

Network Traffic Prediction Model Considering Road Traffic Parameters Using Artificial Intelligence Methods in VANET

SANAZ SHAKER SEPASGOZAR^{ID} AND SAMUEL PIERRE, (Senior Member, IEEE)

Mobile Computing and Networking Research Laboratory (LARIM), Department of Computer and Software Engineering, Polytechnique Montreal, Montreal, QC H3T 1J4, Canada

Corresponding author: Sanaz Shaker Sepasgozar (sanaz-2.shaker-sepasgozar@polymtl.ca)

ABSTRACT Vehicular Ad hoc Networks (VANETs) are established on vehicles that are intelligent and can have Vehicle-to-Vehicle (V2V) and Vehicle-to-Road Side Units (V2R) communications. In this paper, we propose a model for predicting network traffic by considering the parameters that can lead to road traffic happening. The proposed model integrates a Random Forest- Gated Recurrent Unit- Network Traffic Prediction algorithm (RF-GRU-NTP) to predict the network traffic flow based on the traffic in the road and network simultaneously. This model has three phases including network traffic prediction based on V2R communication, road traffic prediction based on V2V communication, and network traffic prediction considering road traffic happening based on V2V and V2R communication. The hybrid proposed model which implements in the third phase, selects the important features from the combined dataset (including V2V and V2R communications), by using the Random Forest (RF) machine learning algorithm, then the deep learning algorithms to predict the network traffic flow apply, where the Gated Recurrent Unit (GRU) algorithm gives the best results. The simulation results show that the proposed RF-GRU-NTP model has better performance in execution time and prediction errors than other algorithms which used for network traffic prediction.

INDEX TERMS Vehicular network, network traffic prediction, road traffic prediction, regression methods, classification methods, machine learning algorithms, deep learning algorithms.

I. INTRODUCTION

One of the important technologies for the Intelligent Transportation System (ITS) is VANET that tries to make the environment safer and have better transportation using wireless communications [1].

The traffic flow prediction with high accuracy is a significant issue in current transportation systems. It can help have the best path planning, make a better choice in selecting the greater route for travelers and decrease the traffic flow. Distinguishing that where and when the traffic will happen is a promising solution for managing transportation [2]. However, the new perspective of network traffic flow is that the traffic in the road could affect network traffic. According to the V2V communications in VANET, vehicles can send packets to each other to forecast the road traffic. By increasing the

number of vehicles and traffic on the road, the number of packets sent would grow, leading to network traffic.

Previous studies worked on road traffic and network traffic separately, and we investigated them in the literature review. However, most of them addressed the traffic problem on the road or in the network independently, while in this paper, we will discover the relation between road and network traffic parameters together with the aim of network traffic prediction. Intelligent ways via machine learning (ML) techniques are the optimum solutions that can address traffic prediction problems with the aim of traffic flow prediction. There are some computational approaches like Bayesian modeling, fuzzy logic, hybrid modeling, Neural Networks (NN), and statistical modeling, which most of them, specially the NN, are promising solutions aiming to improve the accuracy of prediction in data traffic flow [3].

The significant point that should consider in all these ways, is the accuracy of prediction. ML techniques are divided into three types: Unsupervised Learning (training would be based

The associate editor coordinating the review of this manuscript and approving it for publication was Cheng Chin^{ID}.

on unlabeled data), Supervised Learning (training would be based on labeled data), and Reinforcement Learning (it learns from the performance of the learning agent). Moreover, some types of ML schemes like Transfer Learning and Online Learning are sub-categorized by these three types of ML schemes [4].

Another promising solution in the case of a large and complex dataset is deep learning (DL) algorithms for prediction problems. It has different types of algorithms that Recurrent Neural Network (RNN) [5], [6] and Convolutional Neural Network (CNN) [7] are the two famous algorithms that are used in many studies. Generally, the RNN has two modules called Long Short-Term Memory (LSTM) [8] and Gated Recurrent Unit (GRU) [9], [10], where the LSTM algorithm is similar to RNN by intention to address the vanishing problem. One of the most critical features of these algorithms is that they can learn dependencies for a long time with the aim of prediction in time-series datasets, and the GRU algorithm is like LSTM with more minor complications due to the number of its gate that leads to making it faster than LSTM [11]. Furthermore, to extract more features and bidirectional dependencies, Bi-directional Long Short-Term Memory (Bi-LSTM) algorithm can be used. In this kind of algorithm, the sequence of the process can be done in two directions (forward and backward) using two different hidden layers [12].

In this work, we propose a network traffic prediction method considering road traffic parameters. To the best of our knowledge, it is the first time that network traffic is predicted due to road traffic happening. We try different machine learning and deep learning algorithms with the aim of network traffic prediction and divide our work into three phases. The first phase is about network traffic prediction. The second phase is about road traffic prediction. The third phase is a combination of the two previous phases with the aim of network traffic prediction considering the parameters, which are effective in road traffic that could affect the network traffic as well. We deploy two datasets from the Global Positioning System (GPS): the first one is based on V2R communication that is used for network traffic prediction and the second one is based on V2V communication that is used for road traffic prediction. The significant contributions of this work are as follows:

- We predict the network traffic flow using ML algorithms applied on a real dataset derived from V2R communications based on the packets sent by the vehicles to the Road-Side Units (RSUs). Furthermore, we implement variant ML algorithms like the RF, K-Nearest Neighbors (KNN), Support Vector Machines (SVM), and Naive Bayes (NB) algorithms, where the best performance belongs to the RF.
- We concentrate on the impact of the sender's speed as a road parameter and predict the road traffic flow considering its effect on the network traffic flow. We take advantage of a real dataset built on V2V communications intending to predict the speed of senders using

deep learning algorithms to aim for road traffic prediction. We tried different DL algorithms like the LSTM, Bi-LSTM, and GRU, where we got the best performance by the GRU algorithm.

- We investigate the effect of the last two previous steps based on road and network parameters on network traffic happening with the aim of network traffic prediction using both machine learning and deep learning algorithms. At this step, we combine both datasets (V2V and V2R), then we do feature selection using the RF algorithm to find the most influential parameters in the network traffic. After that, by implementing the GRU algorithm, we predict network traffic based on the adequate parameters in the road and network together. The results show that the novel proposed RF-GRU-NTP model would predict the network traffic affected by road traffic happening in the VANET environment more accurately than pure algorithms. Also, duration time for executing the proposed RF-GRU-NTP model aiming predict network traffic flow takes in average 70% less time than LSTM and Bi-LSTM.

The originality of this paper lies on the fact that it combines machine learning and deep learning techniques into a single network traffic flow prediction model by considering road parameters and network parameters simultaneously.

The rest of this paper is organized as follows. Section II presents related work and provides an overview of the proposed prediction approaches in road traffic prediction and network traffic prediction. Section III explains the methodology and describes the implementation of the proposed prediction model phases. Section IV provides the evaluation results of the experimental validation for each phase. Finally, the conclusion of the study and our direction for future work are placed in section V.

II. BACKGROUND AND RELATED WORK

Intelligent traffic prediction can support the critical advantages of solving traffic problems in the cities [13]. On the other hand, intelligent methods can help us predict the traffic flow considering different effective parameters. ML and DL methods are promising solutions for analyzing data and getting more accurate prediction results [14]. Several studies have been proposed different models to predict network traffic and road traffic independently using learning algorithms. Moreover, some researchers attempt to predict the road traffic flow using weather conditions that affect traffic flow. From a certain point of view, we can divide the previous works into two parts: road traffic prediction and network traffic prediction, where most of them used different machine learning and deep learning algorithms.

A. NETWORK TRAFFIC PREDICTION

In [15], the authors proposed a framework aiming to improve the prediction of network-wide link-level traffic deploying LSTM. They collected statics by Software Defined

Network (SDN) or through SNMP measurements to forecast future throughputs of the network. They implemented their model on a real dataset, and they considered one hour, including a large amount of traffic of packets. They tried three different variations of LSTM, including Vanilla LSTM (vlstm), Delta LSTM (dlstm) and Multivariate LSTM (mlstm), and three types of Auto Regressive Integrated Moving Average model (ARIMA), including simple ARIMA model, Delta ARIMA model (darima), First-Order Autoregressive ARIMA (as1). Their experimental results showed that all variants of the LSTM had better performance than ARIMA based models for modeling the network traffic. However, they could have tried with other deep learning algorithms, which are more accurate than ARIMA baselines, to compare.

Intending to optimize network resource allocation and network traffic prediction, in [16], the authors deployed deep learning algorithms using a real dataset with five minutes time step for data-driven. They proposed the Evaluation Automatic Module (EAM) algorithm, which made the learning process automate and generalize the model for prediction aiming to have the best performance. Their proposed model is composed of two parts: in the first part, they used Artificial Neural Network (ANN) based on GRU algorithm to train the prediction model; and in the second part, they used EAM algorithm for model evaluation in every single iteration during the learning process. In addition, they used the Mean Absolute Error (MAE) metric for performance evaluation of the proposed model. The resource allocation part compared their results with static planning, which calculates the highest bandwidth related to the links in the traffic matrix. The experimental results showed that the proposed model had good accuracy in prediction and allocation resources.

A new methodology was aimed to improve the network traffic prediction proposed in [17]. They tried to predict network traffic intelligently using sequence mining, and for this purpose, they implemented the LSTM and Adaptive Neuro-Fuzzy Inference System (ANFIS) as a time series model. They selected real network data to implement their proposed model. Moreover, they clustered similar data using fuzzy c-means clustering. As well, they clustered the objects into five clusters as input for the LSTM and ANFIS algorithms with the aim of network traffic prediction. They evaluated the proposed model using metrics that diagnosed the prediction errors, which showed their model could decrease the prediction error and increase the network's performance.

Different types of CNN on real data, to find parameters of a network, which are optimal for network traffic prediction implemented in [18]. They run the algorithm in 1000 iterations with a [0.01-0.5] learning rate and tried different algorithms like multi-layer perceptron (MLP), CNN-RNN, CNN, CNN-LSTM and CNN-GRU. The results showed that the CNN and its different types of it are able to overcome other classical machine learning algorithms. The authors in [3], tried to develop and optimize the traffic learning by deploying the Taguchi method via layer-by-layer features categorized with the unsupervised algorithm. They proposed an

optimized structure for traffic flow prediction trained with stacked autoencoder (SAE) and Levenberg–Marquardt (LM) algorithm to implement a short-term traffic prediction. Their approach aims to increase the precision of traffic flow prediction. The proposed model is a kind of deep architecture of the neural network using the Taguchi method to learn the features of traffic flow. To implement and evaluate their method, they used real collected data and implemented some constraints to the hidden layers, like considering five hidden layers. They assumed that the neurons were inactive most of the time and used the LM algorithm for training the input that the last autoencoder generated. They compared their result with different algorithms like a radial basis function NN (RBFNN), hybrid exponential smoothing and the LM algorithm with NNs (EXP-LM), particle swarm optimization algorithm with NNs (PSO) and they found that the accuracy rate of prediction in the proposed model is almost 90% which had the best performance. The traffic flow data might be irregular because of unpredictable situations, and the proposed model was practical in dealing with this problem.

A new method for LTE network traffic prediction is presented in [19]. They used three different machine learning algorithms, including the RF, Bagging, and SVM on public cellular traffic datasets with the aim of network traffic prediction. They evaluated their performances using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Coefficient of Determination R^2 . The results showed that the Bagging has good results in a combination of numerical and categorical features. The Bagging and RF got the lowest and highest RMSE, 2.59 and 3.38, respectively. Moreover, the Bagging and RF got the lowest and highest MAE, 1.60 and 2.19 as well as the Bagging got the highest R^2 with 50.8%. However, the Bagging took 116 (s) and RF took 112 (s) time for learning, and the SVM took 6 (s) that is less time in comparison with them.

B. ROAD TRAFFIC PREDICTION

Deep Neural Network (DNNs) can forecast the traffic flow with big data. However, there are some challenges about spatial-temporal issues. In [2], the authors proposed a forecast model to cover these problems and to analyze the internal mechanism of DNN based traffic flow prediction (DNN-BTF) on traffic flow data with reliable precision, in order to enhance the exactness of prediction. For validation of their model, they used data from the open-access database PeMS.¹ Moreover, they used the RNN and CNN (with an entire convolutional network) algorithms for temporal and spatial features, respectively.

They adopted three different indexes for performance evaluation: the Mean Relative Error (MRE), the MAE, and the RMSE. Furthermore, they compared their proposed model with the adaptive absolute shrinkage and selection operator (LASSO) [20], traditional shallow back-propagation neural network (BPNN), stacked autoencoder (SAE) [21],

¹<http://pems.dot.ca.gov/>

DeepST [22], and Sequence to Sequence learning (StoS) methods. The results showed that their model outperformed them. In [23], the authors proposed a spatial-temporal feature selection algorithm using GRU (GRU + STFSA) for prediction of short-term traffic flow, which was a combination of spatial and temporal analysis using the GRU. They compared the results with CNN and simple GRU algorithms. They evaluated their model by using the MAE, the Mean Absolute Percentage Error (MAPE), and the RMSE metrics. The proposed model showed better performance in stability and accuracy with a minor error in prediction.

The hybrid CNN-LSTM model proposed in [24] for traffic flow prediction of GPS data. They adopted a greedy policy approach to train the CNN-LSTM model to solve hybrid models' complexity and time-consuming training process. The experimental results showed that the deep hybrid proposed model could predict the traffic more accurately in less run time than the Linear(a model with dense layer using softmax), CNN and CNN-LSTM model, considering both temporal and spatial features. The authors in [25], presented a hybrid framework for predicting short-term traffic flow based on Support Vector Regression (SVR). They applied the RF for finding the compelling features and improved the Genetic Algorithm (GA) to find optimal features. They collected data from real-world datasets to implement the proposed RF-CGASVR model, and after selecting features by the RF, the training phase was done. Intending to evaluate the proposed model, they tried it into two layouts. The first one was straight, which was designed with the aim of performance evaluation of light scenarios. The second one was the crossroad layout, which was about intersections and different roads. They used the RMSE and MAPE as metrics for performance evaluation. The results showed that the proposed model performed well while they did not consider long-term traffic. They also compared the evaluation results of proposed model with different algorithms like ARIMA and SVR-based models. However, there are some studies about traffic prediction considering weather conditions using different deep learning algorithms [26]–[30].

Based on the large amount of data produced in smart cities by different intelligent vehicles, road parameters that can cause road traffic could affect network traffic. The Artificial Intelligence (AI) approach can be an optimal solution for network traffic prediction. However, the previous works focused on network traffic prediction or road traffic prediction, so the lack of study on considering these two issues together motivated us to propose a novel network traffic prediction approach based on road traffic flow.

Table 1 shows a summary of previous studies in road traffic prediction, network traffic prediction and their proposed methods.

III. METHODOLOGY

In this section, we propose a novel RF-GRU-NTP model using machine learning and deep learning algorithms with the aim of network traffic flow prediction

TABLE 1. Summary of related work in road and network traffic prediction.

Proposed method	Category	Comparison algorithms	Approach
LSTM	Network traffic prediction	vlstm, mlstm, dlstm, arima, darima and asl	[15]
EAM	Network traffic prediction	GRU	[16]
ANFIS	Network traffic prediction	LSTM	[17]
CNN	Network traffic prediction	MLP, CNN-RNN, CNN-LSTM and CNN-GRU	[18]
SAE-LM	Network traffic prediction	RBFNN, EXP-LM and PSONN	[3]
LTE network prediction method	Network traffic prediction	RF, Bagging and SVM	[19]
DNN-BTF	Road traffic prediction	RNN and CNN	[2]
GRU+STFSA	Road traffic prediction	GRU and CNN	[23]
Hybrid CNN-LSTM	Road traffic prediction	Linear and CNN	[24]
RF-CGASVR	Road traffic prediction	ARIMA and SVR-based models	[25]

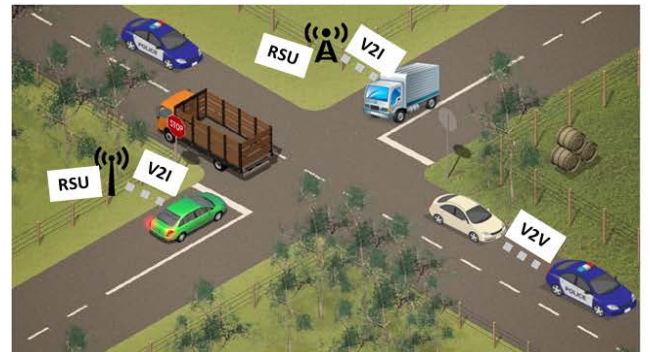


FIGURE 1. The architecture of VANET.

in VANET. Figure 1 shows the architecture of VANET environment.

VANET is a dynamic environment due to the presence of moving characters (like vehicles) in the network that make it complicated in prediction problems. By applying machine learning algorithms on appropriate datasets, we can increase prediction accuracy in traffic problems [31]. Furthermore, deep learning algorithms are employed to predict complex patterns in faster and more accurate ways.

Many parameters can affect the network traffic flow, so the proposed model is divided into three phases with the objective of network traffic prediction considering different effective parameters. Figure 2 shows the architecture of the proposed model, including three phases.

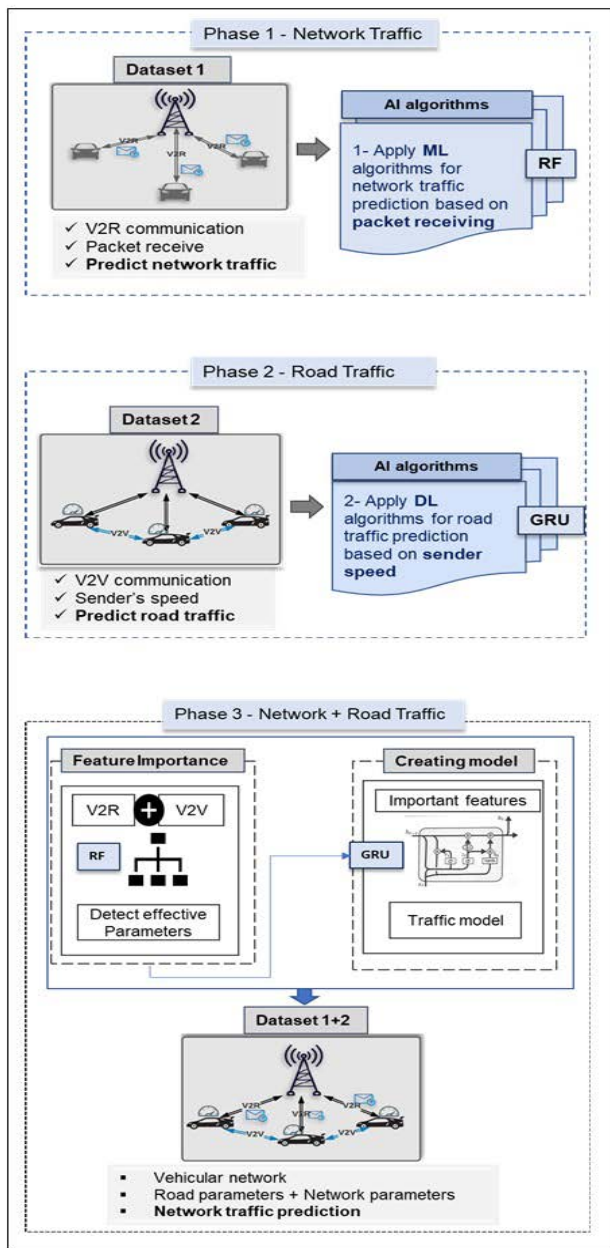


FIGURE 2. The architecture of proposed RF-GRU-NTP model.

A. PHASE 1 - NETWORK TRAFFIC FLOW

In the first phase, we just focused on V2R communications, and we predicted the network traffic flow based on the packets sent by vehicles to the RSUs. In this step, we implemented classification methods, then we tried different machine learning algorithms like the KNN [32], RF [33], NB [34], and SVM [35], intending to have the best performance and accuracy in prediction results with regards to precision, recall, F1-score, and accuracy. Moreover, we used the area under the Receiver Operating Curve (ROC) metric, confusion matrix, and computation time to evaluate the algorithms considering all aspects of the fittest algorithm aiming to network traffic prediction.

1) RANDOM FOREST ALGORITHM

The RF is a supervised learning algorithm composed of several decision trees that are combined together to get more accurate prediction results. Therefore, we can use the RF algorithm in regression and classification problems, as well as it can overcome the main problem of machine learning algorithms that is over fitting [36]. In our research work, we deployed four different machine learning algorithms on a dataset that got by GPS [37], and we got the best results with the RF algorithm for network traffic prediction. The vehicular network dataset (the network type was 802.11 adhoc) that we used was including the measure of short-range communications performance between vehicles together and vehicle and RSUs. Some parameters like latitude, longitude, speed of sender, speed of the receiver, packet receiving are reported in the dataset. We selected 10000 records of one day to predict network traffic flow based on packets that are sent by vehicles to the RSUs. To evaluate all deployed classification methods, we used a confusion matrix intending to find out the number of correct classifier's predictions, the incorrect ones, and even in the situation that they got confused.

2) CLASSIFICATION METRICS

We used four metrics to evaluate our classification model [38]: the precision, the recall, the F1-score, and the accuracy.

- 1) *Precision* shows the number of true positive predicted observations over all the positive predicted including false and true, which is denoted by:

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

- 2) *Recall* represents the ratio of true positive predicted observations over all the observations in a real class, which is designated by:

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

- 3) *F1 - score* is affected by Precision and Recall, and it is denoted by:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3}$$

- 4) *Accuracy* shows the number of true positive and true negative observations over all the number of samples, which is designated by:

$$Accuracy = \frac{TP + TN}{(TP + TN + FP + FN)} \times 100 \tag{4}$$

where,
 TP: True Positive;
 FP: False Positive;
 TN: True Negative;
 FN: False Negative.

Figure 3 shows the first phase workflow of the proposed model including the algorithms, libraries, tools, and metrics

that we used. Our vehicular network dataset contains V2R communications that got by GPS receiver.

We labeled data into two classes: class 1 for a case that packet is received, and we assumed it in a non-traffic situation; and class 0 for not receiving a packet that we presumed as a traffic situation.

$$\text{Traffic flow classes} = \begin{cases} \text{In non-traffic situation;} \\ \text{Packet received: class 1} \\ \text{In traffic situation;} \\ \text{Packet does not receive: class 0} \end{cases} \quad (5)$$

After that, we split the dataset based on different tries and considering the best computational cost into 75% for train and 25% for test. At the next level, we implemented four different machine learning algorithms on the train set to predict network traffic and create our model with the highest performance. Based on various metrics that we used to evaluate the models, we had got the best results when we deployed the RF algorithm for network traffic prediction considering packet receiving as a network parameter.

B. PHASE 2 - ROAD TRAFFIC PREDICTION

The second phase focuses on V2V communication considering the speed of vehicles (“sender speed”), which are sending packets as a road parameter to predict the road traffic flow. At this phase, we tried regression methods and deployed three different deep learning algorithms, including the GRU, Bi-LSTM, and LSTM. We evaluated their performances using some evaluation metrics for deep regression algorithms, and we got the best result with the GRU algorithm in road traffic prediction.

1) GRU ALGORITHM

The RNN algorithm has different types, and the GRU is one of the efficient algorithms derived from the LSTM. The GRU inherits the essential advantage of RNN, which is learning features automatically. It can act like the LSTM and memorize the performance of prediction for the long term in a faster way. The LSTM has three gates containing input gate, forget gate, and output gate while the GRU algorithm has just two gates including update gate and reset gate which makes the GRU less complicated and more efficient [23]. The role of the reset gate is determining how much of the previous information needs to be forgotten, and the update gate defines the amount of memory needed to keep around and pass through to the future information [11]. Figure 4 shows the internal structure of the GRU algorithm, and its equations are as follows:

$$\text{Update gate: } Z_t = \sigma(W_z x_t + U_z h_{t-1}) \quad (6)$$

$$\text{Reset gate: } r_t = \sigma(W_r x_t + U_r h_{t-1}) \quad (7)$$

$$\text{Process input: } \tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t]) \quad (8)$$

$$\text{Hidden state update: } h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \quad (9)$$

The update gate has been calculated in (6) for time step t , where x_t is the input of update gate and W_z represents its weight that their multiplication would be accumulate with the sum of h_{t-1} that shows the past information in time $t - 1$ and its weight that shows with U_z . Then the result will be squashed between 0 and 1 by applying the sigmoid activation function (σ) [39]. Activation functions can normalize the output of neural network models. This gate determines how much of the previous information is necessary for passing to the future. The reset gate is calculated in (7) that is similar to the update gate. However, the difference is that the reset gate will decide which past information is needed to forget. The effect of gates on the final output will be calculated in (8). W shows the weight of the update gate, r_t represents the input that got by the reset gate, the output of the previous neuron showed by h_{t-1} and the input of the current neuron is presented by x_t , which they are multiply on tanh as a nonlinear activation function. This procedure shows the process for a new memory content. The output of the current neuron is represented by h_t in (9), z_t represents the update gate, h_{t-1} is showing the output of the previous unit and \tilde{h}_t represents the output value that is pending in the current unit [23], [39].

We considered a situation for traffic flow in the road, which is when the speed of senders drops off under 60 Km/h, we would have traffic in the road considering the speed of vehicles in our dataset are in the range of (0,104 Km/h) and there was not any speed limit.

$$\text{Road traffic happening} = \begin{cases} \text{In non-traffic situation;} \\ \text{Sender speed} > 60 \text{ Km/h} \\ \text{In traffic situation;} \\ \text{Sender speed} < 60 \text{ Km/h} \end{cases} \quad (10)$$

After data collection, we did pre-process, data cleansing, and scaling using the MinMax method [40]. We split the dataset into 75% for train and 25% for a test. Based on our dataset and parameters, we implemented three different regression algorithms, including the LSTM, GRU and Bi-LSTM, to speed prediction and created our model with the highest performance. To implement the regression algorithms, the Keras library alongside TensorFlow was used [41]. The data has been passed 120 times for training; thus, the model was fit to run on 120 epochs. Different optimizers have been tried like Stochastic Gradient Descent methods (SGDs) [42], Adaptive Gradient Algorithm (AdaGrad) [43], Adaptive Moment Estimation (Adam) [44], and Root Mean Square Propagation (RMSProp) [45]. We got the best result from the Adam optimizer for stochastic optimization, then to avoid overfitting and improve the performance, we added a dropout layer into our algorithms, then we evaluated our model using different evaluation metrics. Figure 5 shows the workflow and structure of phase two.

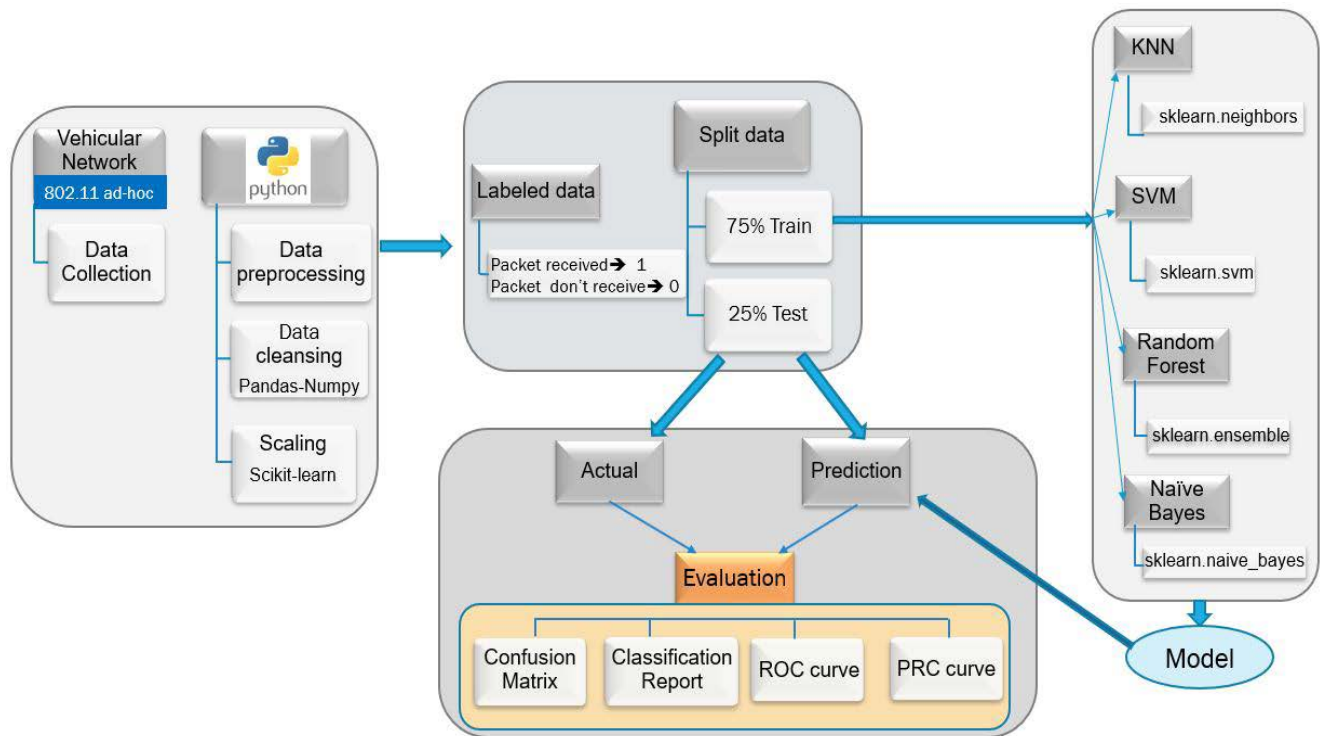


FIGURE 3. The workflow of network traffic prediction (phase1).

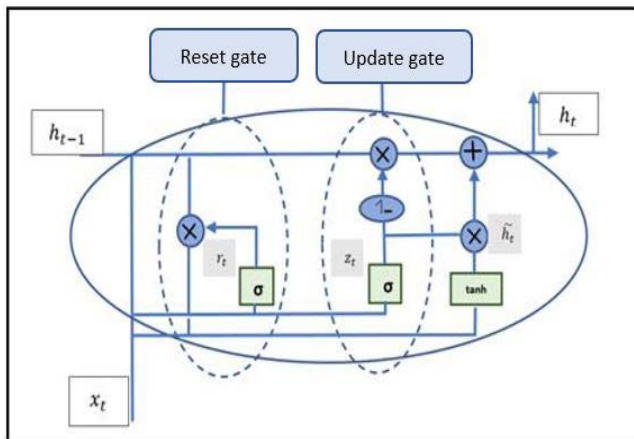


FIGURE 4. The structure of the GRU.

2) REGRESSION EVALUATION METRICS

We used four evaluation metrics including the Mean absolute error (MAE), the Mean Squared Error (MSE), the Root Mean Squared Error (RMSE) and the $R^2 - SCORE$, to evaluate our regression models [46].

- 1) MAE (Mean absolute error), shows the sum of the total difference between the actual values and the predicted values, which is denoted by:

$$MAE = \left(\frac{1}{n}\right) \sum_{i=1}^n |x_i - \hat{x}_i| \quad (11)$$

- 2) MSE (Mean Squared Error), illustrates the difference between the actual and the predicted value by squared, which is designated by:

$$MSE = \left(\frac{1}{n}\right) \sum_{i=1}^n (x_i - \hat{x}_i)^2 \quad (12)$$

- 3) RMSE (Root Mean Squared Error), can show a more accurate error rate by squared MSE metric, which is denoted by:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2} \quad (13)$$

- 4) $R^2 - SCORE$, represents the ratio and the efficiency of the predicted model and how to fit it in comparison with the actual value, as well as it ranges from 0 to 1 where the closest value to 1 shows the better model. It is denoted by:

$$R^2 = 1 - \frac{\sum(x_i - \hat{x}_i)}{\sum(x_i - x_{avg})} \quad (14)$$

where,

x_i = Real Value;

\hat{x}_i = Predicted Value;

x_{avg} = Scores of all outputs are averaged with uniform weight; n = The number of examples of test or verification set.

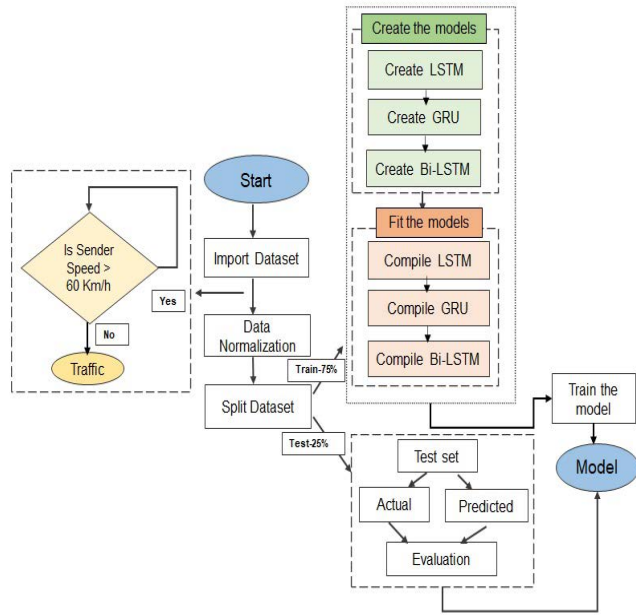


FIGURE 5. The workflow of road traffic prediction (phase 2).

Based on evaluation results, we got the best performance by deploying the GRU algorithm to predict “sender speed” as a road parameter to predict road traffic.

C. PHASE 3 - NETWORK TRAFFIC PREDICTION CONSIDERING ROAD PARAMETERS

Finally, in the third phase, we intend to predict the network traffic flow considering the parameters that affect road traffic. We propose an RF-GRU-NTP model to predict the network traffic. To the best of our knowledge, it is the first time that network traffic flow is predicted based on effective parameters in road traffic flow. Furthermore, we propose a model that combines deep learning and machine learning algorithms to predict the network traffic flow. The structure of the proposed model in Phase 3 is presented in Figure 6.

This phase includes a machine learning algorithm as a feature selection and a deep learning algorithm in terms of network traffic prediction considering road network parameters. In this way, at the first step, we combined V2V and V2R datasets. We investigated on parameters including “receiver speed,” “packet receive,” “time,” “signal strength,” and “noise strength,” then by deploying the RF algorithm, the features that are effective on “sender speed” as a road parameter in road traffic happening have been found. Afterward, we passed the important features through the GRU as inputs to predict the network traffic flow. Also, we considered 75% of data for training and 25% of that for testing.

1) EFFECTIVE PARAMETERS DETECTION BY THE RANDOM FOREST

The RF algorithm plays the role of detecting effective parameters on the network traffic flow. The RF algorithm is based on selecting a vector (A, B) irregularly and including trees

that are growing to the decision tree. Then, by voting through the n trees and considering the training data (A, B), which provides k samples for building a regression tree, it extracts the c subset and puts them back randomly. The $i(i \in c)$ subspace has been extracted, and the other samples that did not extract in the subspace are out-of-bag (OOB) data. To construct the regression tree’s feature space, a fixed dimension vector would be chosen from an M dimensional vector, and it can be considered an input variable [47].

Then considering minimum variance, which is computed during the growth process, the splitting can be calculated as follows:

$$I = \min_s \frac{\sum_{s=1}^k (y_s - \bar{y}_s)^2}{k} \tag{15}$$

where,

- I = Optimal splitting variable;
- S = Embedded sample dimension;
- y_s = Value of variable;
- \bar{y}_s = Average vale of variable.

The RF algorithm is made by the growth of trees, and we can compute the effect of out of bag data on the model as follows:

$$MSE = \left(\frac{1}{n}\right) \sum_{i=1}^n (x_i - \hat{x}_i)^2 \tag{16}$$

where,

- x_i = Real Value;
- \hat{x}_i = Predicted Value;
- n = The number of samples out of bag.

The RF algorithm, in terms of forming the out of bag data matrix, will generate n new trees and will get the different ranks of $MSE [MSE_1, MSE_2, \dots, MSE_n]$. Moreover, measuring the importance of the input variable is calculated as follows [47]:

$$V_{im} = \frac{\sum_{j=1}^n (MSE_j - MSE_{nj})^2}{n \cdot S_E} \tag{17}$$

where,

- V_{im} = Importance score;
- $n(j \in n)$ = Number of decision tree;
- S_E = Standard error of n decision tree.

2) NETWORK TRAFFIC PREDICTION BY THE GRU

After finding out which features are more important to be as an input variable for GRU, we made a new filtered dataset for more accurate prediction using effective parameters. At this level, we filtered our combined dataset (V2V + V2R) and defined traffic based on “sender speed” and assume if the sender speed is less than 60 Km/h, then we will have traffic. Then using RF, we figured that the most effective parameters on “sender speed” were “receiver speed” and “packet receiving.” Then by applying the GRU algorithm, we predicted the network traffic flow based on the most important parameters in the road and network. Based on evaluation results, the proposed model can predict the network traffic

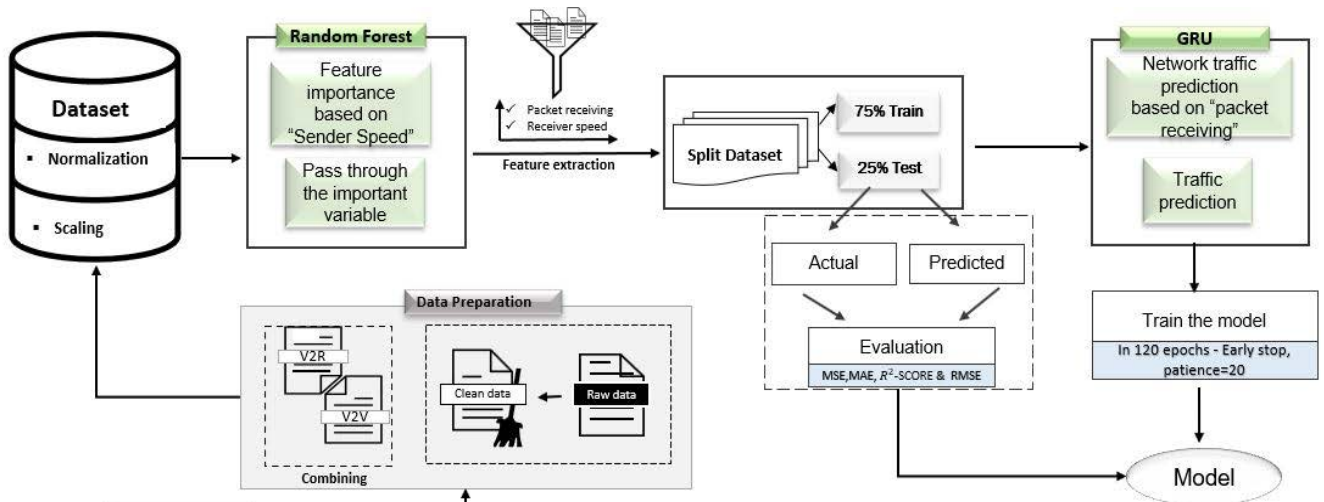


FIGURE 6. The structure of the proposed RF-GRU-NTP model (phase 3).

flow more accurately than other algorithms like LSTM and Bi-LSTM, which have been tried.

IV. DATA PREPARATION AND PERFORMANCE EVALUATION

A. DATASET

We used a vehicular network dataset (the network type was 802.11 ad-hoc) [37] to measure short-range communications performance based on V2V and V2R communications in a highway. For data gathering, they put an external antenna on the roof of vehicles. The vehicle's longitude, latitude, speed, and heading were reported by GPS every two seconds. The data got from the highway in Atlanta has five usual lanes and one High Occupancy Vehicle (HOV) lane, which has been monitored between 2 pm and 5 pm. It reports the location's information with an accuracy of five to seven meters, and they got information of location via interpolation. In V2R communication, 1470 bytes of packets were broadcasted by the senders at a mean rate of about 150 packets/s. In V2V communication, the sender and receiver were installed on vehicles [48]. Both datasets based on V2V and V2R communication were used in our implementation. For the first phase, our focus was on network traffic prediction. We used a dataset based on V2R communication, and we predicted the network traffic flow based on receiving packets by the RSUs. Then, for the second phase, we used the dataset of V2V communication; and we targeted the "sender speed" to predict road traffic, while both sender and receiver vehicles were in the same lane (the second one from the right side of the road) but they had different distances. Finally, for the

third phase, since we aimed to predict the network traffic flow considering the effect of road traffic on it, we combined V2V and V2R datasets. After data collection, we did pre-processing, data cleansing and scaling using StandardScaler method which applied in order to scale and normalize the data. Then, using a machine learning algorithm, we extracted the effective parameters in a combined dataset of network traffic flow.

To implement the algorithms, we used Python version 3.6 [49], and for each phase, we used different libraries. For the first phase, different machine learning algorithms had been implemented. We used Scikit-learn, Pandas, NumPy, Matplotlib, Mlxtend, and some more libraries intending to deploy machine learning algorithms to find out the fittest algorithm for network traffic prediction. In consequence, we find that the RF had better performance. For phases two and three, which implemented deep learning algorithms, we used Keras libraries and TensorFlow [41] to implement the LSTM, Bi-LSTM and RF-GRU-NTP. After data collection, we did pre-process, data cleansing, and scaling to have more accurate prediction results.

B. PERFORMANCE EVALUATION OF THE NETWORK TRAFFIC FLOW PREDICTION

As mentioned previously, we used different metrics, such as confusion matrix, precision, recall, F1-Score, accuracy, ROC curves and Precision-Recall curve as classification metrics to evaluate the machine learning algorithms. For the deep learning part (phase two and three) we used the MSE, MAE, RMSE, and $R^2 - SCORE$ to evaluate the deep learning

algorithms. Moreover, for all the phases, we computed execution time to find the most suitable algorithm in our model.

We performed a confusion matrix as an evaluation metric in classification problems, which shows the different situations between true labels and the predicted ones [50], as shown in Figure 7.

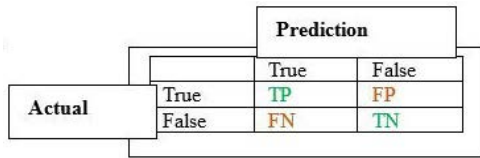


FIGURE 7. The structure of the confusion matrix.

where,

TP: When (Actual = 1, Predicted = 1)

FP: When (Actual = 0, Predicted = 1)

TN: When (Actual = 0, Predicted = 0)

FN: When (Actual = 1, Predicted = 0)

The true labels are presented in the rows representing the correct label and the predicted labels are placed in the column to determine the value that the classifiers have predicted. The number of times that true labels and predicted ones are matched would be presented in the diagonal.

In the first phase, our target was network traffic prediction based on packet receiving and we considered two classes for packet receiving in our dataset. Then, we labeled them: class 0 for NO (not receiving the packet) and class 1 for Yes (receiving the packet) as shown in (5). After that, we split the dataset into 75% for train and 25% for test and implemented four different classification algorithms. Then, we evaluated the algorithms by a confusion matrix. Figure 8 shows the confusion matrix of the RF, NB, KNN and SVM algorithms.

The confusion matrix of the RF indicates in 468 cases that the true label was 0 and it has been predicted correctly; and in 45 cases, the true label was 0, and it has been predicted wrong. Moreover, the prediction results for label 1 are shown in the next row. The NB’s confusion matrix indicates 433 correct prediction cases for label 0 and 734 true prediction cases for label 1. The confusion matrix of the SVM and KNN algorithms shows that the SVM has better prediction in label 1 with 734 true cases, and the KNN has better performance at predicting label 0 with 476 correct cases.

One of the common graphical metrics to present the result of classification problems is the Receiver Operator Characteristic (ROC) curves considering the Area Under the Curve (AUC). Moreover, another metric called the Precision-Recall (PR) curve can give us more valuable graphs to evaluate the performance of the algorithms [48]. By using the ROC curve, we clarify the accuracy of classifiers in prediction results.

The ROC curve will calculate the probability of predictions, and it shows the False Positive Rate (FPR) on the X-axis and True Positive Rate (TPR) on the Y-axis. The TPR can describe the model’s predictive performance in positive class in the situation that the true label is positive as well.

Figure 8 depicts the results of four different algorithms that we used for network traffic prediction. However, describing the prediction of positive class in the condition, where the true label is negative, is the responsibility of FPR. Moreover, there is a classifier with a random acting level shown in a red dotted line. It would separate the area into two parts that can improve the performance estimation in a way, where the curves that have better performance would be located above this line and close to the top-left corner, and the poor algorithms would be placed under the red dotted line.

Furthermore, the larger AUC indicates that the algorithms have better performance. As shown in Figure 9, the blue curve, which belongs to the KNN, and the orange curve that indicates the RF algorithm, are closer to the top left corner and they have larger AUC, which means they have higher performance, respectively. In addition, the green curve that represents the SVM and the red one that belongs to the NB algorithm, show the lower performance with 0.94 and 0.95 AUC consecutively.

However, the other evaluation metric for machine learning algorithms is the PR curve that presents a balance between precision indicated at the Y-axis and recall pointed out at the X-axis. Figure 10 illustrates the PR curve for different machine learning algorithms that we implemented. The blue dotted horizontal line demonstrates a “baseline” classifier, where the ideal classifier is above this line and the closest to the right top corner. Thus, the lowest performance would belong to the curves that are closer to the baseline. As depicted in Figure 10, the KNN and RF algorithms, shown in orange and green, respectively, have the highest performance; and the NB and SVM, shown in purple and red, have the lowest performance accordingly.

We tried classification reports and other metrics like precision, recall and F1-score to get more insight into our model performance. Table 2 shows the evaluation results of each algorithm. Therefore, the KNN algorithm has the highest accuracy with 96%, and the SVM has the lowest one with 91% in our prediction model, while the RF and NB give the same percentage of accuracy with 93%.

After the performance evaluation that we have done, the results of the KNN and RF was almost the best. However, the other factor that should be considered in performance evaluation is the execution time to find the best algorithm for network traffic prediction, specifically when the volume of data is large. Table 3 present the execution time for each algorithm based on the results. The KNN is the most time-consuming algorithm, and the NB took less time. Therefore, based on all the results that we have got, we can conclude that the RF is the perfect match for our model with network traffic prediction, because it is not as time consuming as KNN. On the other hand, its performance is good enough in other evaluations.

For the second phase, our target was road traffic prediction based on “sender speed.” We deployed three different deep learning algorithms; then, we evaluated their performances to find the fittest algorithm for our model. The MAE and the

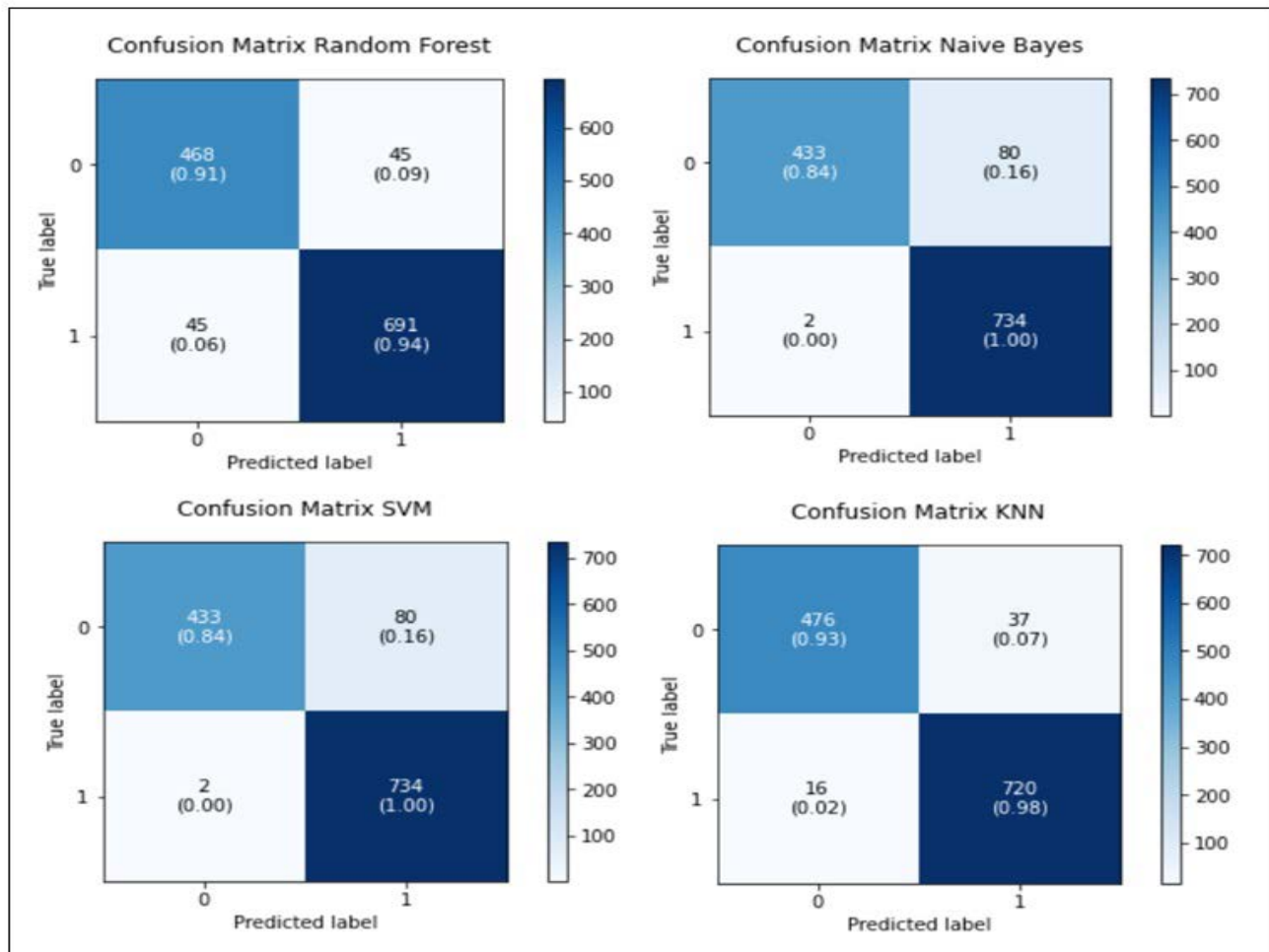


FIGURE 8. Confusion matrices of the RF, NB, SVM and KNN.

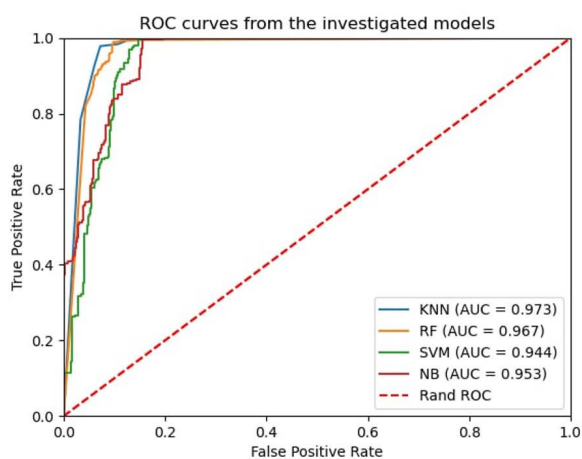


FIGURE 9. ROC curve for KNN, RF, SVM and NB.

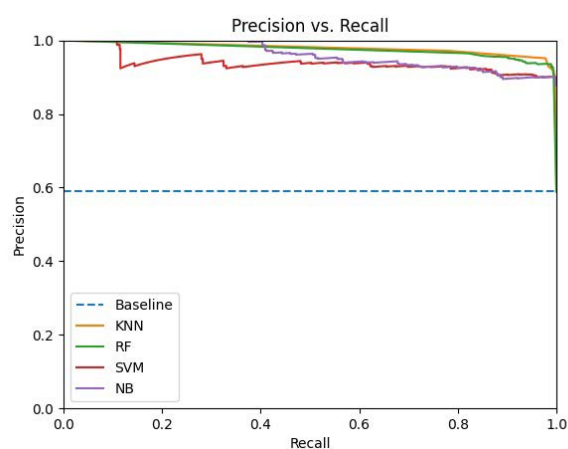


FIGURE 10. PR curve for the KNN, RF, SVM and NB in network traffic prediction.

RMSE are two standard evaluation metrics to represent the average performance of the model [51].

However, we calculated the MSE as a metric for evaluating the prediction results and the $R^2 - SCORE$ that is the most important metric for our model, because it can indicate how

the predicted model is good in future observation which the higher value of it, indicates a better prediction model, and it means the difference between the actual and predicted value is insignificant. Finally, we considered the time that

TABLE 2. Classification reports.

Algorithm	Accuracy (%)	Label (packet receiving)	Precision	Recall	F1-score
KNN	96	No	0.97	0.91	0.94
		Yes	0.94	0.98	0.96
RF	93	No	0.94	0.91	0.92
		Yes	0.93	0.96	0.95
SVM	91	No	0.99	0.85	0.91
		Yes	0.90	0.99	0.95
NB	93	No	0.99	0.85	0.91
		Yes	0.90	0.99	0.95

TABLE 3. Execution time for the RF, NB, SVM, KNN.

Algorithm	Execution time(s)
RF	0.1403975
NB	0.819984
SVM	0.1957004
KNN	0.7682998

each of the deep learning algorithms took for fitting the model. The results of calculating the MAE for all algorithms are illustrated in Figure 11, where it shows that the LSTM experienced the highest error at the beginning. On the other hand, the GRU has the lowest one, and it has less iteration to get the best result, however, the LSTM and Bi-LSTM have more epochs, respectively.

By considering the growing number of vehicles and facing a large amount of data, execution time is a vital parameter for choosing our model’s fittest algorithm. Thus, we calculated the time that each algorithm took for fitting the model, and as illustrated in Table 4, the GRU is the fastest one with 1 minute and 3 seconds in model fitting, and the Bi-LSTM and LSTM are the slowest, respectively.

TABLE 4. Duration time for GRU, Bi-LSTM & LSTM.

Algorithm	Duration time (m)	Epochs
GRU	1:03.246713	Epoch 49/120
Bi-LSTM	03:24.082677	Epoch 56/120
LSTM	05:27.551359	Epoch 53/120

Then, we predicted the sender speed with the aim of road traffic prediction by implementing the GRU, LSTM, and Bi-LSTM algorithms. As shown in Figure 12, the blue line indicates the test data and the prediction that different algorithms have done is in the orange line. The results show that the GRU has performed better than the LSTM and Bi-LSTM. Thus, we can see that the difference between test and predicted data is insignificant.

Moreover, we calculated some evaluation metrics like the MSE, MAE, RMSE and $R^2 - SCORE$ for all three algorithms for the final evaluation step. As demonstrated in Table 5,

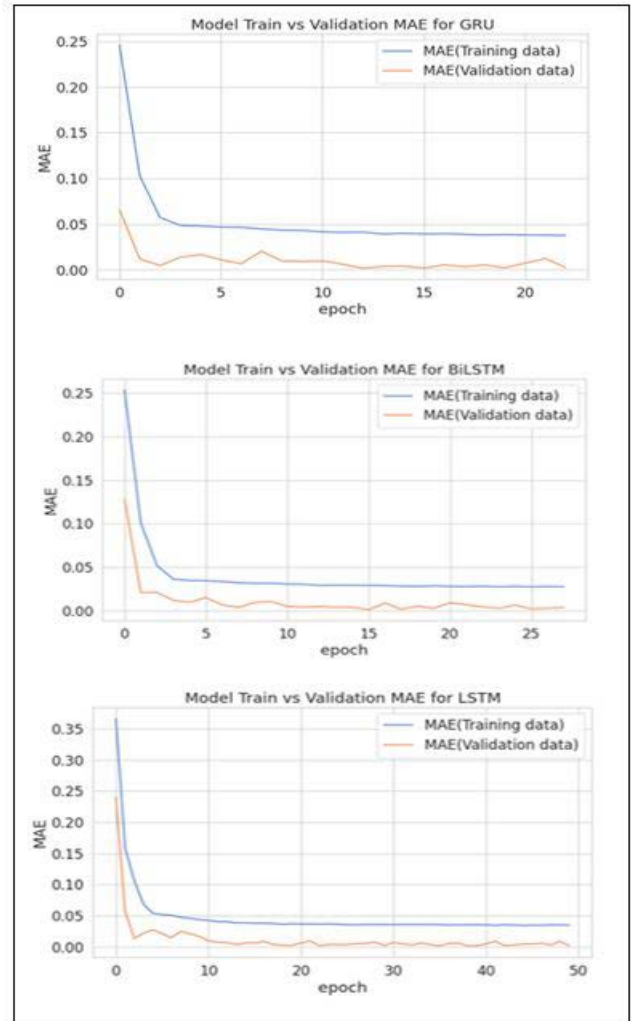


FIGURE 11. The MAE for GRU, LSTM and Bi-LSTM (phase2).

we got the best results in all metrics with the GRU algorithm. Furthermore, the highest $R^2 - SCORE$, the most crucial factor for evaluating our model, was 0.995, which belonged to the GRU algorithm.

TABLE 5. Evaluation metrics for Bi-LSTM, GRU and LSTM.

Algorithm	MSE	MAE	RMSE	$R^2 - SCORE$
Bi-LSTM	2.062	0.98	1.436	0.993
GRU	1.402	0.416	1.184	0.995
LSTM	1.975	0.984	1.405	0.993

Based on all evaluation ways done, the GRU is the best algorithm that can fit our model in the lowest time and error, and the highest $R^2 - SCORE$.

Finally, we predicted network traffic flow considering road traffic flow by implementing machine learning and deep learning algorithms for the third phase. By deploying the RF algorithm, we detected the most effective features on “sender speed.” Figure 13 shows that the “receiver speed”

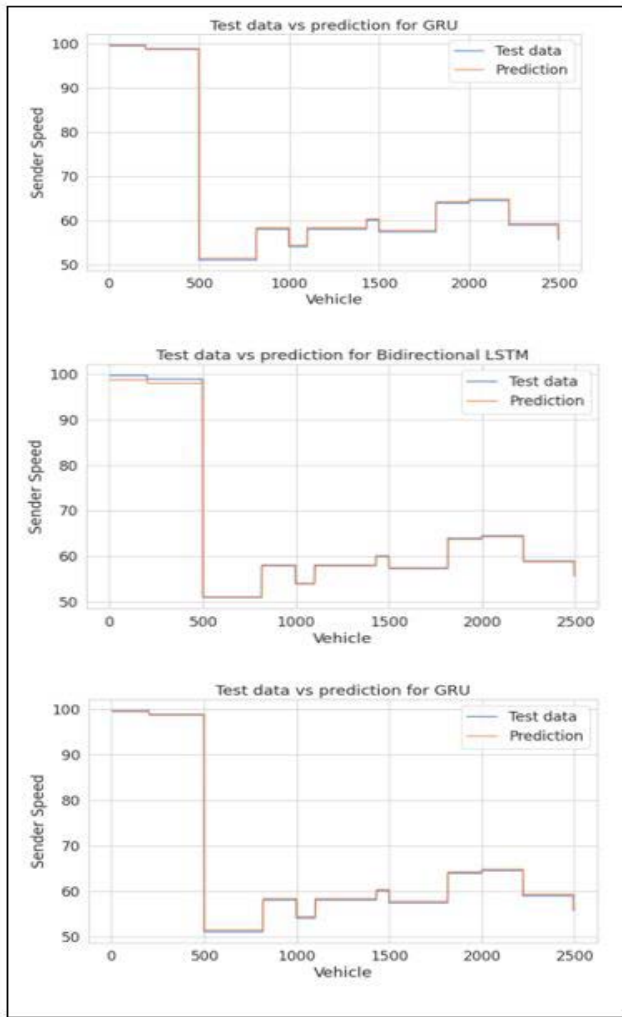


FIGURE 12. Test data vs prediction for the LSTM, GRU and Bi-LSTM (phase2).

and “packet receiving” are the most effective parameters on “sender speed,” which we consider as a network traffic parameter.

The next step is passing the most influential parameters on network traffic as input variables into the GRU algorithm with the aim of network traffic prediction and implementing the proposed RF-GRU-NTP model. Figure 14 shows the MAE for the LSTM, RF-GRU-NTP and Bi-LSTM algorithms. As depicted, the LSTM experienced the highest error at the beginning and Bi-LSTM had the most iteration while the proposed RF-GRU-NTP had better performance in less than 25 iterations.

We predicted network traffic flow considering road traffic parameters and evaluated the performance of our model using four different evaluation metrics as the MAE, MSE, RMSE, and $R^2 - SCORE$. Table 6 shows the value of these evaluation metrics for each algorithm, demonstrating that the proposed RF-GRU-NTP model has the lowest error and the highest $R^2 - SCORE$, which means it has better performance in comparison with the LSTM and Bi-LSTM.

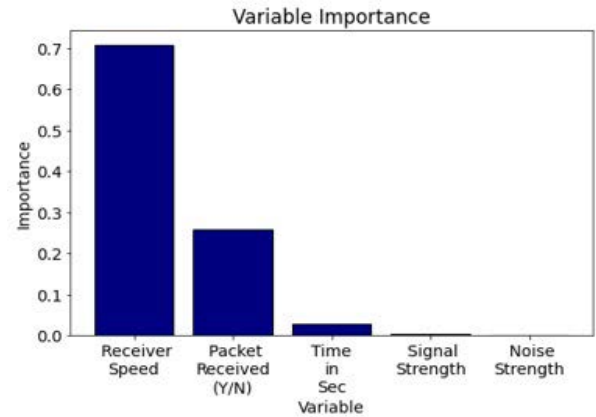


FIGURE 13. Effective parameters in network traffic.

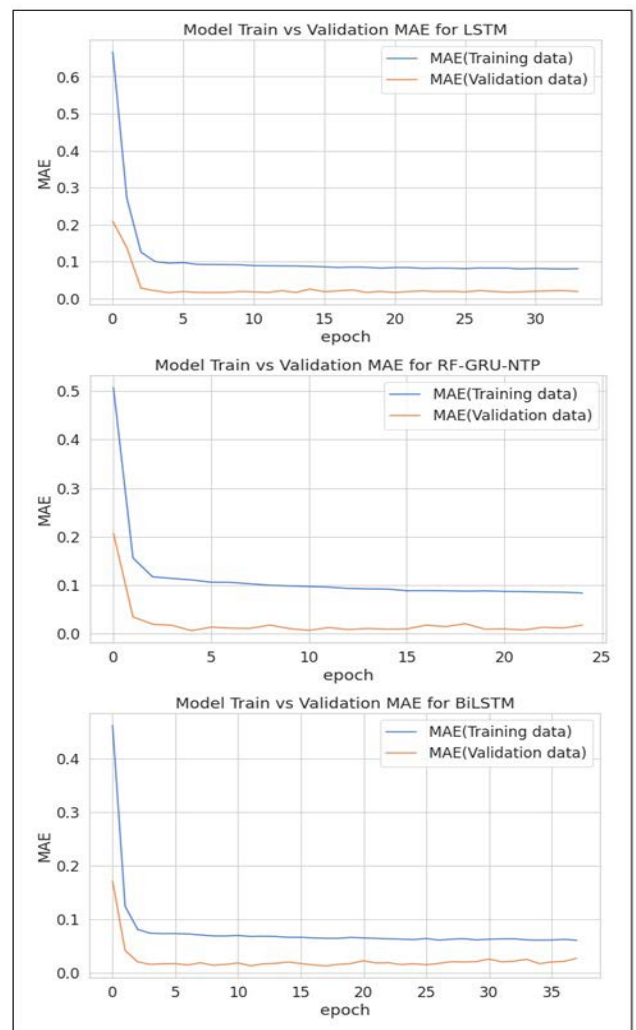


FIGURE 14. The MAE for GRU, LSTM and Bi-LSTM (phase3).

Figure 15 depicts the prediction results for sender speed considering effective parameters on it in order to network traffic prediction. The blue line indicates the test data and the prediction that has been done by models is in orange line. As shown, the proposed RF-GRU-NTP model performed

TABLE 6. Evaluation metrics for Bi-LSTM, RF-GRU-NTP and LSTM.

Algorithm	MSE	MAE	RMSE	$R^2 - SCORE$
Bi-LSTM	1.911	0.711	1.382	0.993
RF-GRU-NTP	1.437	0.372	1.199	0.995
LSTM	1.677	0.427	1.295	0.994

TABLE 7. Duration time for RF-GRU-NTP, Bi-LSTM and LSTM.

Algorithm	Duration time (m)
RF-GRU-NTP	0:33.302000
Bi-LSTM	02:06.236182
LSTM	04:09.028719

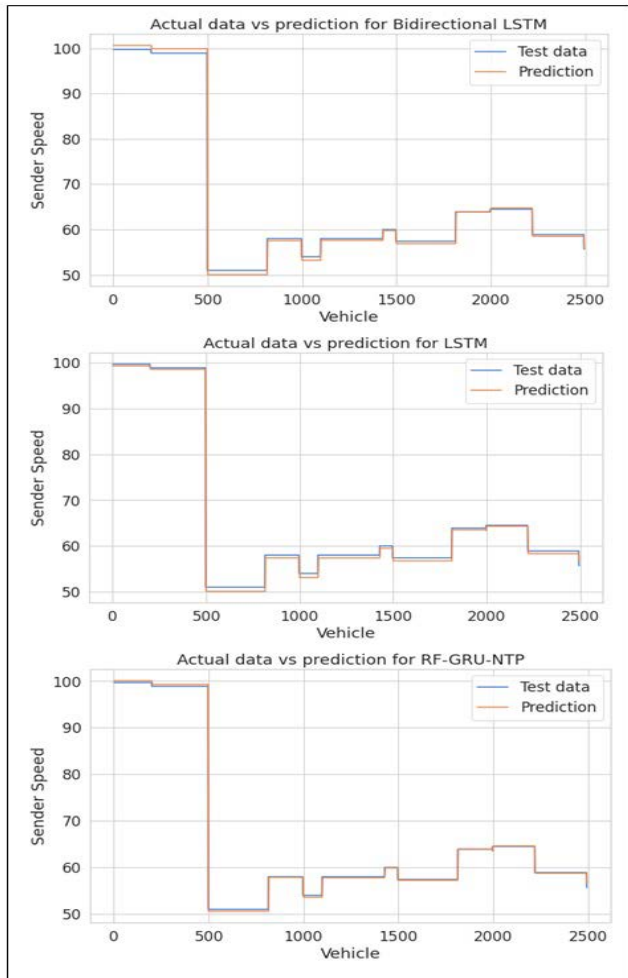


FIGURE 15. Test data vs prediction for the LSTM, RF-GRU-NTP and Bi-LSTM (phase 3).

better than other pure algorithm that we implemented, and the difference between test and predicted data is insignificant while the highest difference between predicted and actual value belongs to the Bi-LSTM algorithm.

After calculating the evaluation metrics, we computed the time that each algorithm took for fitting the model. Moreover, we set an early stop method to stop training when the algorithm has not improved, considering 20 for patience's value. As shown in Table 7, whereas the RF-GRU-NTP algorithm could fit the model in less than 34 seconds, the Bi-LSTM and LSTM needed about 2 minutes and 4 minutes, respectively.

Figure 16 shows the difference in fitting time between the proposed RF-GRU-NTP model and the pure algorithms that

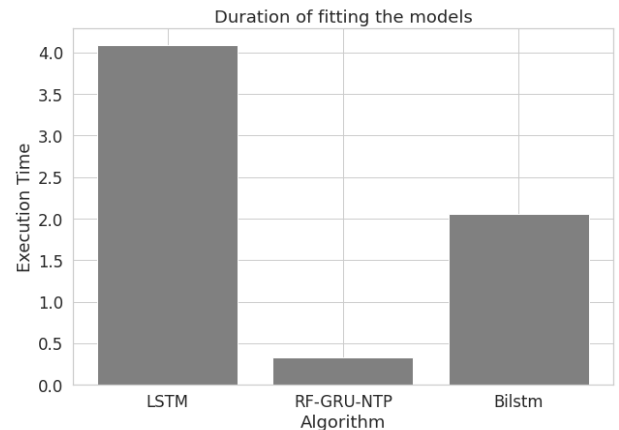


FIGURE 16. Time for fitting the LSTM, RF-GRU-NTP and Bi-LSTM (phase3).

we implement them. The proposed RF-GRU-NTP model has significant different fitting time form other algorithms.

Consequently, based on all results that we got, the proposed RF-GRU-NTP model has the best performance in network traffic prediction considering road traffic parameters.

V. CONCLUSION

In this paper, we proposed an RF-GRU-NTP model with the aim of network traffic flow prediction based on the traffic in the road and network simultaneously. We divided our research into three phases. In the first phase, we focused on network traffic prediction. We used the V2R dataset and considered the receiving packets sent by vehicles to the RSUs as a network parameter to predict network traffic flow. Then, we tried different machine learning algorithms like the RF, NB, KNN, and SVM algorithms, and we evaluated them using some classification metrics. After all evaluations, the RF has the better performance to predict network traffic flow while our target was “packet receiving.”

In the second phase, we tried to predict the road traffic flow using the V2V dataset while our target was “sender speed” to define the road traffic. We assumed that the traffic would happen on the road if the senders' speed were less than 60 Km/h. Therefore, we implemented different deep learning algorithms, including the LSTM, GRU, and Bi-LSTM. Finally, we evaluated the results using some regression evaluation metrics, which, based on the results we got, the GRU was the fittest algorithm for road traffic prediction.

Then at the third phase, we implemented our target, which is network traffic flow considering road traffic flow, by combining machine learning and deep learning algorithms.

For this purpose, we combined V2V and V2R datasets, and used the RF algorithm for feature selection. We found the most important features, which were “packet receive” and “receiver speed” that can affect “sender speed” and the network traffic flow. Then by implementing the proposed RF-GRU-NTP model, we predicted network traffic flow.

Therefore, we compared our results with a pure algorithm like LSTM and Bi-LSTM to make sure that the proposed model has good results in network traffic flow prediction.

The main complexity of the proposed model was combining two datasets in order to implementing machine learning and deep learning algorithms with the aim of network traffic prediction considering different types of parameters. To the best of our knowledge, this is the first research that predicts the network traffic flow based on road traffic flow.

However, by growing up the number of vehicles, the volume of produced data by them would take shape of big data which in our future work we will implement our proposed model in big data.

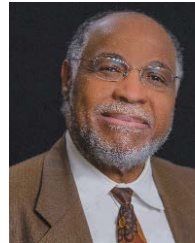
REFERENCES

- [1] N. Taherkhani and S. Pierre, “Improving dynamic and distributed congestion control in vehicular ad hoc networks,” *Ad Hoc Netw.*, vol. 33, pp. 112–125, Oct. 2015.
- [2] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, “A hybrid deep learning based traffic flow prediction method and its understanding,” *Transp. Res. C, Emerg. Technol.*, vol. 90, pp. 166–180, May 2018.
- [3] H.-F. Yang, T. S. Dillon, and Y.-P. P. Chen, “Optimized structure of the traffic flow forecasting model with a deep learning approach,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2371–2381, Oct. 2016.
- [4] W. Tong, A. Hussain, W. X. Bo, and S. Maharjan, “Artificial intelligence for vehicle-to-everything: A survey,” *IEEE Access*, vol. 7, pp. 10823–10843, 2019.
- [5] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” Dec. 2014, *arXiv:1412.3555*.
- [6] A. Karpathy, J. Johnson, and L. Fei-Fei, “Visualizing and understanding recurrent networks,” Jun. 2015, *arXiv:1506.02078*.
- [7] M. Coşkun, Ö. Yildirim, U. Ayşegül, and Y. Demir, “An overview of popular deep learning methods,” *Eur. J. Techn.*, vol. 7, no. 2, pp. 165–176, Dec. 2017.
- [8] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, “LSTM: A search space odyssey,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2016.
- [9] R. Jozefowicz, W. Zaremba, and I. Sutskever, “An empirical exploration of recurrent network architectures,” in *Proc. Int. Conf. Int. Conf. Mach. Learn.*, Jun. 2015, pp. 2342–2350.
- [10] P. Sun, A. Boukerche, and Y. Tao, “SSGRU: A novel hybrid stacked GRU-based traffic volume prediction approach in a road network,” *Comput. Commun.*, vol. 160, pp. 502–511, Jul. 2020.
- [11] P. T. Yamak, L. Yujian, and P. K. Gadosey, “A comparison between ARIMA, LSTM, and GRU for time series forecasting,” in *Proc. 2nd Int. Conf. Algorithms, Comput. Artif. Intell.*, Sanya, China, Dec. 2019, pp. 49–55.
- [12] Z. Cui, R. Ke, Z. Pu, and Y. Wang, “Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction,” Jan. 2018, *arXiv:1801.02143*.
- [13] R. Jia, P. Jiang, L. Liu, L. Cui, and Y. Shi, “Data driven congestion trends prediction of urban transportation,” *IEEE Internet Things J.*, vol. 5, no. 2, pp. 581–591, Apr. 2018.
- [14] F. Falahatrafar, S. Pierre, and S. Chamberland, “A centralized and dynamic network congestion classification approach for heterogeneous vehicular networks,” *IEEE Access*, vol. 9, pp. 122284–122298, 2021.
- [15] A. Lazaris and V. K. Prasanna, “Deep learning models for aggregated network traffic prediction,” in *Proc. 15th Int. Conf. Netw. Service Manage. (CNSM)*, Halifax, NS, Canada, Oct. 2019, pp. 1–5.
- [16] S. Troia, R. Alvizu, Y. Zhou, G. Maier, and A. Pattavina, “Deep learning-based traffic prediction for network optimization,” in *Proc. 20th Int. Conf. Transparent Opt. Netw. (ICTON)*, Bucharest, Romania, Jul. 2018, pp. 1–4.
- [17] T. H. H. Aldhyani, M. Alrasheedi, A. A. Alqarni, M. Y. Alzahrani, and A. M. Bamhdi, “Intelligent hybrid model to enhance time series models for predicting network traffic,” *IEEE Access*, vol. 8, pp. 130431–130451, 2020.
- [18] R. Vinayakumar, K. P. Soman, and P. Poornachandran, “Applying convolutional neural network for network intrusion detection,” in *Proc. Int. Conf. Adv. Comput., Commun. Inform. (ICACCI)*, Udupi, India, Sep. 2017, pp. 1222–1228.
- [19] N. Stepanov, D. Alekseeva, A. Ometov, and E. S. Lohan, “Applying machine learning to LTE traffic prediction: Comparison of bagging, random forest, and SVM,” in *Proc. 12th Int. Congr. Ultra Modern Telecommun. Control Syst. Workshops (ICUMT)*, Brno, Czech Republic, Oct. 2020, pp. 119–123.
- [20] Y. Kamarianakis, W. Shen, and L. Wynter, “Real-time road traffic forecasting using regime-switching space-time models and adaptive LASSO,” *Appl. Stochastic Models Bus. Ind.*, vol. 28, no. 4, pp. 297–315, Jul. 2012.
- [21] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, “Traffic flow prediction with big data: A deep learning approach,” *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Sep. 2014.
- [22] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi, “DNN-based prediction model for spatio-temporal data,” in *Proc. 24th ACM SIGSPATIAL Int. Conf. Adv. Geograph. Inf. Syst.*, Burlingame, CA, USA, Oct. 2016, pp. 1–4.
- [23] G. Dai, C. Ma, and X. Xu, “Short-term traffic flow prediction method for urban road sections based on space–time analysis and GRU,” *IEEE Access*, vol. 7, pp. 143025–143035, 2019.
- [24] Z. Duan, Y. Yang, K. Zhang, Y. Ni, and S. Bajgain, “Improved deep hybrid networks for urban traffic flow prediction using trajectory data,” *IEEE Access*, vol. 6, pp. 31820–31827, 2018.
- [25] L. Zhang, N. R. Alharbe, G. Luo, Z. Yao, and Y. Li, “A hybrid forecasting framework based on support vector regression with a modified genetic algorithm and a random forest for traffic flow prediction,” *Tsinghua Sci. Technol.*, vol. 23, no. 4, pp. 479–492, Aug. 2018.
- [26] J.-Y. Yang, L.-D. Chou, Y.-C. Li, Y.-H. Lin, S.-M. Huang, G. Tseng, T.-W. Wang, and S.-P. Lu, “Prediction of short-term average vehicular velocity considering weather factors in urban VANET environments,” in *Proc. Int. Conf. Mach. Learn. Cybern.*, Qingdao, China, Jul. 2010, pp. 3039–3043.
- [27] F. Li, W. Chen, and Y. Shui, “Study on connectivity probability of VANETs under adverse weather conditions at 5.9 GHz,” *IEEE Access*, vol. 8, pp. 547–555, 2020.
- [28] X. Bao, D. Jiang, X. Yang, and H. Wang, “An improved deep belief network for traffic prediction considering weather factors,” *Alexandria Eng. J.*, vol. 60, no. 1, pp. 413–420, Feb. 2021.
- [29] F. I. Rahman, “Short term traffic flow prediction using machine learning-KNN, SVM and ANN with weather information,” *Int. J. Traffic Transp. Eng.*, vol. 10, no. 3, pp. 413–420, Jun. 2020.
- [30] Y. Bie, T. Z. Qiu, C. Zhang, and C. Zhang, “Introducing weather factor modelling into macro traffic state prediction,” *J. Adv. Transp.*, vol. 2017, pp. 1–15, Feb. 2017.
- [31] N. Julio, R. Giesen, and P. Lizana, “Real-time prediction of bus travel speeds using traffic shockwaves and machine learning algorithms,” *Res. Transp. Econ.*, vol. 59, pp. 250–257, Nov. 2016.
- [32] P. Cunningham and S. J. Delany, “*k*-Nearest neighbour classifiers: 2nd edition (with Python examples),” 2020, *arXiv:2004.04523*.
- [33] M. Pal, “Random forest classifier for remote sensing classification,” *Int. J. Remote Sens.*, vol. 26, no. 1, pp. 217–222, Nov. 2005.
- [34] J. Ren, S. D. Lee, X. Chen, B. Kao, R. Cheng, and D. Cheung, “Naive Bayes classification of uncertain data,” in *Proc. 9th IEEE Int. Conf. Data Mining*, Miami Beach, FL, USA, Dec. 2009, pp. 944–949.
- [35] G. M. Foody and A. Mathur, “Toward intelligent training of supervised image classifications: Directing training data acquisition for SVM classification,” *Remote Sens. Environ.*, vol. 93, nos. 1–2, pp. 107–117, Oct. 2004.
- [36] N. Donges, *A Complete Guide to the Random Forest Algorithm*. Accessed: Jul. 22, 2021. [Online]. Available: <https://builtin.com/data-science/random-forest-algorithm>
- [37] R. M. Fujimoto, R. Guensler, M. P. Hunter, H. Wu, M. Palekar, J. Lee, and J. Ko, *CRAWDAD Dataset Gatech/Vehicular (V. 2006 03 15)*, Mar. 2006. [Online]. Available: <https://crawdad.org/gatech/vehicular/20060315>, doi: 10.15783/C74S3Z.

- [38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Nov. 2011.
- [39] S. Kostadinov. *Understanding GRU Networks*. Accessed: Dec. 16, 2017. [Online]. Available: <https://towardsdatascience.com/understanding-gru-networks-2ef37df6c9be>
- [40] S. G. K. Patro and K. K. Sahu, "Normalization: A preprocessing stage," Mar. 2015, *arXiv:1503.06462*.
- [41] A. Géron, *Hands-On Machine Learning With Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. Sebastopol, CA, USA: O'Reilly Media, Sep. 2019.
- [42] Q. V. Le, J. Ngiam, A. Coates, A. Lahiri, B. Prochnow, and A. Y. Ng, "On optimization methods for deep learning," in *Proc. ICML*, Jan. 2004, pp. 1–8.
- [43] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, no. 7, pp. 2121–2159, Jul. 2011.
- [44] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," Dec. 2014, *arXiv:1412.6980*.
- [45] T. Tieleman and G. Hinton, "Lecture 6.5-RMSPROP: Divide the gradient by a running average of its recent magnitude," *COURSERA, Neural Netw. Mach. Learn.*, vol. 4, no. 2, pp. 26–31, Oct. 2012.
- [46] A. Botchkarev, "Performance metrics (error measures) in machine learning regression, forecasting and prognostics: Properties and typology," Sep. 2018, *arXiv:1809.03006*.
- [47] Y. Xuan, W. Si, J. Zhu, Z. Sun, J. Zhao, M. Xu, and S. Xu, "Multi-model fusion short-term load forecasting based on random forest feature selection and hybrid neural network," *IEEE Access*, vol. 9, pp. 69002–69009, 2021.
- [48] H. Wu, M. Palekar, R. Fujimoto, R. Guensler, M. Hunter, J. Lee, and J. Ko, "An empirical study of short range communications for vehicles," in *Proc. 2nd ACM Int. Workshop Veh. Ad Hoc Netw.*, 2005, pp. 83–84.
- [49] G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA, USA: CreateSpace, 2009.
- [50] J. Davis and M. Goadrich, "The relationship between precision-recall and ROC curves," in *Proc. 23rd Int. Conf. Mach. Learn.*, Pittsburgh, PA, USA, Jun. 2006, pp. 233–240.
- [51] C. J. Willmott and K. Matsuura, "Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance," *Climate Res.*, vol. 30, no. 1, pp. 79–82, Dec. 2005.



SANAZ SHAKER SEPASGOZAR received the B.Sc. degree in computer software engineering from Islamic Azad University Central Tehran Branch, Tehran, Iran, in 2010, and the M.Sc. degree in IT management–information resource from Islamic Azad University Science and Research Branch, Tehran, in 2017. She is currently pursuing the Ph.D. degree in computer engineering with Polytechnique Montreal, Montreal, QC, Canada. Her research interests include traffic flow prediction in the road and vehicular ad hoc networks using deep learning and machine learning algorithms.



SAMUEL PIERRE (Senior Member, IEEE) received the Ph.D. degree (Hons.) from the Université du Québec à Trois-Rivières (UQTR), in May 2014, and the Ph.D. degree (Hons.) from the Université du Québec en Outaouais (UQO), in November 2016. He is currently a Full Professor with the Department of Computer and Software Engineering, Polytechnique Montreal, and the Director of the Mobile Computing and Networking Research Laboratory (LARIM). He has authored or coauthored more than 550 technical publications, including articles in refereed archival journals, textbooks, patents, and book chapters. His research interests include wired and wireless communications, mobile computing and networking, cloud computing, and e-learning. He is a fellow of the Engineering Institute of Canada, in 2003, and the Canadian Academy of Engineering, in 2008. He was appointed as a member of the Order of Canada, in December 2011. He has received several awards, including the Prix Poly 1873 for Excellence in Teaching and Training in 2001 and 2005, and the Knight of the National Order of Quebec in 2009. In 2017, he received the El Fasi Prize from the Agence Universitaire de la Francophonie (AUF) to highlight the action of a person who has exerted a significant influence through the quality of his expertise and the innovative nature of his achievements at the international level in the fields of research, training, development and international cooperation, governance and/or transfer of knowledge or skills. In 2020, he received the Grand Prize for Professional Excellence from the Order of Engineers of Quebec (OIQ). In 2021, he received the Gold Medal from Engineers Canada.

• • •