.\|$ is the Euclidean norm, cj, and σj are the center and width of the hidden neuron *j*, respectively.

Then, the output of the node *k* of the output layer of the network can be calculated using Eq. (28.2):

$$y_i = fWgx = \sum_{j=1}^{n} w_i jgm \sum_{k=1} w_{jk} x_k + w_{j0} + w_{i0} \tag{26}$$

Most of the classical approaches deployed in the literature for training RBFNs are performed in two stages. In the first stage, the centers and widths are determined using for example some unsupervised clustering algorithm, while in the second stage, the connection weights between the hidden layer and the output layer are found in a way such as an error criterion like the common Mean Squared Error (MSE) is minimized over all the data set.

RBFNN models consist of three layers: an input layer, a hidden layer with radial basis functions (RBF) as activation functions, and a linear output layer. RBFs are real-valued functions that produce outputs depending solely on the distance between the origin and the inputs, often the Euclidean distance. The Gaussian function is commonly employed for this network among RBFs. RBFNN training usually consists of two steps: unsupervised learning and supervised learning. The parameters of the basis function, such as center and width (or standard deviation), are specified separately for each hidden unit from the input variables in the first phase. This stage can be completed by assigning the input samples to distinct partitions using unsupervised techniques such as clustering algorithms. The number of clusters is a design parameter that correlates to the hidden layer's nodes. The centroids are utilized as centers in the clustering, and widths can be determined using the distance between the samples and the centroids. The weights between the hidden layer and the output layer are calculated in the supervised learning stage using the hidden layer's outputs and the expected outputs. Fitting a linear model to an objective function like the least squares is how training is done. RBFNN has been used in a variety of time-series applications due to its generalization ability and quick training speed [32, 33].

Zhu et al. 2014 [32] investigated traffic flows at adjacent intersections using a radial basis function neural network. Artificial Neural Networks (ANN) could be a suitable answer to this problem because they allow for improved prediction accuracy in a short amount of time. When forecasting the one in the middle, this research proposes a novel strategy that takes into account the travel flows of the adjacent intersections. The suggested paradigm is both effective and feasible, according to computational testing. However, using traffic data from nearby crossings to provide more accurate predicting results might be investigated further. In the year of 2019, Li et al. [34] utilized a new dynamic radial basis function (RBF) neural network to explore short-term passenger flow prediction. Passenger flow control is considered to improve the prediction accuracy by adding passenger flow control coefficients to the model. the proposed method achieves the best prediction performance at a half-hour prediction time lag. The proposed method can also identify crucial stations and periods 30 min in advance, which contributes when considering proactive passenger flow control to alleviate congestion during peak hours in metro networks. All in all, this study predicts passenger flow only in scenarios that include passenger flow control, large passenger flow volume, and normal passenger flow. In addition, a better way to determine the center of the RBF neural network in future studies can be considered. Jun et al. 2014 [35], used the ant colony optimization (ACO) algorithm to optimize the parameters of the RBF neural network for network traffic prediction. The disadvantages of the traditional radial basis function (RBF) neural network during the network traffic prediction process, such as a slow convergence rate and easy occurrence of local optima, result in low prediction precision. The experimental results show that compared with the genetic algorithm (GA)-RBF and particle swarm optimization (PSO)-RBF traffic prediction models, the proposed model exhibits higher prediction accuracy and can describe the varying trends in the network traffic well. The model used in this study exhibits strong generalization ability and good stability and therefore has practical value in network traffic prediction. Chen *et al.* 2017 [36] propose an optimized prediction algorithm for radial basis function neural networks based on an improved artificial bee colony (ABC) algorithm in the big data environment. To verify the efficiency of this algorithm in the big data environment, apply it to Lozi and Tent chaotic time series and measured traffic flow time series, and then compare it with K-nearest neighbor (KNN) model, radial basis function (RBF) neural network model, improved back propagation (IBP) neural network model and radial basis function neural network based on cloud genetic algorithm (CARBF) model. The experimental results indicate the effectiveness of the proposed algorithm in predicting traffic flow time series.

The weights of the connections between the hidden layer and the output layer may be trained in one step, making RBFNNs simple to train. As a result, they are typically trained faster than MLFNNs. RBF networks provide superior generalization and are more resilient to adversarial or noise examples than typical MLFNNs due to the inclusion of an RBF as the activation function in the hidden layer. However, the RBFNNs' performance is mostly determined by the settings chosen for the RBFs, which is not an easy matter [4, 37].

3- 1- 3- Wavelet neural networks

A WNN is a method that combines wavelet theory and artificial neural networks. It is basically an MLFNN model with a wavelet function in place of a classic activation function like sigmoid or tanh in the hidden layers. A wavelet function is a function that represents the signal's evolution in both the time and frequency domains and has an average value of zero. To approximate intricate patterns, this model uses the wavelet transform's multiscale analysis capabilities and NN's self-learning capability. Numerous applications of WNN models have been effective, including time series prediction and signal processing [9, 38].

Li & Sheng 2015 [39] proposed an improved wavelet neural network model based on a modified particle swarm optimization algorithm to prove the availability of the modified prediction method by predicting the time series of real traffic flow. At last, the computer simulations have shown that the nonlinear fitting and accuracy of the modified prediction methods are better than other prediction methods. Following optimization approaches, Chen et al. 2021 [40] introduced an improved wavelet neural network. The improved wavelet neural network is used to predict short-term traffic flow. The experimental results show that the proposed algorithm is more efficient than the WNN and PSO–WNN algorithms alone. The prediction results are more stable and more accurate. Compared with the traditional wavelet neural network, the error is reduced by 14.994%. Through the algorithm simulation comparison, the proposed algorithm has higher prediction accuracy, but because the number of training model samples in the optimized neural network is small, and the urban traffic flow prediction is easily affected by the external environment and emergency events, the prediction effect still exists. Where there are uncertainties and deficiencies, there are still many issues that require further study. Chen et al. 2018 [41] proposed a fuzzy wavelet neural network (FWNN) trained by improved biogeography-based optimization (BBO) algorithm for forecasting short-term traffic flow using past traffic data. The performance indexes show that the FWNN model achieves lower root-mean-squared error (RMSE) and mean absolute percentage error (MAPE), as well as a higher correlation coefficient *(R)*, indicating that the FWNN model is a better predictor. As for future research, it may be desirable to apply the proposed model to evaluate more traffic flow data from different locations.

WNNs, like RBFNNs, take less time to train and have better generalization than MLFNNs. Furthermore, WNNs' parameters are easier to determine than RBFNNs', and WNNs can reach quick converging. WNN could well be ideal for time-series applications, such as traffic state prediction, and that to wavelet functions' ability to cope with time and frequency aspects [4, 42].

3- 1- 4- Time-delay neural networks

TDNN models are multilayer NNs that use a time-shifting method to use delayed inputs or states. The models can now reflect the temporal dynamics of time-series data. The neurons in the hidden layer in the model illustrated in the figure receive the input value at time t, as well as the input values at time t1 and t2. Delays can also be applied to the model's other layers. TDNN uses the internal states or outputs created in earlier time steps when delays are applied to neurons in layers other than the input layer. The model in these circumstances includes feedback connections and might be categorized as an RNN. In the papers, however, TDNN is commonly referred to as an MLFNN that applies delays to entries [42-44]. In this section, we will explore these methods.

Abdulhai *et al.* 2002 [45] presented a short-term traffic flow prediction system based on an advanced Time Delay Neural Network (TDNN) model, the structure of which is synthesized using a Genetic Algorithm (GA). The model's performance is validated using both simulated and real traffic flow data. The model performed acceptably using both simulated and real data. The model also showed the potential to be superior to other well-known neural network models as the multi-layer Feed-forward (MLF) when applied to the same problem. Lingras et al. 2001 [46] Proposed a TDNN model using GA to choose the connections between the time-delayed input layer and hidden layers. The objective of the optimization was to maximize the linear correlation between the input variables and the output variable. The highest correlated subsets of input variables were then used to design the structure of the TDNN. In the experiment, the data set was analyzed using three types of states depending on the traffic. Zhong *et al.* 2005 [47] offers genetic algorithms that were used to design time delay neural network (TDNN) models as well as locally weighted regression models to predict short-term traffic for rural roads. Refined TDNN models developed in this study can limit most average errors to less than 10% for all study roads. Refined regression models show even higher accuracy. Average errors for the refined regression models are less than 2% for roads with stable patterns. Even for roads with unstable patterns, average errors are below 4%, and the 95th percentile errors are less than 7%.

In a feedforward model, using TDNNs is a straightforward technique to describe the correlations between past and present values. TDNN delays are calculated and addressed throughout the training process. In comparison to RNN models, which are discussed in the next part, TDNNs need less computation and are easier to train. On the other hand, these set delays may not be enough to capture temporal dynamics, which may be dynamic and alter through time [44].

3- 1- 5- Recurrent neural networks

3- 1- 5- 1 Standard RNNs

RNNs are neural networks that are especially good at coping with time sequences. RNNs, unlike feedforward NNs, have feedback connections that feed the previous phase's states or outputs to the following step. The most well-known RNN is an MLFNN, which is sometimes referred to as an Elman network since the hidden states are sent to the next stage together with the inputs to the hidden layer. The main memory is represented by the hidden state in this paradigm. Jordan networks are another sort of RNN that contains output values that are transmitted back to the network's hidden layer in the next time step [48, 49].

Ishak et al. (2003) [50] describe a method for improving short-term traffic prediction performance by combining numerous topologies of dynamic neural networks, the Elman network, partial RNN, and TDNN, with a variety of network- and traffic-related variables. A statistical nonlinear time series technique was compared to the optimum performance of the dynamic neural networks, which was exceeded in most circumstances. For all prediction horizons studied, no single topology consistently outperformed the others. Dia (2001) [51] investigated the problem using several Elman networks, including completely connected and partially connected RNNs, as well as TDRNs. The results show that utilizing an object-oriented approach for short-term traffic prediction is feasible, with significant gains over traditional model performance.

3- 1- 5 -2 Long short-term NNs

The long short-term memory (LSTM) model was proposed to avoid the vanishing gradient problem in typical RNNs, which hinders them from detecting long-term interconnections. This model uses a gating mechanism to determine when and how it should refresh its memory. A cell, an input gate, a forget gate, and an output gate make up an LSTM unit (or block). The memory is represented by the cell, and the input gate defines what fresh information should be placed in it, the forget gate indicates what information should be erased from it, and the output gate demonstrates how the memory is utilized to estimate the LSTM unit's output [52, 53].

Ma et al. 2015 [54] utilized LSTM NN to predict traffic speed. A comparison with different topologies of dynamic neural networks as well as other prevailing parametric and nonparametric algorithms suggests that LSTM NN can achieve the best prediction performance in terms of both accuracy and stability. Zhao et al. 2017 [55] proposed a novel traffic forecast model based on long short-term memory (LSTM) network. Different from conventional forecast models, the proposed LSTM network considers temporal–spatial correlation in traffic systems via a two-dimensional network that is composed of many memory units. A comparison with other representative forecast models validates that the proposed LSTM network can achieve a better performance. According to the comparison, the performance of the proposed LSTM network is better than SAE, RBF, SVM, and ARIMA model, especially when the forecast time is long. Cui et al. 2018 [56] suggested a deep stacked bidirectional and unidirectional LSTM (SBU- LSTM) neural network architecture is proposed, which considers both forward and backward dependencies in time series data, to predict network-wide traffic speed. A bidirectional LSTM (BDLSM) layer is exploited to capture spatial features and bidirectional temporal dependencies from historical data. To the best of our knowledge, this is the first time that BDLSTMs have been applied as building blocks for a deep architecture model to measure the backward dependency of traffic data for prediction. The proposed model can handle missing values in input data by using a masking mechanism. Further, this scalable model can predict traffic speed for both freeway and complex urban traffic networks. Comparisons with other classical and state-of-the-art models indicate that the proposed SBU-LSTM neural network achieves superior prediction performance for the whole traffic network in both accuracy and robustness. In another study, Abduljabbar *et al.* 2021 [57] adopts the Long Short-Term Memory (LSTM) recurrent neural network to predict speed by considering both the spatial and temporal characteristics of real-time sensor data. These results demonstrate the superior performance of LSTM models in capturing the spatial and temporal traffic dynamics, providing decision-makers with robust models to plan and manage transport facilities more effectively. Future work will explore incorporating more inputs such as flow, occupancy as well as other important external factors such as weather to further refine the prediction models.

The LSTM NN overcomes the problem of back-propagated error decay by using memory blocks, demonstrating greater capabilities for time series prediction with extended temporal dependencies. The LSTM NN can also automatically calculate the best time delays [58].

3- 1- 5 -2 Gated recurrent unit NNs

Cho et al. proposed the gated recurrent unit (GRU), a variant of the LSTM model (2014). Although LSTM and GRU networks perform similarly in many applications, GRU networks have fewer parameters and are thus faster to train. An update gate and a reset gate are both present in a GRU unit. The update gate controls how much of the prior state is kept. How the new input is fused with the existing state is determined by the reset gate.

A GRU-based network was first applied to traffic flow prediction by Fu, Zhang, and Li (2016) [59]. In this study, both LSTM and GRU networks were implemented. Experiment results showed that the two models performed better than ARIMA, while the GRU network's performance was slightly better than the LSTM network. The MAE achieved by GRU NN was lower than around 10 and 5% on average compared with that of ARIMA and LSTM NNs, respectively. Guo et al. 2017 [60] proposed GRU neural network and autocorrelation analysis for multi-step prediction. The model dynamically updates the network with the input of the measured real-time data, namely on-line prediction, to work effectively and constantly. Through the theoretical derivation and simulation analysis, it is shown that the prediction accuracy of the proposed 1GRU prediction model is improved. The model can be used as an effective method for multi-step traffic prediction.

3- 1- 6- Convolutional neural networks

In the field of machine learning, a CNN is a sort of NN that has been effectively implemented. A CNN design contains hidden layers like convolution layers, pooling layers, and fully connected (fc) layers in addition to input and output layers. To produce the outcome features, a convolution layer convolves the input features with distinct kernels (or filters), which are sets of weights with quantities learned forward through the training phase. The input to this layer is subsampled by a pooling layer, which is generally utilized after a convolution layer. Implementing a maximum operator on the neighboring values in the matrix to produce a smaller matrix containing one of the most important characteristics is one technique to accomplish this pooling. In classic feedforward NNs, one layer is entirely related to the next layer, whereas, in CNN, the convolution layers are linked locally by sliding filters. With this local connection, output neurons are solely coupled to input neurons in their immediate vicinity, allowing the model to reflect the input matrices' local spatial properties. The CNN model shown in the diagram is intended for binary image classification.

The CNN model was used by Song *et al.* in 2017 [61] to forecast traffic speed. The prediction performance of the CNN-based model is higher than the prediction performance of the other two multi-layer perceptron (MLP) models, according to the comparative results. In the other study, Willis *et al.* 2017 [62] used the CNN approach to achieve 95 percent accuracy on held-out test samples after collecting data from traffic cameras. An improved convolutional neural network (CNN) with asymmetric kernels was proposed by Zang *et al.* in 2017 [63]. This study's method was compared to some traditional traffic speed estimating methods. The results of the experiments show that the technique outperforms all other approaches. Zhang et al. 2019 [64] proposed CNN method outperforms baseline models in terms of accuracy. To verify the effectiveness of the proposed method, it was compared with six baseline algorithms. Using traffic flow data, the proposed model achieved good prediction performance with different forecasting intervals. To estimate traffic speed, Liu et al. 2018 [65] suggest attention CNN. Three-dimensional data matrices based on traffic flow, speed, and occupancy are used in the model. The convolution unit handles the extraction of spatial–temporal features as well as the attention models. Experiments using traffic data at 15-minute intervals show that the proposed method outperforms other commonly used algorithms in forecasting tasks and that the developed model improves in circumstances where data is missing.

With using fully linked layers in classical MLFNNs, no assumptions about the input are made other than the premise that each component might be beneficial to the objective. In such approaches, the model creator must normally extract or choose relevant features for the specified task to be used as inputs using specific methodologies [66]. This phase may be impracticable for complex tasks where identifying relevant attributes is challenging. Given that the input can be expressed as a tensor with locally linked elements, CNNs are able to obtain such characteristics automatically (e.g., images, videos, audio data, etc.) [67]. Convolution operations could be used to achieve this. CNNs have lately been used to extract traffic correlations in traffic state prediction using these capacities. CNNs use fewer operations than typical MLFNNs for the same number of neurons because of their limited connection and shared weights. However, because CNNs often include multiple convolutional layers and a large number of filters for learning features at various levels, the computational cost of CNNs is still considerable [68, 69].

3- 1- 7- Deep belief networks

(DBN model the joint distribution between observed vector x and the l hidden layers h^k is as follows:

$$Pxh^{1}...h^{l}=\prod_{k=0}^{l-2}Ph^{k}h^{k+1}Ph^{l-1)}h^{l} \tag{27}$$

where x = ho, $P(h^k \mid h^{k+`})$is a conditional distribution for the visible units conditioned on the hidden units of the RBM at level k, and $P(h^{l-`}, h^l)$is the visible-hidden joint distribution in the top-level RBM. A deep learning model made up of numerous layers of restricted Boltzmann machines is known as a DBN (RBMs). RBMs are two-layer stochastic NNs that can learn a probability distribution over the input data and have undirected links between the layers. The visible layer (or input layer) of an RBM is the first layer, while the hidden layer is the second. The hidden layer outputs in an RBM can then be used to recreate the entries. The output of one RBM is used as the input of the following RBM in DBN. A greedy unsupervised learning technique is used to extract feature vectors from the stack of RBMs, which can then be used for classification or prediction tasks. A regression output layer is placed on top of the RBMs to do these tasks, and the DBN could perhaps be trained using supervised learning techniques [4, 44].

For the first time, Huang et al. 2013 [70] used a deep-learning approach in transportation research. A deep architecture is presented in this paper, which comprises two parts: a Deep Belief Network at the bottom and a regression layer at the top. For unsupervised feature learning, the Deep Belief Network was used. Experiments reveal that the method can generate outcomes that are nearly 3% better than the state-of-the-art. For traffic flow forecasting, Hong et al. 2014 [71] suggest a deep neural network. The metric-based multi-task neural network outperforms the Euclidean-based multi-task neural network in traffic flow predictions on a real dataset, according to the final results. Zhao et al. 2019 [72] propose a master-slave parallel computing structure in which slave computing nodes learn the properties of their sub-datasets and send them to the master computing node. The proposed parallel computing method is used to estimate traffic flow using real-world traffic data. The experimental results show that the parallel computing method of DBN learning processes is successful in terms of reducing pre-training and fine-tuning times while preserving significant feature learning abilities. Kong et al. 2019 [73] investigate a deep learning-based strategy for constructing random neural networks for predicting short-term traffic flow that can assist in the development of a smart multimedia system for the Internet of Vehicles (IoVs). The RBM technique provides a high level of resilience and adaptability, laying a solid platform for future traffic signal control research. As compared to other approaches, the RBM model has superior nonlinear fitting ability and higher prediction accuracy for typical chaotic time series as the number of training samples increases. For feature extraction and performance comparison, Tan et al. 2016 [74] constructed two deep-learning-based traffic flow prediction models: The first is a DBN based on restricted Boltzmann machines (RBMs) with Gaussian visible units and binary hidden units, and the second is a DBN based on RBMs with all units binary. After a series of studies, it is concluded that the earlier performs better in traffic flow prediction.

RBMs are pre-trained in a DBN using a greedy unsupervised learning method before being progressively trained and supervised to perfect the weights. When compared to typical DNNs with the same number of layers, our training approach enables DBNs to be trained rapidly. The methods can derive significant features from the input by using RBMs, which consider the structure and distribution of the data. These qualities improve in model generalization and alleviate the problem of overfitting [75, 76].

3- 1- 8- Fuzzy neural networks

FNNs, also known as neuro-fuzzy systems, are machine learning models that combine the properties of fuzzy systems and neural networks. Using membership functions of fuzzy sets, a fuzzy system converts the input data to real values between zero and one. The mapped values are then subjected to fuzzy rules, which result in output values. The customizable estimation feature of NNs is used to extract fuzzy parameters (i.e., fuzzy sets and fuzzy rules) in these models. Integrating rule-based systems with NNs can help them be more interpretable. The input layer, the hidden layer containing the fuzzy rules, and the output layer make up a three-layer feedforward NN. The fuzzy sets are represented by the connection weights between the layers in this model [77, 78].

The fuzzy neural networks (FNN) may be defined by the function:

$$y_i = fWgx = \sum_{j=1}^{n} w_i jgm \sum_{k=1} w_{jk} x_k + w_{j0} + w_{i0} \tag{28}$$

Where, W = weight vector of weights connecting jth hidden unit to inputs and ith output to hidden units, $g(\cdot)$ = sigmoidal transfer function, and x_k = inputs.

Chan et al. (2014) [79] present an orthogonal array-based systematic and successful experimental design method for determining optimum on-road sensor configurations for Takagi-Sugeno models TS-models. The created TS-model can produce reliable traffic flow forecasting, according to the results. Pang et al. (2008) [80] introduce a fuzzy neural network predicting model that can be easily trained online. In addition, the clustering radius was calculated using a genetic method. The simulation's outcome demonstrates its accuracy and applicability. Deshpande et al. (2016) [81] propose using a neuro-fuzzy hybrid system for short-term traffic flow prediction, which combines the complementing skills of both neural networks and fuzzy logic to increase forecast accuracy. The employment of a hybrid system boosts the performance measure to a suitable level. Li et al. (2016) [82] explore traffic flow prediction using dynamic fuzzy neural networks (D-FNN). In reality, this approach can automatically create proper network structure. By combining chaos theory with the neural network technique, the approach is used to the traffic flow time series to examine and compare the forecasting performance of the predicting model based on the neural network method and the adaptive neural fuzzy inference system. The simulation results demonstrate that this strategy is quite effective and can increase the accuracy of prediction.

FNNs consist of a combination of fuzzy systems and neural networks. It is not easy to identify membership functions and appropriate fuzzy rules for a fuzzy system. By utilizing the customizable estimation feature of NNs, FNNs are capable of learning these elements of fuzzy systems. Furthermore, NN-based models are notorious for their lack of understandability. The superior interpretability of fuzzy systems could offset this drawback in FNN models. Tang et al. (2017) [83] proposed a novel method for generating fuzzy neural networks to anticipate travel speed for multi-step ahead using 2-min travel speed data obtained from three distant traffic microwave sensors. To complete the fuzzy inference, the first-order Takagi–Sugeno system is used. Two learning procedures are presented to train an evolving fuzzy neural network (EFNN). To assess the membership degree of samples to the cluster centers, a Gaussian fuzzy membership function is created for each cluster. Because of their excellent learning ability, the results imply that EFNN's prediction performances are superior to classical models.

FNNs combine the merits of fuzzy systems and NNs. Finding membership functions and appropriate fuzzy rules of a fuzzy system is not an easy task. FNNs are capable of learning these elements of fuzzy systems by engaging the adaptive approximation ability of NNs [84]. Additionally, NN-based models are known for their low interpretability. In FNN models, this disadvantage could be compensated by the good interpretability of fuzzy systems [85].

3- 1- 9- Drawbacks of NNs

NNs have various disadvantages that may limit their capacity to generalize for traffic status forecasting, despite their great nonlinear fitting capabilities. The limited interpretability or black-box aspect of NNs is one of their most significant drawbacks. To put it another way, the behavior of the models employed to generate prediction outcomes is a mystery [86]. Regardless of how well they predict, there is no obvious causation analysis of traffic mechanisms, which would be useful in research studies. In addition, to ensure statistical precision, NN-based models typically require huge data sets, especially for systems with several layers. Moreover, while trying to predict traffic situation, NN-based models use more computer resources than statistical model structures [4]. The amount of training data and the complexity of NN structures determine the computational cost of NNs for training the network. Deep neural networks (DNNs) for large-scale traffic networks can take several weeks to train. Picking the best structure of NNs is also an issue when implementing NN models. Many design factors must be tweaked to reach the optimum performance for a given road network or data collection (for example, the number of hidden layers, hidden nodes, learning rate, kind of activation functions, etc) [87]. Table (2) below summarizes all of the machine learning models that have been discussed thus far:

3- 2- Support vector machine/Support vector regression

3- 2- 1- Support vector machine

The method called Support Vector Machine (SVM) is used to categorize both linear and non-linear data. The "Hyperplane" plane is used to make a division between two classes. The main goal is to find the best hyperplane for classifying data points. The case of linear SVM exists where the data points can be categorized using a straightforward linear hyperplane. But if a linear classifier is unable to separate the data points into classes, a transformation function is first applied to the dataset to move the data points belonging to one class to different coordinates or a higher dimension, and then the data points are separated using the same linear classifier. Support vectors and margins are used to create the hyperplane for separating data points. Some chosen data points that are on the border of either class are known as support vectors. Margins are created along the support vectors using these support vectors. In addition, "Gutter Space" refers to the distance between the data points [88-90].

Support vectors are input vectors that barely touch the margin's border. The fictional lines drawn with the aid of support vectors are known as margins. It is vital to note that SVM uses two different distances, d+ and d-. On the separating hyperplane, d+ is on the right-hand side (R.H.S.), and d- is on the left-hand side (L.H.S.). Maximizing margins is the primary goal of SVM [91, 92].

Support vector machines (SVM), a machine learning technique, were explored by Vanajakshi and Rilett (2007) [93] for the short-term prediction of travel time. According to the findings, SVM can be a good substitute for traditional methods for solving short-term prediction issues, especially when the data is sparse or noisy. Theja and Vanajakshi (2010) [94] employed SVM to forecast a variety of traffic characteristics over the short term, including speed, volume, density, travel time, headways, etc. It is important to note that this study was among the first to comprehensively look into the use of SVM in this area. A short-term traffic condition prediction technique for urban road networks was presented by Yan et al. (2017) [95] and is based on enhanced support vector machines. Since outliers will always be present in traffic data gathered, the strategy utilized in this paper could efficiently deal with their adverse impacts, strengthening the model's robustness and enhancing generalization ability. The usefulness method is further supported by experimental findings that indicate the enhanced approach has higher classification accuracy than other machine learning approaches.

Based on the Least Squares Support Vector Machine (LS-SVM), Chen *et al.* (2011) [96] introduced a new transit flow forecast model. The acquired result demonstrated that there was little variation between the actual value and the prediction, and the majority of the equal coefficients of a training set are larger than 0.90, demonstrating the validity of the method. Wang (2020) [97] studied the traffic flow prediction capabilities of the bat algorithm support vector machine (BA-SVM). Using a small sample of data, SVM can achieve a balance between model complexity and learning capacity to provide the optimum generalization capability. The suggested prediction model is qualified, and the fitting accuracy was high, the prediction impact was good, and it could be utilized to predict short-term traffic flow, according to empirical analysis and comparison of the short-term traffic flow observed at a non-detector intersection.

3- 2- 2- Support vector regression

Based on the idea of Vapnik's support vectors, Drucker *et al.* (1996) [98] first presented support vector regression (SVR) as a supervised learning algorithm. By identifying the hyperplane and lowering the difference between the anticipated and observed values, SVR seeks to minimize the error.

A support vector regression-based short-term traffic forecasting model was presented and investigated by Gong et al. (2013) [99]. In this study, the traffic volumes at the current period of time were regarded as the output, while the traffic levels at earlier periods of time, as well as upstream and downstream, were considered the inputs. Castro-Neto et al. (2009) [100] proposed the use of an Online Support Vector Machine for Regression, or OL-SVR, supervised statistical learning technique for the prediction of short-term highway traffic flow under both normal and exceptional circumstances. Three well-known prediction models, including the Gaussian maximum likelihood (GML), the Holt exponential smoothing, and artificial neural network models, were compared with the OL-SVR model. The best performance under non-repeating unusual traffic situations, according to the results, was OL-SVR. In a different study, an adaptive SVR model for predicting short-term traffic was put out by Wei and Liu (2013) [101]. As heuristic information, the time-varying deviation of the daily traffic variation, which is expressed in a bilevel formula, is incorporated into SVR to create an adaptive -margin that takes into account both local and normalized parameters. Field traffic data comparison experiments show that the suggested model routinely beats the traditional SVR with higher computational efficiency. To increase the time efficiency of forecasting, Zeng et al. (2008) [102] introduced a new short-term traffic flow prediction model and approach based on accurate online support vector regression (AOSVR), which can update the prediction function in real-time via incremental learning. Results showed that the AOSVR predictor may greatly lower projected trip time relative mean errors and root mean squared errors. As a result, AOSVR-based traffic flow prediction is useful and effective for analyzing traffic data.

One of SVR's key benefits is that its computational complexity is independent of the input space's dimensionality. Additionally, it has a strong capacity for adaptation and improved prediction accuracy [49].

3- 3- K-nearest neighbor regression

When there is little or no prior knowledge about the distribution of the data, K-nearest-neighbor (KNN) classification should be one of the first available approaches. It is one of the most basic classification techniques. The necessity to perform discriminant analysis when accurate parametric estimates of probability densities are unknown or challenging to ascertain led to the development of K-nearest-neighbor classification [103].

To predict short-term traffic flow, Zhang et al. (2009) [104] suggested a KNN-NPR (K-nearest neighbors non-parametric regression) method based on balanced binary trees. Clustering techniques and balanced binary trees were used to develop the case database To increase forecasting accuracy and meet real-time objectives. Mladenovic et al. (2022) [105] looked at the k-nearest technique for predicting nighttime traffic flow. The model built using the K-Nearest Neighbors algorithm has, in the end, produced the best results. The total and average monthly nighttime traffic for the upcoming year at the chosen traffic counting stations has been predicted using this model. In a study on short-term traffic volume forecasting using an improved k-nearest neighbor approach, Zheng and Su (2014) [106] did the research. Comparison analysis demonstrates that it frequently surpassed the other algorithms. For the forecasting of short-term traffic flow, Zhao et al. (2020) [107] suggested a new intelligent parameter adjustment k-nearest neighbor method (IPA-KNN). The data experiments demonstrate that the error is reduced by more than 23% when evaluated with the modified KNN approach, which was given by Habtemichael and Cetin (2016) [108]. Without using predetermined models or training for the parameters, Sun et al. (2018) [109] provide a completely automatic dynamic process kNN (DP-kNN) that makes the kNN parameters robust and self-adjustable. The findings indicate that, in regards to accuracy on average, the DP-kNN can outperform the manually adjusted kNN and other benchmarking techniques. Hourly average traffic speed and hourly volume are two traffic characteristics that Rasaizadi et al. (2021) [110] identified as indicators of traffic status. The results show that the K-nearest neighbor, which is equivalent to 61 and 91 percent, respectively, achieves the maximum accuracy. The artificial neural network and K-nearest neighbor models have significantly increased prediction accuracy when using the historical average as a benchmark model.

KNN has a number of benefits, which are why many academics have used it in their studies. First of all, it is easy to use and simple to comprehend. When used for classification and regression, it can learn non-linear decision boundaries. To modify K's value To create a very flexible decision boundary. Additionally, there is no explicit training phase in the KNN algorithm; all work is done during prediction. It is important to note that because there is no explicit training stage, the prediction is changed as newer data is continuously added to the dataset without the need to retrain the model. There are numerous distance measurements available. Euclidean, Manhattan, Minkowski, hamming distance, and other common distance metrics are utilized [111, 112].

4- Conclusion

While there is a substantial quantity of literature on short-term traffic status prediction, there have been numerous studies done in this field recently. Parametric models are those that require the specification of some parameters before they can be used to make predictions. In contrast to “nonparametric” procedures, “parametric” approaches have a distribution with a defined number of parameters. They usually rely on softer presumptions like continuity or symmetry. In the research, it has been shown that NN-based models in general and deep architectures, in particular, in particular can handle the nonlinear correlations and dynamics in traffic data. Different NN models have been employed to forecast various traffic factors with better precision by utilizing a variety of NN types with varying capacities in modeling nonlinearities.

The limitation of short-term traffic prediction is that it can only provide accurate forecasts for a limited period of time, typically up to a few hours. Beyond this time horizon, the accuracy of the predictions tends to decrease due to the increasing uncertainty and variability of traffic patterns. Additionally, short-term traffic prediction models may not account for unexpected events such as accidents or road closures, which can significantly impact traffic flow and result in inaccurate forecasts.

5- Future Researches

The applications of these methodologies can be further enhanced to make them more adaptable and efficient in the future thanks to the rich advancement of parametric and nonparametric methods.

This review points to several potential future research directions, including:

Recent research has focused a lot of emphasis on hybrid strategies, which can combine parametric-parametric, parametric-nonparametric, and nonparametric-nonparametric approaches. Thus, emphasizing hybrid approaches will contribute to the literature in a number of positive ways.

To improve the effectiveness of traffic flow predictions, more research should be done on the intricate topologies and workings of metropolitan arterial networks [113].

Building prediction algorithms that can cope with incomplete and inaccurate data, as well as the randomness in the temporal-spatial coverage of GPS data, is another practical use due to shortcomings in real-world data [114].

NN-based models are renowned for being difficult to interpret. To extract relevant information from the prediction model, such as relationships between road segments and the functions of the segments in a transportation network, it may therefore be helpful to increase the transparency of NN models [4].

Another intriguing field is the development and assessment of prediction models for nonrecurrent situations. Utilizing additional forms of data, such as weather, incidents, or special events, could be a potential strategy to increase flexibility so the models perform well under both typical and atypical circumstances [115].

References

[1] M.J. Br-uton, Introduction to transportation planning, Routledge, 2021.

[2] I. Lana, J. Del Ser, M. Velez, E.I. Vlahogianni, Road traffic forecasting: Recent advances and new challenges, IEEE Intelligent Transportation Systems Magazine, 10(2) (2018) 93-109.

[3] W. Min, L. Wynter, Real-time road traffic prediction with spatio-temporal correlations, Transportation Research Part C: Emerging Technologies, 19(4) (2011) 606-616.

[4] L.N. Do, N. Taherifar, H.L. Vu, Survey of neural network-based models for short-term traffic state prediction, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 9(1) (2019) e1285.

[5] D. Nikovski, N. Nishiuma, Y. Goto, H. Kumazawa, Univariate short-term prediction of road travel times, in: Proceedings. 2005 IEEE Intelligent Transportation Systems, 2005., IEEE, 2005, pp. 1074-1079.

[6] G. Bontempi, S. Ben Taieb, Y.-A.L. Borgne, Machine learning strategies for time series forecasting, in: European business intelligence summer school, Springer, 2012, pp. 62-77.

[7] B.L. Smith, B.M. Williams, R.K. Oswald, Comparison of parametric and nonparametric models for traffic flow

forecasting, Transportation Research Part C: Emerging Technologies, 10(4) (2002) 303-321.

[8] B. Pan, U. Demiryurek, C. Shahabi, Utilizing real-world transportation data for accurate traffic prediction, in: 2012 ieee 12th international conference on data mining, IEEE, 2012, pp. 595-604.

[9] Y. Ji, D. Tang, P. Blythe, W. Guo, W. Wang, Short-term forecasting of available parking space using wavelet neural network model, IET Intelligent Transport Systems, 9(2) (2015) 202-209.

[10] B.M. Williams, Multivariate vehicular traffic flow prediction: evaluation of ARIMAX modeling, Transportation Research Record, 1776(1) (2001) 194-200.

[11] M. Khashei, M. Bijari, A novel hybridization of artificial neural networks and ARIMA models for time series forecasting, Applied soft computing, 11(2) (2011) 2664-2675.

[12] A. Adas, Traffic models in broadband networks, IEEE communications Magazine, 35(7) (1997) 82-89.

[13] T. Alghamdi, K. Elgazzar, M. Bayoumi, T. Sharaf, S. Shah, Forecasting traffic congestion using ARIMA modeling, in: 2019 15th international wireless communications & mobile computing conference (IWCMC), IEEE, 2019, pp. 1227-1232.

[14] C. Xu, Z. Li, W. Wang, Short-term traffic flow prediction using a methodology based on autoregressive integrated moving average and genetic programming, Transport, 31(3) (2016) 343-358.

[15] H. Dong, L. Jia, X. Sun, C. Li, Y. Qin, Road traffic flow prediction with a time-oriented ARIMA model, in: 2009 Fifth International Joint Conference on INC, IMS and IDC, IEEE, 2009, pp. 1649-1652.

[16] D. Yan, J. Zhou, Y. Zhao, B. Wu, Short-term subway passenger flow prediction based on ARIMA, in: International Conference on Geo-Spatial Knowledge and Intelligence, Springer, 2017, pp. 464-479.

[17] N. Sadek, A. Khotanzad, Multi-scale high-speed network traffic prediction using k-factor Gegenbauer ARMA model, in: 2004 IEEE International Conference on Communications (IEEE Cat. No. 04CH37577), IEEE, 2004, pp. 2148-2152.

[18] M. Delavary Foroutaghe, A. Mohammadzadeh Moghaddam, V. Fakoor, Time trends in gender-specific incidence rates of road traffic injuries in Iran, PloS one, 14(5) (2019) e0216462.

[19] H. Zhang, X. Wang, J. Cao, M. Tang, Y. Guo, A hybrid short-term traffic flow forecasting model based on time series multifractal characteristics, Applied Intelligence, 48(8) (2018) 2429-2440.

[20] S.V. Kumar, L. Vanajakshi, Short-term traffic flow prediction using seasonal ARIMA model with limited input data, European Transport Research Review, 7(3) (2015) 1-9.

[21] B. Ghosh, B. Basu, M. O'Mahony, Bayesian time-series model for short-term traffic flow forecasting, Journal of transportation engineering, 133(3) (2007) 180-189.

[22] M. Milenković, L. Švadlenka, V. Melichar, N. Bojović, Z. Avramović, SARIMA modelling approach for railway passenger flow forecasting, Transport, volume 33, issue: 5, (2015).

[23] O. Giraka, V.K. Selvaraj, Short-term prediction of intersection turning volume using seasonal ARIMA model, Transportation Letters, 12(7) (2020) 483-490.

[24] M. Yang, J. Xie, P. Mao, C. Wang, Z. Ye, Application of the ARIMAX model on forecasting freeway traffic flow, in: 17th COTA International Conference of Transportation Professionals, 2017, pp. 7-9.

[25] G.J. Ross, Parametric and nonparametric sequential change detection in R: The cpm package, Journal of Statistical Software, 66 (2015) 1-20.

[26] J. Lu, S. Chen, W. Wang, H. Van Zuylen, A hybrid model of partial least squares and neural network for traffic incident detection, Expert Systems with Applications, 39(5) (2012) 4775-4784.

[27] H. Zhang, S.G. Ritchie, Z.-P. Lo, Macroscopic modeling of freeway traffic using an artificial neural network, Transportation Research Record, 1588(1) (1997) 110-119.

[28] N.G. Polson, V.O. Sokolov, Deep learning for short-term traffic flow prediction, Transportation Research Part C: Emerging Technologies, 79 (2017) 1-17.

[29] M. Elhenawy, H. Rakha, Spatiotemporal traffic state prediction based on discriminatively pre-trained deep neural networks, Advances in Science, Technology and Engineering Systems, 2(3) (2017) 678-686.

[30] A. Boukerche, J. Wang, Machine Learning-based traffic prediction models for Intelligent Transportation Systems, Computer Networks, 181 (2020) 107530.

[31] Q. Liu, Y. Cai, H. Jiang, J. Lu, L. Chen, Traffic state prediction using ISOMAP manifold learning, Physica A: Statistical Mechanics and its Applications, 506 (2018) 532-541.

[32] J.Z. Zhu, J.X. Cao, Y. Zhu, Traffic volume forecasting based on radial basis function neural network with the consideration of traffic flows at the adjacent intersections, Transportation Research Part C: Emerging Technologies, 47 (2014) 139-154.

[33] I. Tahyudin, Time Series Prediction Using Radial Basis Function Neural Network, International Journal of Electrical & Computer Engineering (2088-8708), 5(4) (2015).

[34] H. Li, Y. Wang, X. Xu, L. Qin, H. Zhang, Short-term passenger flow prediction under passenger flow control using a dynamic radial basis function network, Applied Soft Computing, 83 (2019) 105620.

[35] L. Jun, G. Zuhua, Network traffic prediction using radial basis function neural network optimized by ant colony algorithm, Sensors & Transducers, 172(6) (2014) 224.

[36] D. Chen, Research on traffic flow prediction in the big

data environment based on the improved RBF neural network, IEEE Transactions on Industrial Informatics, 13(4) (2017) 2000-2008.

[37] J. Wang, S. Lu, S.-H. Wang, Y.-D. Zhang, A review on extreme learning machine, Multimedia Tools and Applications, (2021) 1-50.

[38] X. Jiang, H. Adeli, Dynamic wavelet neural network model for traffic flow forecasting, Journal of transportation engineering, 131(10) (2005) 771-779.

[39] T. Li, L. Sheng, Prediction for short-term traffic flow based on optimized wavelet neural network model, AIRCC's International Journal of Computer Science and Information Technology, 7(2) (2015) 183-192.

[40] Q. Chen, Y. Song, J. Zhao, Short-term traffic flow prediction based on improved wavelet neural network, Neural Computing and Applications, 33(14) (2021) 8181-8190.

[41] J.-F. Chen, S.-K. Lo, Q.H. Do, Forecasting short-term traffic flow by fuzzy wavelet neural network with parameters optimized by biogeography-based optimization algorithm, Computational intelligence and neuroscience, 2018 (2018).

[42] S. Reza, H.S. Oliveira, J.J. Machado, J.M.R. Tavares, Urban safety: an image-processing and deep-learning-based intelligent traffic management and control system, Sensors, 21(22) (2021) 7705.

[43] L. Shen, Freeway travel time estimation and prediction using dynamic neural networks, Florida International University, 2008.

[44] A. Miglani, N. Kumar, Deep learning models for traffic flow prediction in autonomous vehicles: A review, solutions, and challenges, Vehicular Communications, 20 (2019) 100184.

[45] B. Abdulhai, H. Porwal, W. Recker, Short-term traffic flow prediction using neuro-genetic algorithms, ITS Journal-Intelligent Transportation Systems Journal, 7(1) (2002) 3-41.

[46] P. Lingras, P. Mountford, Time delay neural networks designed using genetic algorithms for short term inter-city traffic forecasting, in: International conference on industrial, engineering and other applications of applied intelligent systems, Springer, 2001, pp. 290-299.

[47] M. Zhong, S. Sharma, P. Lingras, Short-term traffic prediction on different types of roads with genetically designed regression and time delay neural network models, Journal of computing in civil engineering, 19(1) (2005) 94-103.

[48] C. Heghedus, A. Chakravorty, C. Rong, Neural network frameworks. comparison on public transportation prediction, in: 2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), IEEE, 2019, pp. 842-849.

[49] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, Y. Wang, Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction, Sensors, 17(4) (2017) 818.

[50] S. Ishak, P. Kotha, C. Alecsandru, Optimization of dynamic neural network performance for short-term traffic prediction, Transportation Research Record, 1836(1) (2003) 45-56.

[51] H. Dia, An object-oriented neural network approach to short-term traffic forecasting, European Journal of Operational Research, 131(2) (2001) 253-261.

[52] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation, 9(8) (1997) 1735-1780.

[53] R. Li, Y. Hu, Q. Liang, T2F-LSTM method for long-term traffic volume prediction, IEEE Transactions on Fuzzy Systems, 28(12) (2020) 3256-3264.

[54] X. Ma, Z. Tao, Y. Wang, H. Yu, Y. Wang, Long short-term memory neural network for traffic speed prediction using remote microwave sensor data, Transportation Research Part C: Emerging Technologies, 54 (2015) 187-197.

[55] Z. Zhao, W. Chen, X. Wu, P.C. Chen, J. Liu, LSTM network: a deep learning approach for short-term traffic forecast, IET Intelligent Transport Systems, 11(2) (2017) 68-75.

[56] Z. Cui, R. Ke, Z. Pu, Y. Wang, Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction, arXiv preprint arXiv:1801.02143, (2018).

[57] R.L. Abduljabbar, H. Dia, P.-W. Tsai, S. Liyanage, Short-term traffic forecasting: an LSTM network for spatial-temporal speed prediction, Future Transportation, 1(1) (2021) 21-37.

[58] J. Bhatia, R. Dave, H. Bhayani, S. Tanwar, A. Nayyar, SDN-based real-time urban traffic analysis in VANET environment, Computer Communications, 149 (2020) 162-175.

[59] R. Fu, Z. Zhang, L. Li, Using LSTM and GRU neural network methods for traffic flow prediction, in: 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), IEEE, 2016, pp. 324-328.

[60] J. Guo, Z. Wang, H. Chen, On-line multi-step prediction of short term traffic flow based on GRU neural network, in: Proceedings of the 2nd International Conference on Intelligent Information Processing, 2017, pp. 1-6.

[61] C. Song, H. Lee, C. Kang, W. Lee, Y.B. Kim, S.W. Cha, Traffic speed prediction under weekday using convolutional neural networks concepts, in: 2017 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2017, pp. 1293-1298.

[62] C. Willis, D. Harborne, R. Tomsett, M. Alzantot, A deep convolutional network for traffic congestion classification, in: Proc NATO IST-158/RSM-010 Specialists' Meeting on Content Based Real-Time Analytics of Multi-Media Streams, 2017, pp. 1-11.

[63] D. Zang, J. Ling, J. Cheng, K. Tang, X. Li, Using convolutional neural network with asymmetrical kernels to predict speed of elevated highway, in: International conference on intelligence science, Springer, 2017, pp.

212-221.

[64] W. Zhang, Y. Yu, Y. Qi, F. Shu, Y. Wang, Short-term traffic flow prediction based on spatio-temporal analysis and CNN deep learning, Transportmetrica A: Transport Science, 15(2) (2019) 1688-1711.

[65] Q. Liu, B. Wang, Y. Zhu, Short-term traffic speed forecasting based on attention convolutional neural network for arterials, Computer-Aided Civil and Infrastructure Engineering, 33(11) (2018) 999-1016.

[66] R. Zoppoli, M. Sanguineti, G. Gnecco, T. Parisini, Neural approximations for optimal control and decision, Springer, 2020.

[67] D. Bhatt, C. Patel, H. Talsania, J. Patel, R. Vaghela, S. Pandya, K. Modi, H. Ghayvat, CNN variants for computer vision: History, architecture, application, challenges and future scope, Electronics, 10(20) (2021) 2470.

[68] S. Guo, Y. Lin, N. Feng, C. Song, H. Wan, Attention based spatial-temporal graph convolutional networks for traffic flow forecasting, in: Proceedings of the AAAI conference on artificial intelligence, 2019, pp. 922-929.

[69] B. Yu, H. Yin, Z. Zhu, Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting, arXiv preprint arXiv:1709.04875, (2017).

[70] W. Huang, H. Hong, M. Li, W. Hu, G. Song, K. Xie, Deep architecture for traffic flow prediction, in: International conference on advanced data mining and applications, Springer, 2013, pp. 165-176.

[71] H. Hong, W. Huang, G. Song, K. Xie, Metric-based multi-task grouping neural network for traffic flow forecasting, in: International symposium on neural networks, Springer, 2014, pp. 499-507.

[72] L. Zhao, Y. Zhou, H. Lu, H. Fujita, Parallel computing method of deep belief networks and its application to traffic flow prediction, Knowledge-Based Systems, 163 (2019) 972-987.

[73] F. Kong, J. Li, B. Jiang, H. Song, Short-term traffic flow prediction in smart multimedia system for Internet of Vehicles based on deep belief network, Future Generation Computer Systems, 93 (2019) 460-472.

[74] H. Tan, X. Xuan, Y. Wu, Z. Zhong, B. Ran, A comparison of traffic flow prediction methods based on DBN, in: CICTP 2016, 2016, pp. 273-283.

[75] S. Sengupta, S. Basak, P. Saikia, S. Paul, V. Tsalavoutis, F. Atiah, V. Ravi, A. Peters, A review of deep learning with special emphasis on architectures, applications and recent trends, Knowledge-Based Systems, 194 (2020) 105596.

[76] M. Labonne, Anomaly-based network intrusion detection using machine learning, Institut polytechnique de Paris, 2020.

[77] O. Yazdanbakhsh, S. Dick, Forecasting of multivariate time series via complex fuzzy logic, IEEE Transactions on Systems, Man, and Cybernetics: Systems, 47(8) (2017) 2160-2171.

[78] M. Han, K. Zhong, T. Qiu, B. Han, Interval type-2 fuzzy neural networks for chaotic time series prediction: A concise overview, IEEE transactions on cybernetics, 49(7) (2018) 2720-2731.

[79] K.Y. Chan, T.S. Dillon, Traffic flow prediction using orthogonal arrays and Takagi-Sugeno neural fuzzy models, in: 2014 International Joint Conference on Neural Networks (IJCNN), IEEE, 2014, pp. 35-41.

[80] M.-b. Pang, X.-p. Zhao, Traffic flow prediction of chaos time series by using subtractive clustering for fuzzy neural network modeling, in: 2008 Second International Symposium on Intelligent Information Technology Application, IEEE, 2008, pp. 23-27.

[81] M. Deshpande, P.R. Bajaj, Short term traffic flow prediction based on neuro-fuzzy hybrid sytem, in: 2016 International Conference on ICT in Business Industry & Government (ICTBIG), IEEE, 2016, pp. 1-3.

[82] H. Li, Research on prediction of traffic flow based on dynamic fuzzy neural networks, Neural Computing and Applications, 27(7) (2016) 1969-1980.

[83] J. Tang, F. Liu, Y. Zou, W. Zhang, Y. Wang, An improved fuzzy neural network for traffic speed prediction considering periodic characteristic, IEEE Transactions on Intelligent Transportation Systems, 18(9) (2017) 2340-2350.

[84] F. Saâdaoui, H. Saadaoui, H. Rabbouch, Hybrid feedforward ANN with NLS-based regression curve fitting for US air traffic forecasting, Neural Computing and Applications, 32(14) (2020) 10073-10085.

[85] A. Boukerche, Y. Tao, P. Sun, Artificial intelligence-based vehicular traffic flow prediction methods for supporting intelligent transportation systems, Computer networks, 182 (2020) 107484.

[86] Z. Han, J. Zhao, H. Leung, K.F. Ma, W. Wang, A review of deep learning models for time series prediction, IEEE Sensors Journal, 21(6) (2019) 7833-7848.

[87] T. Huang, S. Wang, A. Sharma, Highway crash detection and risk estimation using deep learning, Accident Analysis & Prevention, 135 (2020) 105392.

[88] A. Pradhan, Support vector machine-a survey, International Journal of Emerging Technology and Advanced Engineering, 2(8) (2012) 82-85.

[89] L. Vanajakshi, L.R. Rilett, A comparison of the performance of artificial neural networks and support vector machines for the prediction of traffic speed, in: IEEE Intelligent Vehicles Symposium, 2004, IEEE, 2004, pp. 194-199.

[90] A. Este, F. Gringoli, L. Salgarelli, Support vector machines for TCP traffic classification, Computer Networks, 53(14) (2009) 2476-2490.

[91] M. Amirabadi, A survey on machine learning for optical communication [machine learning view], arXiv preprint arXiv:1909.05148, (2019).

[92] M. Kylänpää, T. Pahikkala, Big Data Preprocessing for Multivariate Time Series Forecast, Computer Science,

(2020).

[93] L. Vanajakshi, L.R. Rilett, Support vector machine technique for the short term prediction of travel time, in: 2007 IEEE Intelligent Vehicles Symposium, IEEE, 2007, pp. 600-605.

[94] P. Theja, L. Vanajakshi, Short term prediction of traffic parameters using support vector machines technique, in: 2010 3rd International Conference on Emerging Trends in Engineering and Technology, IEEE, 2010, pp. 70-75.

[95] H. Yan, D.-J. Yu, Short-term traffic condition prediction of urban road network based on improved SVM, in: 2017 International Smart Cities Conference (ISC2), IEEE, 2017, pp. 1-2.

[96] Q. Chen, W. Li, J. Zhao, The use of LS-SVM for short-term passenger flow prediction, Transport, 26(1) (2011) 5-10.

[97] R. Wang, Application research of short-term traffic flow forecast based on bat algorithm support vector machine, in: IOP Conference Series: Materials Science and Engineering, IOP Publishing, 2020, pp. 012168.

[98] H. Drucker, C.J. Burges, L. Kaufman, A. Smola, V. Vapnik, Support vector regression machines, Advances in neural information processing systems, 9 (1996).

[99] J. Gong, L. Qi, M. Liu, X. Chen, Forecasting urban traffic flow by SVR, in: 2013 25th Chinese Control and Decision Conference (CCDC), IEEE, 2013, pp. 981-984.

[100] M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, L.D. Han, Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions, Expert systems with applications, 36(3) (2009) 6164-6173.

[101] D. Wei, H. Liu, An adaptive-margin support vector regression for short-term traffic flow forecast, Journal of Intelligent Transportation Systems, 17(4) (2013) 317-327.

[102] D. Zeng, J. Xu, J. Gu, L. Liu, G. Xu, Short term traffic flow prediction based on online learning SVR, in: 2008 Workshop on Power Electronics and Intelligent Transportation System, IEEE, 2008, pp. 616-620.

[103] L.E. Peterson, K-nearest neighbor, Scholarpedia, 4(2) (2009) 1883.

[104] X.-l. Zhang, G. He, H. Lu, Short-term traffic flow forecasting based on K-nearest neighbors non-parametric regression, Journal of Systems Engineering, 24(2) (2009) 178-183.

[105] D. Mladenović, S. Janković, S. Zdravković, S. Mladenović, A. Uzelac, Night Traffic Flow Prediction Using K-Nearest Neighbors Algorithm, Operational Research in Engineering Sciences: Theory and Applications, 5(1) (2022) 152-168.

[106] Z. Zheng, D. Su, Short-term traffic volume forecasting: A k-nearest neighbor approach enhanced by constrained linearly sewing principle component algorithm, Transportation Research Part C: Emerging Technologies, 43 (2014).

[107] X. Zhao, R. Bi, R. Yang, Y. Chu, J. Guo, W. Huang, J. Cao, Short-Term Traffic Flow Prediction Based on the Intelligent Parameter Adjustment K-Nearest Neighbor Algorithm, in: International Conference on Artificial Intelligence and Soft Computing, Springer, 2020, pp. 466-477.

[108] F.G. Habtemichael, M. Cetin, Short-term traffic flow rate forecasting based on identifying similar traffic patterns, Transportation research Part C: emerging technologies, 66 (2016) 61-78.

[109] B. Sun, W. Cheng, P. Goswami, G. Bai, Short-term traffic forecasting using self-adjusting k-nearest neighbours, IET Intelligent Transport Systems, 12(1) (2018) 41-48.

[110] A. Rasaizadi, A. Ardestani, S. Seyedabrishami, Traffic management via traffic parameters prediction by using machine learning algorithms, International Journal of Human Capital in Urban Management, 6(1) (2021) 57-68.

[111] P. Dell'Acqua, F. Bellotti, R. Berta, A. De Gloria, Time-aware multivariate nearest neighbor regression methods for traffic flow prediction, IEEE Transactions on Intelligent Transportation Systems, 16(6) (2015) 3393-3402.

[112] H.A. Abu Alfeilat, A.B. Hassanat, O. Lasassmeh, A.S. Tarawneh, M.B. Alhasanat, H.S. Eyal Salman, V.S. Prasath, Effects of distance measure choice on k-nearest neighbor classifier performance: a review, Big data, 7(4) (2019) 221-248.

[113] Y. Gong, Z. Li, J. Zhang, W. Liu, Y. Zheng, Online spatio-temporal crowd flow distribution prediction for complex metro system, IEEE Transactions on Knowledge and Data Engineering, (2020).

[114] N. Cressie, C.K. Wikle, Statistics for spatio-temporal data, John Wiley & Sons, 2015.

[115] X. Fei, C.-C. Lu, K. Liu, A bayesian dynamic linear model approach for real-time short-term freeway travel time prediction, Transportation Research Part C: Emerging Technologies, 19(6) (2011) 1306-1318.

[116] Tianhe L, Xiaojing Z, Dayi Q, Yufeng Y, Yicheng C, Short-term traffic flow prediction based on the optimization study of initial weights of the attention mechanism, Sustainability 2023, 15(2), 1374.

[117] Huiming D, Guan W, Partial differential grey model based on control matrix and its application in short-term traffic flow prediction, Applied Mathematical Modelling 116, 763-785, 2023.

[118] Tao J, Chenxi C, Forecasting citywide short-term turning traffic flow at intersections using an attention-based spatiotemporal deep learning model, Transportmetrica B: Transport Dynamics 11(1), 683-705, 2023

[119] Quang H, Yao F, Tien C, Short-term traffic speed forecasting model for a parallel multi-lane arterial road using GPS-monitored data based on deep learning approach, Sustainability 2022, Volume 14.

[120] Leina Z, Xinyu W, Yanpeg W, Yiming S, A novel

hybrid model of ARIMA-MCC and CKDE-GARCH for urban short-term traffic flow prediction, IET Intelligent Transport System. Volume 16, 2022.

[121] Xuecai X, Xiaofei J, Diaquan X, Chngxi M, A hybrid autoregressive fractionally integrated moving average and nonlinear autoregressive neural network model for short-term traffic flow prediction, IET Intelligent Transport System. Volume 27, 2023.

HOW TO CITE THIS ARTICLE

A. Taheri, R. Moghadasi, A. Rasaizadi, The short-term prediction of traffic parameters: a review of parametric and nonparametric approaches, AUT J. Civil Eng., 6(3) (2022) 369-386.

DOI: 10.22060/ajce.2023.22199.5821