IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 8 February 2023, accepted 2 March 2023, date of publication 7 March 2023, date of current version 10 March 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3253625

RESEARCH ARTICLE

An Ensemble-Based Machine Learning Model for Forecasting Network Traffic in VANET

PARVIN AHMADI DOVAL AMIRI AND SAMUEL PIERRE, (Senior Member, IEEE)
Mobile Computing and Networking Research Laboratory (LARIM), Department of Computer and Software Engineering, Polytechnique Montréal, Montreal, QC H3T 1J4, Canada

Corresponding author: Parvin Ahmadi Doval Amiri (parvin-2.ahmadi-doval-amiri@polymtl.ca)

ABSTRACT Vehicular Ad-hoc Networks (VANETs), as the most significant element of the Intelligent Transportation Systems (ITS), have the potential to enhance traffic efficiency and road safety by making the transportation system smarter and are still at the initial point of development. In this paper, we propose an ensemble-based machine learning model for network traffic prediction in VANET. We take advantage of Ensemble Learning (EL), which combines different Machine Learning (ML) models to achieve better performance and improve accuracy. We consider the most informative attributes of the VANET dataset using Boruta and LightGBM as ensemble feature selection methods. Our proposed model is based on Stacking Ensemble Learning with Booster Model (STK–EBM) designed with a stacking ensemble of heterogeneous ML models. The framework of the proposed model consists of two layers, including a base layer and a meta layer. The first layer integrates Random Forest (RF), K-Nearest Neighbor (KNN) and XGBoost as a booster of the base learners. An optimized Logistic Regression (LR) employs as our meta learner in the second layer. We evaluate the performance of our model considering classification metrics and then compare it with the most popular traffic predictive models. Simulation results show that the STK–EBM model gives a more stable prediction than the single algorithm, as well as better overall performance in terms of prediction accuracy and execution time.

INDEX TERMS Vehicular ad-hoc network, network traffic prediction, classification, machine learning, deep learning, ensemble learning.

I. INTRODUCTION

Currently, people have become more dependent on transportation, and the number of road users has significantly increased. This growth leads to multiple problems in terms of air pollution, fuel consumption and costs by wasting the time of drivers in traffic and various losses due to road accidents. An efficient way to enhance road safety and tackle these various losses is by taking advantage of technology to raise the awareness of drivers, especially about the probability of an accident or congestion on the road. Intelligent Transportation System (ITS) uses information, communication, and control technology to manage transportation networks. Vehicular Ad-hoc Networks (VANETs) are considered the most significant elements of ITS to enhance traffic efficiency and road safety [1]. The basic architecture of VANETs includes Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication [2]. These communications in VANETs employ the Dedicated Short-Range Communication (DSRC) and Wireless Access in Vehicular Environment (WAVE) (or IEEE 802.11p) standards [3]. VANET application uses the data collected from the mentioned communication that can produce valuable and shareable knowledge among vehicular users on the road to raise the awareness of drivers to make an informed decision, especially to prevent accidents and traffic congestion. Therefore, it brings safety, efficiency, reliability, comfort and convenience [4]. However, it imposes various service requirements, such as network performance, which is highly important with regard to VANET safety applications.

In the case of a time-sensitive application that relates to human life, not only time but also data accuracy and reliability are critical [4], [5]. Consequently, the existence of traffic data and easily reachable tools for analyzing these data and extracting knowledge pave the way for researchers to enhance the requirements of networks connected to road

The associate editor coordinating the review of this manuscript and approving it for publication was Ines Domingues.

users' lives. Ultimately, researchers can make the transportation system smarter. They enhance road safety and decrease crashes and other losses related to traffic by developing new driving rules, policies and predictive models based on Artificial Intelligence (AI) that optimizes traditional data-driven approaches. Therefore, vehicular networks are in the early stages of challenges related to the exploitation and adaptation of AI tools [6]. Moreover, the implementation of Machine Learning (ML) techniques as a subset of AI could optimize the operation of the networks in predicting failure before it causes a significant reduction in the Quality of Service (QoS) [7]. In VANETs, information about road conditions and other vehicles will be exchanged among communications when the number of sending and receiving packets through these communications increases (i.e., many vehicular users on the road) and traffic occurs in the network, which will cause a delay or decline in important services. Accordingly, an efficient prediction of traffic in the network can help enhance the QoS, accuracy, reliability and time for road users. It can improve whole driver-vehicle-road performance [8]. The question is how can we propose an efficient network traffic prediction model using AI?

ML, as a major part of AI, can be categorized into three parts: supervised learning, unsupervised learning and reinforcement learning. Moreover, transfer learning, online learning and Q-learning can be considered subclasses of these three main learning models [6], [9]. DL is closely related to the three mentioned classes of the ML model. It is a deeper network of neurons in multiple layers that is used for traffic prediction in a large and complex dataset [6]. In this study, we focused on supervised learning, in which training data are based on labeled data. Moreover, supervised learning can be designed as a classification and regression task [9]. We considered our problem as a classification task. Furthermore, supervised ML algorithms, including Random Forest(RF), K-Nearest Neighbor(KNN), Naive Bayes (NB), Decision Tree(DT) and Support Vector Machines (SVM), are commonly considered for designing predictive models in traffic [7]. Among them, Support Vector Machines (SVM), which can adapt to the dynamic and nonlinear nature of traffic data, have problems with selecting the kernel type and resolving this issue. The optimized SVM is an adaptive model for forecasting traffic and fitting times [8]. Moreover, Neural Network(NN) is a simple DL model mentioned as a well-chosen problem solution in increasing the accuracy of traffic flow prediction [10]. However, due to the limitations and drawbacks of single ML models in traffic prediction, Ensemble Learning(EL) has become popular and used in various domains, including health, finance, and energy [11]. The abovementioned studies on VANET application issues, AI limitations and advantages can work together to match the traffic problem in VANETs around the best AI solution. In summary, VANET applications impose some service requirements, such as network performance, that are highly important specifically for safety applications that relate to human life [8]. When traffic occurs in the network, it causes a significant reduction in QoS and failure in the network. ML, as a subset of AI, can optimize the operation of networks [7]. Making an intelligent prediction of traffic in the network can help us to precisely identify the failure of the network and avoid service degradation, especially for services that are highly dependent on the performance of the network. However, each ML model has limitations and drawbacks for traffic prediction. In this case, ensemble learning can significantly improve the performance of machine learning in most problems [11]. Ensemble learning can achieve better performance with enhanced stability and generalization ability in addition to higher prediction accuracy and fast computation than a single ML [12]. We need to identify the best proper strategy to integrate ML models through EL. This motivated us to use EL in an efficient way for network traffic prediction to maintain the QoS and performance of the network in VANET applications.

In this paper, we propose a network traffic prediction model using ensemble learning and integrating various ML approaches. We identify the best strategy to integrate ML methods through ensemble learning in an efficient way with the aim of providing a balanced result in addition to improving the overall performance.

The originality of our model lies in proposing an efficient ensemble of ML models to predict traffic with the aim of obtaining more stable and accurate prediction results. We consider VANET communication from the integration of V2V and V2I data. In addition, the most informative features are built from the extracted datasets using LightGBM and Boruta as the feature selection approaches. Using this approach, we can maintain the quality of input data that will be highly important for efficient prediction. Therefore, the main objective of our proposed model is to design an efficient network traffic prediction model for VANETs. The detailed contents of this paper can be summarized as follows:

- Compare the top popular ML models for traffic prediction, including RF, KNN, NB, DT, SVM and MLP, as simple Deep Learning (DL) models to identify which one results in better performance and to realize the effective incorporation of ML models that bring more accuracy and adaptability to the dynamic nature of traffic, then use them as the base learners in the first layer of our proposed model.
- Propose a hybrid stacking ensemble model to predict traffic to obtain a more stable prediction and overcome the inherent weakness of every single model. The proposed model has two layers, including the base layer and meta layer, in which the most effective combination of ML models was selected. The first layer integrates RF, KNN and XGBoost as a booster of base learners, and an optimized Logistic Regression (LR) is employed as our meta-learner in the second layer.
- Evaluate the performance of the proposed model according to the classification metrics that can effectively assess the prediction results.

Eventually, we develop an efficient AI solution for network traffic prediction to prevent service degradation in VANET applications. The proposed model considers highly important points for being a more beneficial and powerful model, such as maintaining the quality of data, a heterogeneous integration strategy with reducing complexity and covering single ML model problems, in addition to taking advantage of the most effective models. Therefore, it can provide a more adaptable and stable prediction model with better overall performance in terms of accuracy, prediction error and execution time.

The rest of this paper is organized as follows. Section II discusses the related work. Section III describes the proposed methodology based on ensemble learning for network traffic prediction, and our results are presented in Section IV. Section V concludes the paper.

II. BACKGROUND AND RELATED WORK

Currently, the onward movement in the field of communication and computing systems gives researchers the opportunity of a new way of solving problems related to intelligent traffic to enhance traffic efficiency and road safety. Several researchers have used AI to optimize traditional data-driven approaches. The various branches of AI will be able to bring out an optimized solution that will not cause or generate more problems. Vehicular networks are in the early stages of challenges relevant to the exploitation and adaptation of AI tools [6]. Sultan et al. [2] indicated VANET architecture and applications. They discussed V2V and V2I communication that make possible the advancement of many applications. Faezipour et al. [13] discussed this communication and related challenges, as well as available solutions in intelligent Vehicle Area Networks as a future transportation system. Therefore, VANETs offer several applications, including safety and no-safety, using technologies to provide operative traffic management in vehicular networks. However, these services have diverse requirements in VANETs, such as network performance, which plays a significant role, notably for VANET safety applications. In this matter, the outcome data must be reliable, accurate and timely due to their effects on the vehicular road user's life [4], [5].

ML and DL, as a subset of AI, can be utilized for developing effective models and provide a better and higher rate of prediction accuracy [6], [14]. Although ML approaches provide better performance than the traditional model, each of them has challenges and issues. The way to cover a single ML model's problem to be more beneficial and powerful is to combine different ML models, which is called EL [11]. Previous related works [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29] presented different ML, DL, EL and optimized approaches to enhance the performance of their model for traffic prediction on a road or in other networks or other domains. Most of them focus on improving accuracy. In this study, we try to take advantage of EL and focus on providing a model that improves the accuracy with an efficient prediction time and keeps the overall performance better and more stable. Moreover, the proposed model is designed explicitly for VANETs considering the effective features of both V2V and V2I datasets to maintain the quality of input data, which is challenging in intelligent vehicle area networks. The authors in [15] indicated that the existing traffic flow prediction in ITS has problems adapting to real-world applications. They worked on traffic data obtained from V2V with important features such as location, direction and speed. They applied three ML models including DT, SVM and RF. The obtained results were assessed based on classification metrics (i.e., accuracy, precision, recall and time) and showed a higher value of accuracy for the RF and consumed longer time than other models. The minimum time of prediction was assigned to SVM. The study needs to consider more metrics like the Receiver Operating Characteristic (ROC) curve and Area Under Curve (AUC) metrics to better evaluate the models, and none of these models performed well in all performance metrics (i.e., accuracy and time). In urban traffic prediction, Lee and Min [16] compared four different and popular ML models consisting of RF, Gradient Boosting Regression (GBR), K-Nearest Neighbor (KNN) and Multi-Layer Perceptron (MLP). The trajectory data with features including Vehicle ID, Vehicle position and lane, time and speed were considered. The evaluated results showed GBR and KNN performed best and worst, respectively, among the methods. Tong et al. [17] discussed traffic flow prediction in VANETs. They used an improved particle swarm optimization (PSO) algorithm to enhance support vector regression (SVR) parameters. The presented algorithm performed the best compared to DT and SVR with grid search optimization.

The hybrid LSTM-SAEs model was proposed in [18] for urban road traffic prediction on the Internet of Vehicles (IoV). The authors considered Long Short-term Memory (LSTM) as a developed structure of Recurrent Neural Network (RNN) and a strong model for prediction of time-series traffic data, which requires historical data. Moreover, they used the influential DL model stacked Auto Encoder (SAE) that can learn automatically from input data to have a pinpoint feature description. This research work took advantage of these dominant DL methods by merging them into one model called the EL model [19], [20]. Of note, considering the nonlinearity of traffic data, EL models have been famous these days. In other words, using different models together, we can build a stronger new model than each individual model and cover their flaws [21]. In addition, this EL model used feature engineering to improve accuracy. They considered big traffic data of 500,000 recorded samples from 51 road sections collected every 5 minutes. The approach achieves less prediction error than the base models, such as the autoregressive integrated moving average (ARIMA), Gated Recurrent Unit model (GRU), Deep Belief Network (DBN), LSTM and SAE.

Zheng et al. [22] presented a new EL model named EM for short-term traffic flow prediction. They combined three DL algorithms comprised LSTM, deep autoencoder (DAE) and Convolutional Neural Network (CNN), where CNN and

LSTM allow to consider both temporal-spatial traffic features. They employed two real-world traffic datasets to validate the model performance. Their approach includes hidden and softmax layers for final prediction. The output of each model was individually used as input for the EL hidden layer to ensure that each individual output came up with an equal quantity of features for the softmax layer before the final prediction. The obtained result shows a higher accuracy value compared with every single model besides the other two EL models named DA and CNN-LSTM (CLTFP). Moreover, they mentioned that their approach was robust in the case of high variance.

Stepanov et al. [23] emphasized the point that ML and DL models can optimize network traffic prediction. They collected cellular traffic data and applied three ML algorithms consisting of RF and bagging. They indicated the advantage of bagging that each tree can learn freely from realizing the results in another tree. In contrast, RF obtained results for each object based on the output from each tree. The evaluation of the results by RMSE, MAE and coefficient of determination (R2) shows that bagging performed well in all metrics. However, it consumed more learning time than the others. In the case of learning time, SVM is the best. This research work mentioned some interesting and helpful points related to the use of ML models for network traffic prediction, such as keeping the quality of feed data to ML by preprocessing, evaluating the importance of the features, and tuning the hyperparameters of ML models that can provide better prediction results.

A novel stacking ensemble learning approach aimed at mobile traffic prediction was presented in [24]. The proposed EL-MS model merges two ML algorithms named MLP as a base learner and the self-adaptive support vector regression model (SSVR) as a meta learner. The obtained result was evaluated by MSE to assess the difference between the actual and predicted values. The model shows stable and more accurate results than some of the other ML models (i.e., RNN, LSTM and CNN) and some ensemble models (i.e., DBN-SVM). Of note, MLP, as a simple DL model and well-known NN, can adapt to network properties and traffic patterns [24], [25], [26]. The EL model for VANETs is also a promising solution for designing efficient predictive models [27]. Designing an efficient and reliable prediction model is essential for network traffic management and optimization [28]. EL methods are applicable for minimizing bias, enhancing predictions and being robust to overfitting [27]. EL methods can achieve better results than standalone ML models in traffic prediction [29]. However, time consumption is one of their main problems [27].

In summary, although VANET services will result in safety and comfort for road users, inaccurate and false predictions, especially for safety applications, may affect the life of vehicular users. To our knowledge, designing an efficient intelligent model with the integration of AI and vehicular networks still needs much consideration. This motivated us to investigate highlighted points of related works from the input data to the model selection and effective evaluation of the model based on important factors for predictions. The existing research works indicate their approaches using a subset or combination of ML for traffic prediction on roads or for different types of networks. Some of them considered the quality of input data, and others just considered optimization on one ML algorithm. The lack of intelligent traffic prediction specifically for VANETs with both V2V and V2I data and covering the overall performance leads us to take advantage of EL and try to design an efficient network traffic prediction model with real data that can provide better overall performance and balance between accuracy and consumption time, which is highly important, especially for safety applications in VANETs.

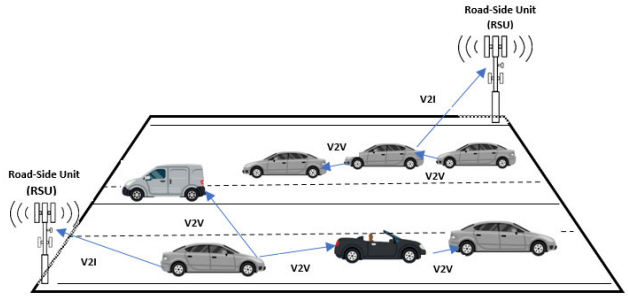

FIGURE 1. The basic architecture of VANET.

III. METHODOLOGY

In this section, we propose an optimized and efficient stacking ensemble learning model by taking advantage of EL models. This model is applied to VANETs to predict network traffic. Figure 1 shows that the basic architecture of VANETs is composed of V2V and V2I communication.

Vehicular networks are in the early stages of challenges relevant to the exploitation and adaptation of AI tools. ML is a subset of AI used for accurate analysis and prediction models [6]. However, each ML model suffers from its weaknesses and drawbacks. The way to cover a single ML model problem to be more beneficial and powerful is to combine different ML models, which is called EL [11]. Of note, considering the nonlinearity of traffic data, EL models have attracted considerable attention. The purpose of EL is to build a new strong model using several simple ML models together that can face the limitation of every single model in addition to taking advantage of their different views in solving the prediction task [21]. It also helps with reducing errors, achieving higher accuracy and robustness, fast computation and a better generalization ability than a single model in most problems [11], [12]. Figure 2 illustrates the workflow of our research work. We divided our model into three parts: data collection and preprocessing, model building and analysis of the result. This classification aims to differentiate the contribution of each part and then explains the procedure of the model in detail.

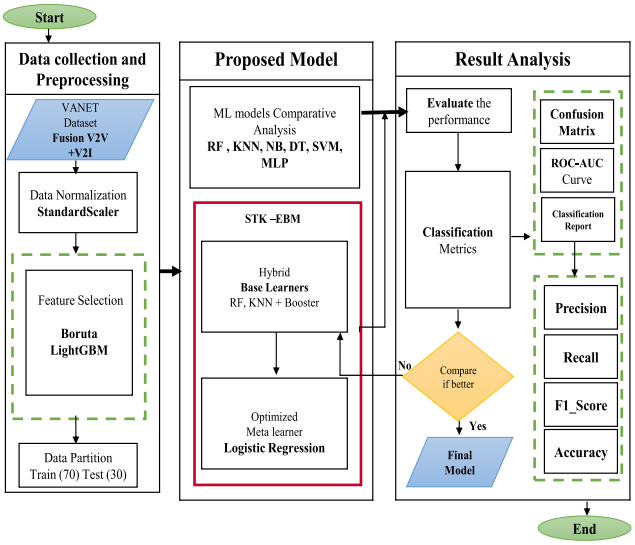

FIGURE 2. The workflow of this study.

A. DATA PREPROCCESSING

The benefits of data preprocessing and feature selection from input data before feeding into ML models are highlighted in several studies. Keeping the quality of the data with some preprocessing, such as cleansing the missing data, normalization and choosing the more relevant important features, can play a significant role in increasing the efficiency of the model [30], [31], [32]. Therefore, in the first part of our model, we considered normalization and feature selection methods to achieve a high-performance predictive model.

1) NORMALIZATION

In this phase, we first removed some redundant, missing, and meaningless values in the raw dataset. Then, we normalized our data using StandardScaler normalization [32], which scaled all feature values in the range of [0,1], and in this way, we performed simple preprocessing. The formula is defined as:

$$Z_{scaled} = \frac{(X - \mu)}{\sigma}, \tag{1}$$

where X = input variable, μ = Mean and σ = Standard Deviation.

After normalization, we selected the important features that will be discussed in Section II. Finally, we separated our dataset into training and testing sets. We assumed variable X with (i) the number of selected features as the input data. Accordingly, we labeled our target variable (y), which is the prediction of the traffic, in two classes of traffic (1) and no traffic (0).

2) FEATURE IMPORTANCE AND SELECTION

In this section, we present some research and several experimental analyses on various techniques that help us gain insight regarding the relevancy and importance of the features with the target variable. Each method gives a different perspective about how the variable can be useful depending on how the algorithms learn the target. Ching et al. [30] performed a comparative analysis of feature selection methods by considering different types of datasets. They focused on feature selection importance with the aim of data classification using machine learning algorithms. Inspired by this research work, we considered Boruta [30], which is an RF-based feature ranking technique. This algorithm determines the importance of variables with statistical judgment and detects all significant relevant features. Then, we used LightGBM as an ensemble feature selection method [31] to find optimal features in our dataset while considering V2V and V2I datasets separately and together. Finally, as one of the contributions of our research work, we come up with the best and most important, efficient and confirmed subset of the selective features by LightGBM and Boruta methods. Therefore, we can optimize network predictive model performance in our real dataset.

B. OVERVIEW OF THE PROPOSED STK-EBM MODEL ARCHITECTURE

In the first section, we collected and merged data from V2V and V2I in VANETs, and then we preprocessed the data. We considered the most informative attributes of the data using LightGBM and Boruta approaches. At the end of the first section, we divided our dataset into training and testing to feed to the ML models for prediction. In the following subsections, we describe the components of the presented model: stacking heterogeneous ensemble model structure, base learner element selection and meta learner.

Stacking ensemble learning, because of a heterogeneous integration strategy, has the ability to increase the generalization of the model. Strong model can be generated by combining several models, and the structure of stacked ensemble learning is composed of two layers [33], [34]: base learner and meta learner. The reason can be justifiable in the real world when an important decision needs to be made. Several experts in the related field have provided a consensus opinion and achieved one strong professional decision. We built our proposed model named the stacking optimized heterogeneous ensemble model for the network traffic prediction problem (STK– EBM). It is composed of two layers. The first layer is constructed from selective ML algorithms, which are called base learners. The combination of the base learners is based on a comparative performance analysis of the most popular ML models used in previous studies [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29] for traffic prediction. The second layer considers one algorithm called the meta learner and is responsible for the final prediction of the whole model. The proposed model focused on enhancing the overall performance in classification evaluations. In addition, there are some considerations and optimization in each layer that will be discussed in the following subsections. The global architecture of the proposed model is presented in Figure 3.

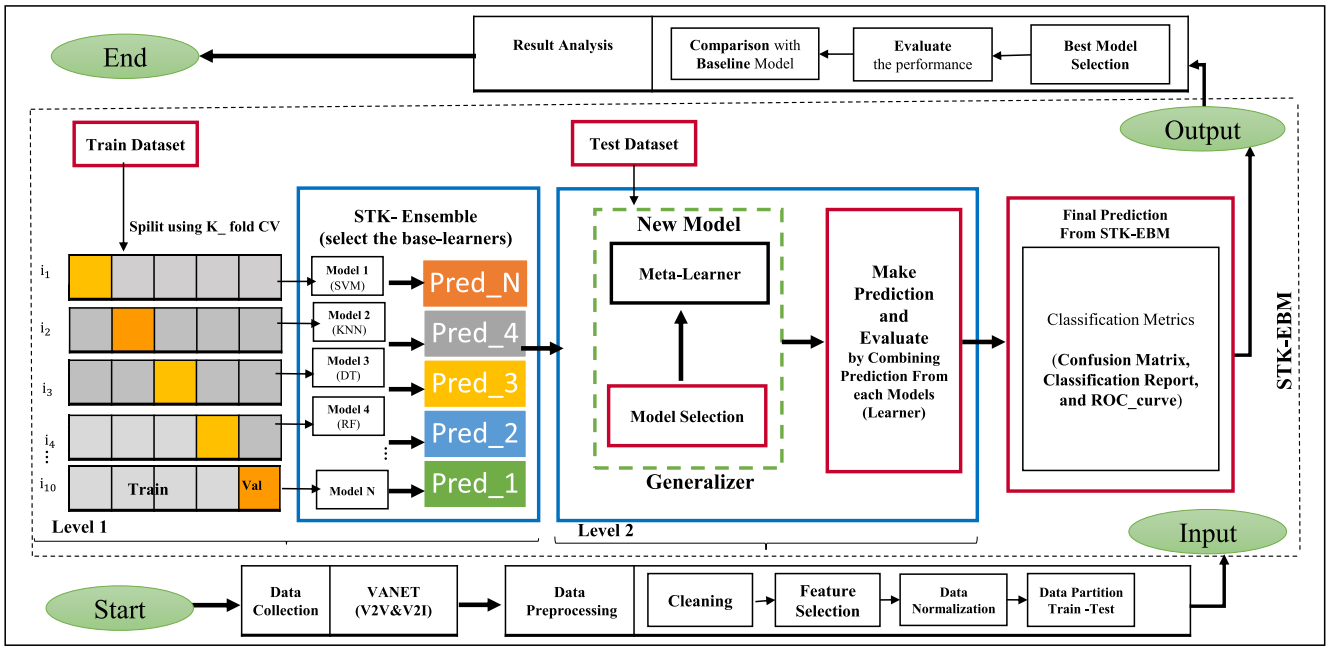

FIGURE 3. Architecture of the proposed stacking ensemble-based machine learning model for traffic prediction in VANETs.

1) STACKING HETEROGENEOUS ENSEMBLE MODEL STRUCTURE

"Stacked generalization is a generic term referring to any scheme for feeding information from one set of generalizers to another before forming the final guess" [34]. The deployment procedures for stacking an ensemble of models with the aim of network traffic forecasting are described as follows.

Step 1) The input variable is input into x to represent $x = x_1, x_2, \ldots, x_n$ that each x_i R^d (which is an attribute vector with d dimensions), and we put the target variable in (Y) that is labeled 0 for no traffic and 1 for the existence of traffic in the network. We split our dataset (Ds) into training and testing sets.

Step 2) The training set is divided into (k) equal-size subsets: $Ds = D_1, D_2, \ldots, D_K$. Therefore, the input of our model is the training set (train on k-1 one of these subsets), and our model is trained on the training set. The model evaluation is performed on the last subset as a validation set. Therefore, the validation is separated from the training set and is used to validate our model performance during training. In this way, we ensure that the same data point is not present in both testing and training. This helps to prevent the model from overfitting. The output is the final prediction from the meta learner of STK–EBM.

Step 3) In layer one, the base learners (b) learn from the training set, where $b = b_1, b_2, \ldots, b_m$ are the (m) base learners and form 1 to *k-1* fold and continue the learning process and the prediction results on the last fold (k). Then, all base learners predict in k repetition, where $p = p_1, p_2, \ldots, p_k$.

Step 4) The new dataset will be generated for the meta learner $D_s^{'}(x_i^{'}, y_i)$, where $x_i \in D_k$.

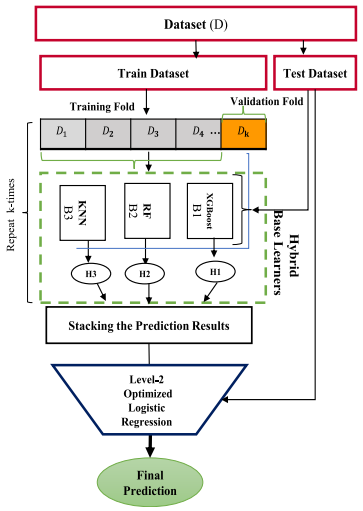

FIGURE 4. Framework of our proposed STK-EBM model.

It is designated by:

$$x_i^{'} = \left\{b_{k_1(x_i)}, b_{k_2(x_i)}, .., b_{k_m(x_i)}\right\}$$

Step 5) In the second layer of our model, the meta learner learns from a newly generated dataset $(p^{'})$

Step 6) The final output is the combination of base learner prediction in layer one by meta-learner as follows.

$$P_{(x)} = p^{'}_{(b_1(x), b_2(x), \ldots, b_T(x))}$$

In summary, we proposed a hybrid stacking ensemble model. The model is composed of two layers. We stacked a set of high-performance heterogeneous base learners in the first layer. These base learners were selected according to

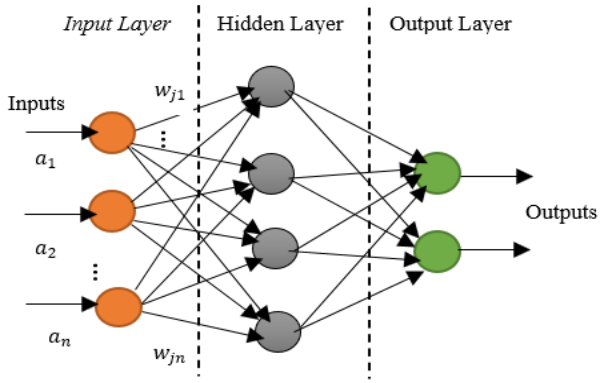

FIGURE 5. The general structure of a Multi-Layered Perceptron (MLP).

the performance analysis of the most popular ML prediction models [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29]. In this way, we reduced the complexity of stacking the useless and unsuitable models. The second layer is the meta learner, which combines the results of base learners and provides a final prediction. The framework of the stacking ensemble model with the steps mentioned above is depicted in Figure 4.

2) BASE LEARNER ELEMENT SELECTION

In this section, several popular ML algorithms used for traffic prediction are taken into account. The important issue is selecting the best combination of these algorithms when creating our model from scratch. This is another contribution of our research. In this context, we performed an experimental analysis of the most mentioned and popular ML models for traffic prediction, including RF [7], [15], [23], KNN [16], NB [15], DT [15], SVM [8], [15], [17], [23], [24] and MLP [10], [16], [24], [25], [26]. These models are individually trained, and the performance of each of them was evaluated. We find the best combination of these learners as the base learners. Base learners in our approach are diverse because each single ML model has a different view about solving the prediction task and brings its advantage and disadvantage for traffic prediction. Furthermore, we add a booster to the base learners to boost the prediction results. For this purpose, some ML algorithms, such as XGBoost [35], [36], and AdaBoost [37], can be employed. In these ways, we are able to bring more accuracy, adaptability and stability to the dynamic nature of traffic. Because our dataset is not sufficiently large to try on different DL models, we decided to perform MLP as a simple DL model, which is also common for the traffic prediction domain [10]. The MLP model, as the classical type of feed-forward neural network [16], [26], consists of three layers: the input layer, the output layer and the hidden layer, while the number of output nodes is based on the machine learning task. The classification task in our problem includes two output nodes. Regression tasks commonly consist of one node [26]. The general structure of MLP is depicted in Figure 5. The related formula is designated by [24], [25], and [33].

$$X = (\sum_{i=1}^{n} w_{ij} a_i) + b_j, \tag{2}$$

where a_i = input variable, n is the number of inputs, w_j = the connection weight, w_{ij} = input into the summing junction, b_j = the bias of neuron employed for summation.

$$F(X)_{=} u_j = F[(\sum_{i=1}^{n} w_{ij} a_i) + b_j], \tag{3}$$

where X generates the output through the transfer function F and u_j is the summing junction.

$$F(X) = \frac{1}{1 + e^{-x}}, \tag{4}$$

where the sigmoid activation function which is the connection weight from the i the input to the j th hidden neuron.

3) META LEARNER

In the final step of this section, we aimed to simplify the interpretation of the base learner prediction results using a simple meta learner. Therefore, we employ an improved Logistic Regression (LR) as the meta learner. It can find the optimal combination of the prediction results of all base learners. In other words, the meta learner was trained based on the prediction made by previous learners. This helps improve the predictive performance of network traffic. We used grid search cross-validation [38], [39] to improve the accuracy of LR. It can tune the hyperparameter and result using the best parameters of the algorithm [40]. Therefore, we can obtain more accurate results:

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k \tag{5}$$

$$f(z) = \frac{1}{1 + e^{-z}} = \frac{1}{1 + e^{-\left(\beta_0 + \sum_{\beta_i X_i}\right)}}, \tag{6}$$

where X = input variable, β = predicted weights and z = predicted output.

input variable (X) with 1 to k as the number of features are merged linearly with predicted weights (β) to predict an binary value as output (z). The weight indicates the variable impact on prediction. $\beta = \beta_1, \beta_2, \ldots, \beta_k$ in which β_1 *to* β_k used for assessing weight of input variable and β_0 for assigning the bias value. Moreover, The probability of existence traffic in the VANET represent by $f(z)$. It called transformation function with a range between one and zero ($0 \leq f(z) \leq 1$). This function transforms probabilities into a binary value. where, $z < 0.5$ *output* $\rightarrow 0$ *(no-traffic)*, else ($z \geq 0.5$) *output* $\rightarrow 1$ *(traffic)*.

In summary, we first perform an experimental analysis of the most popular ML models for traffic prediction, including RF, KNN, NB, DT, SVM and MLP. Second, when we build an EL model from scratch, the important issue is to select the best combination of ML algorithms. Therefore, in our EL approach, which is composed of base learners

and a meta learner, the base learners bring a different view about solving the prediction task. In this way, the model provides more accuracy, adaptability, and stability to the dynamic nature of traffic. The selected algorithms, including RF, KNN and XGBoost, can help us to address some issues. RF can solve the challenges related to scalability, and KNN showed better performance results than the other ML models based on our dataset. Furthermore, XGBoost was selected as a booster of the base learners' performance. It can also enhance the generalization ability, but its parallel learning ability with distributed computation will not impose additional time. Finally, we aimed to simplify the interpretation of the base learner prediction results using a simple meta learner. Accordingly, we employ an improved Logistic Regression (LR) to find the optimal combination of the prediction results of all base learners. Eventually, we take advantage of the most effective combination of ML models to provide a stable prediction model with better overall performance in terms of accuracy, prediction error and execution time.

C. EVALUATION AND ANALYSIS METRICS

We used the confusion matrix, classification report and CPU time as the most common classification evaluation metrics. Furthermore, we considered the Receiver Operating Characteristic (ROC) curve, which is a familiar tool to estimate the performance of binary classifiers and Area Under Curve (AUC) [48] to understand the stability of the model. Table 1 indicates the relationship between the actual and predicted classes [42]. Accuracy represents the ratio of TP and TN to the overall number of samples. Sensitivity (recall) shows the ability of the classifier to identify all positive samples in the actual class. Precision indicates the accuracy of positive prediction. The F1 score is affected by precision and recall, where the best score is 1.0. The related formula is designated by [42].

$$Accuracy = \frac{(TP + TN)}{TP + TN + FN + FP} \tag{7}$$

$$Sensitivity(Recall) = \frac{(TP)}{TP + FN} \tag{8}$$

$$Precision = \frac{(TP)}{TP + FP} \tag{9}$$

$$F1score = 2 * \frac{(Recall * Precision)}{(Recall + Precision)} \tag{10}$$

Area Under Curve(AUC), For a predictor f, an unbiased estimator of its AUC: tests whether positives are ranked higher than negatives

$$AUC(f) = \frac{\sum t_0 \ \in D^0 \sum t_1 \ \in D^1 \ 1\,[f\,(t_0) < f(t_1)]}{\left|D^0\right| . \left|D^1\right|}, \tag{11}$$

where $1[f(t_0) < f(t_1)]$ notes an indicator function which returns 1 if $f(t_0) < f(t_1)$ otherwise return 0, D^0 is the set of negative examples, and D^1 is the set of positive examples.

TABLE 1. Relationship between actual and predicted classes.

		Predicted Class	
		Positive Class	Negative Class
Actual Class	Positive Class	TP	FP
	Negative Class	FN	TN

where TP = True Positive, FP = False Positive[1]
FN = False Negative and TN = True Negative[2]

IV. EXPERIMENTAL RESULTS AND PERFORMANCE EVALUATION

A. DATASET

We used a real VANET dataset with DSRC-based communications between vehicles and between vehicles and roadside units in a realistic highway scenario [43].The experiments were performed in the northwest sector of Atlanta, GA along I-75 between Exit 250 and Exit 255. The selected area of the highway has five regular lanes and one High Occupancy Vehicle (HOV) lane that has been monitored during the day between 2 pm and 5 pm. This can be representative of most roads in the U.S. cities [44]. The data were acquired from GPS in 822.11 ad hoc networks. The GPS reported features such as location, longitude, latitude, speed and heading of the vehicles every two seconds. The accuracy of the location information recorded by interpolation was approximately five to seven meters. Moreover, IPerf was employed cooperatively with GPS reading network parameters. The V2V communication was measured based on the following vehicles, and both the sender and receiver were placed in vehicles that were moving in the same lane. The V2R communication was measured for moving vehicles, and the RSU station was the receiver, which was located on an elevated bridge with different heights. The sender was placed in the vehicle, and it broadcasted the packets while moving in the rightmost lane. The number of packets in V2R communication was 1470 bytes, which were broadcasted by the senders at an approximate rate of 150 packets/s [44]. All communication features, such as "log time", "location information of both sender and receiver", "velocity", "packet sent/received", and "signal quality", were associated and parsed together and were recorded. When the number of sending and receiving packets through VANET communications increases (i.e., many vehicular users on the road), traffic occurs in the network [6], [7]. We obtained 39,998 records of VANET communication data that combined V2V and V2I datasets. We solve our problem, which is intelligent network traffic prediction as a classification task. We take full advantage of all effective features of the real VANET dataset for traffic prediction using the LightGBM and Boruta methods. Therefore, the target is network traffic prediction, and we consider packet receiving as a network parameter to predict the network traffic. The target corresponds to the binary class (0: no traffic, 1: traffic).

B. EXPERIMENTAL DETAILS

For the implementation of the model proposed in this paper, the following modeling environments were used. Jupyter

Notebooks is an open-source and browser-based tool. It can work both locally and on the cloud [50]. Google Collaboratory is ''a product from Google Research'' [51], which is hosted on the Google Cloud Platform and is based on Jupyter Notebook. It is appropriate for ML and data analysis by providing fundamental AI libraries such as TensorFlow, Matplotlib, and Keres. It allows to write, execute and share python code via browser with others [52]. We used Google Colab to implement the model, and the programming language was Python version (3.7.13). Since the values of the dataset vary in unit and range, we normalized the data with the aim of bringing them into the same range for an accurate prediction model. Furthermore, for data visualization and analysis, we employ some well-known libraries, such as NumPy (fundamental computation), Pandas (data analysis), Scikit-learn [45], for scaling the features and data partitioning, and Matplotlib (visualization).

C. PERFORMANCE EVALUATION OF THE PROPOSED MODEL

There are three steps from the loaded dataset to model validation that will be described as follows.

1) In the first step, after importing the data into Google Colab [51] and reading the data, we preprocessed the data (i.e., cleansing redundant data and removing space) and checked for missing values in variables. Then, we put the feature variables to X and the target variable to y. Next, we scale the features using Scikit-Learn libraries. At the end of this step, the data were split using Scikit-Learn [45], in which 70% of the data were considered for training, while the remaining 30% were used as test data. The 10-fold cross-validation method was used in our model for parameter optimization by tuning the hyperparameters and configuration of the model, which eventually led to a boost in the performance of the model [23], [41].
2) In the second step, considering most related features in our dataset, several popular ML algorithms, which have been used for traffic prediction, were taken into account including RF, KNN, NB, DT, SVM and MLP.
3) In the last step, we trained all abovementioned models, and then we evaluated and analyzed the performance of each prediction model from the literature and our proposed model. We compared them in terms of classification metrics, which will be analyzed in the following section. Furthermore, we highlighted the comparative analysis results of popular ML models and our proposed model.

1) FEATURE SELECTION RESULTS

In this section, we extracted the importance of features, their relevancy and how they affected the prediction of the target (i.e., network traffic prediction). We considered the V2V and V2I datasets separately and together to determine which variable needs to be kept in our dataset. First, it is worth mentioning that in all individual and merged datasets, time remained a highly important feature. Second, an interesting correlation was found in V2V data, where sender speed and receiver speed were placed in the almost same degree of importance variable. The significance of sender speed was much higher in the V2I dataset and ranked third after time and sender location. Turning now to the V2V and V2I as combined data, the highest value of importance belongs to the sender and receiver location and then the sender and receiver speed.

We plotted the feature importance for the V2V and V2I datasets separately and together using lightGBM, as shown in Figure 6. The abovementioned analysis, the provided plots, and eventually the features confirmed by Boruta (i.e., relevant features with a ranking of one) were taken into account. In conclusion, we selected as many variables as possible that are important, efficient and confirmed by these methods including time, sender location, sender and receiver speed, and receiver location.

2) CLASSIFICATION RESULTS

Based on the literature review, some ML models may be a good fit for a particular problem. However, each model can fail in different ways. Therefore, we find the best model that fits our dataset to solve the problem of network traffic prediction. We used the confusion matrix, classification report, ROC curve and CPU time as the classification metrics [42] to evaluate the performance of the most commonly used models for traffic prediction. We first applied a ''dummy'' classifier, which is a simple ML model that randomly makes predictions by considering the class distribution of the training set [46]. We obtained 0.668 accuracy, which confirms the importance of considering the feature variable as our input. In addition, it improves our prediction results. In the following paragraphs, we compared well-known popular ML models with our proposed models using the abovementioned metrics. Regarding Table 1, the relationship between the actual and predicted classes is presented [47]. There are four states in the confusion matrix. Therefore, the confusion matrix can give us insight into the probability that a model is confused when it produces the prediction results. The classification reports consist of precision, recall F1 score and accuracy that complement each other to evaluate the classification models. Precision is related to True Positive (TP) and False Positive (FP) states, which measure only positive prediction, and in our case, they are related to correctly predicted existence of traffic or incorrectly predicted traffic situations.

Accordingly, this metric ignores the negative states and must be coupled with recall, which considers True Positive (TP) and False Negative (FN) in which we receive the wrong alarm about the existence of traffic in the network and bring us unexpected decision results. In this paper, FN means no traffic as a prediction result while the network is in a traffic situation, and FP gives us traffic as a prediction result while the network is in a no-traffic situation. Both FN and

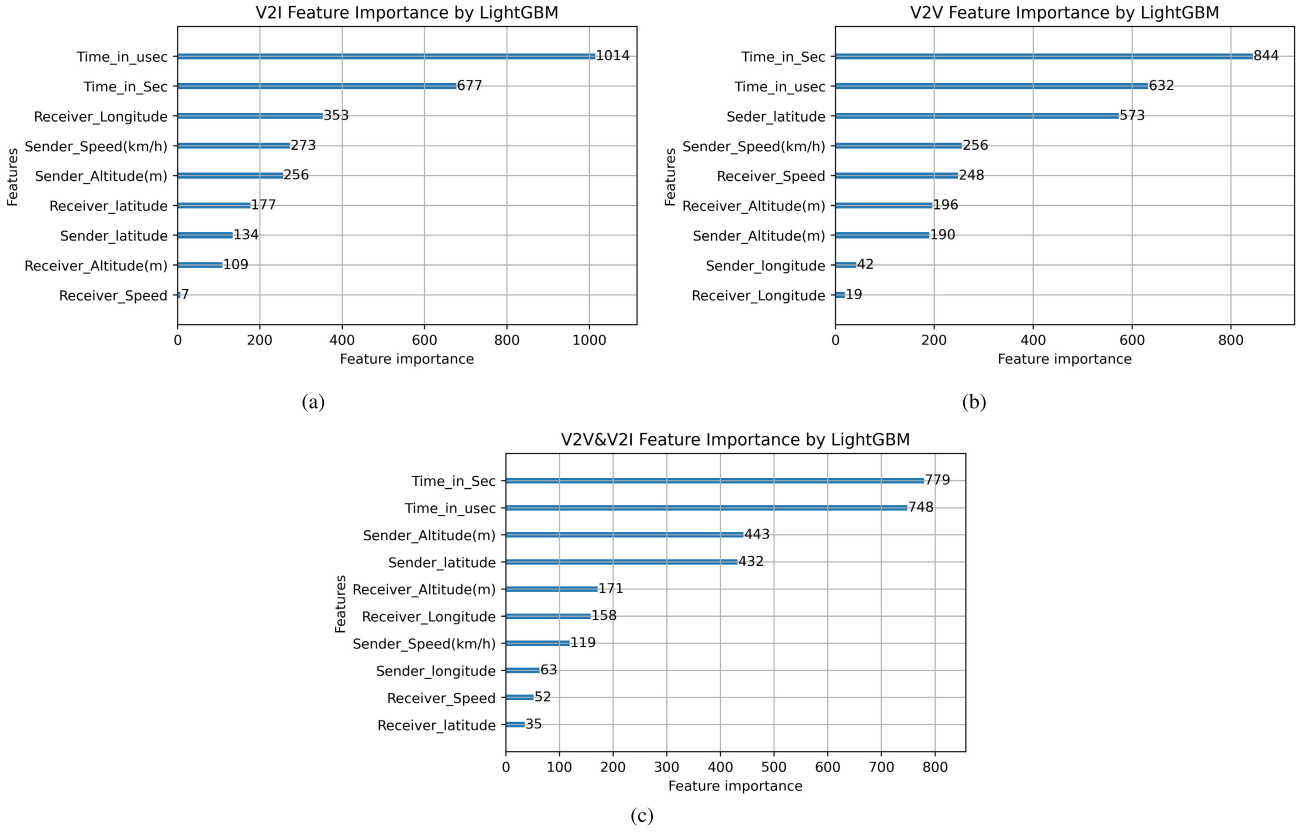

FIGURE 6. Feature importance result. (a) V2I dataset using LightGBM. (b) V2V datasets using LightGBM. (C) V2I and V2V datasets using LightGBM.

FP cause error and incorrect traffic prediction, and the F1 score is a weighted average score of recall and precision. In our analysis result, the predictive model with a high recall value means that the majority of the existence of traffic in the network is predicted correctly, and there is a slight probability of incorrect prediction. A model that can give us good accuracy while consuming more time for prediction results cannot be applicable in the case of time-sensitive tasks such as traffic prediction. Therefore, we tried to maintain a balance between accuracy and time consumption in our model.

Furthermore, the ROC for estimating the performance of binary classifiers and Area Under Curve (AUC) [47] for evaluating the stability of the model can be considered. When the ROC curve is distant from the middle-dotted line and converges to the upper left side and the AUC value is close to one, this implies an ideal model. Ultimately, all discussed issues of the results are provided in Figures 6-8, and Table 3 will be analyzed.

Figure 7 provides the results obtained from the confusion matrix performance of NB, RF, DT, KNN, SVM and MLP. Considering the abovementioned points, an unpleasant outcome that came from the incorrect prediction of traffic with both FN and FP resulted in a negative impact on vehicular network users on the road and a decline in the quality of service.

Regarding measuring the error rate from the confusion matrix, the best FN belonging to RF was 0.02%. However, the worst value of FP is also associated with this model (9.03%). On the other hand, KNN is best in FP, which was 4.15%, but it was not sufficiently good for TP. Comparing the different model results clearly shows that the proposed model maintained a good balance between these factors, which are 4.88% for FP and 1.00% assigned to FN, as well as a better rate of correct prediction (TP) than KNN.

Based on previous studies, SVM and MLP are mostly considered for traffic prediction. SVM can adapt to the dynamic characteristics of traffic. Moreover, the important drawback of this model is related to selecting an adequate kernel and parameter. Thus, in Table 2, we made a comparison of the SVM performance with different types of kernels, and the obtained results confirmed that SVM with the polynomial kernel shows more accuracy than the other types with slightly lower CPU time. Therefore, we considered the best kernel and optimized SVM. Finally, we compared the accuracy of this optimized SVM model with our proposed model.

Additionally, the MLP model is the classical type of feed-forward neural network [16], [24]. It consists of three types of layers: the input layer, the output layer and the hidden layer. Since our dataset is not sufficiently large to try on the different DL models, we decided to perform the simple neural network model, which is popular for the traffic prediction

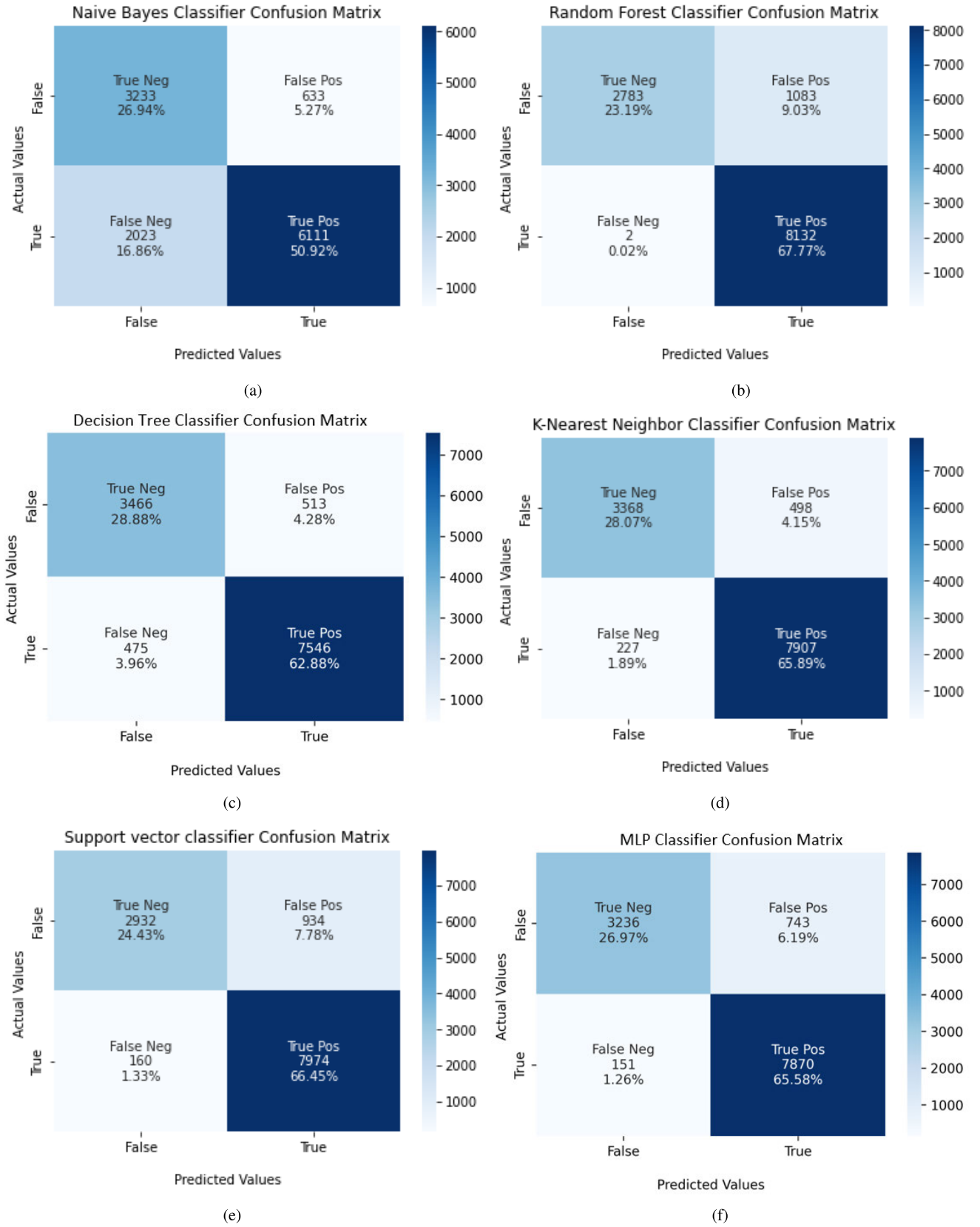

FIGURE 7. Confusion matrix of classification performance by (a) Naive Bayes Classifier, (b) Random Forest Classifier, (c) Decision Tree Classifier, (d) K-Nearest Neighbor Classifier, (e) Support Vector Classifier, and (f) MLP Classifier.

TABLE 2. Comparison of the performance of SVM with different types of kernels.

Support vector machine classifier	Accuracy	Time(ms)
SVM(Linear Kernel)	0.907	8.11
SVM(Sigmoid Kernel)	0.851	7.63
SVM(RBF Kernel)	0.582	5.72
SVM(Polynomial Kernel)	0.910	6.68

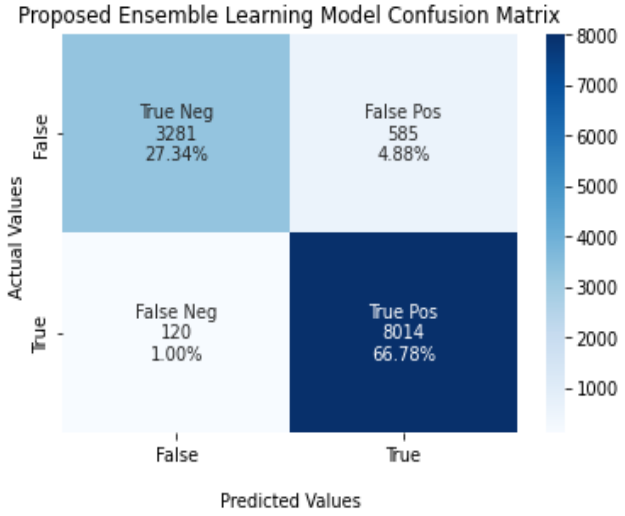

FIGURE 8. Confusion matrix of classification performance by the proposed ensemble learning model.

domain [10]. Figure 7 shows that the percentages of FN and FP for the MLP classifier were not noticeable among the other models and our proposed model.

As we explained in the methodology section, our model consists of two layers: base learners and meta-learners. In the first layer of the stacking ensemble of models, we selected RF, KNN and XGBoost. Each of them can help us to cover an issue. For example, RF can solve the challenges related to scalability since it has the ability to train models in parallel [49]. KNN was selected to obtain better performance than the other ML models. Finally, XGBoost, as an efficient and scalable implementation of the Gradient Boosting Machine (GBM) was selected to act as a booster to our base learner results. Considering the distributed computing and parallel learning ability of this model, XGBoost will not impose extra time for the prediction result. However, it enables higher prediction accuracy and can make our model more precise. Moreover, this model can handle the challenge in DT, which is related to easy overfitting. Eventually, XGBoost enhances the generalization ability [35]. Therefore, we considered a model that can take advantage of all these points and use the best effective combination of these models. In the second layer, which is the meta learner, we used logistic regression. In addition, we made some improvements at this layer, such as hyperparameter tuning, and we built our meta learner using the grid search algorithm. It helps us to select the best configuration of model parameters, which leads to maximizing the model performance.

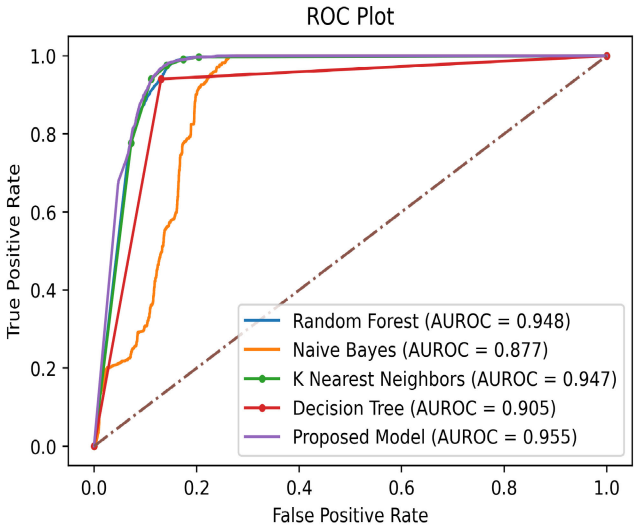

FIGURE 9. Comparison the ROC Curve with different ML models and the baseline models and our proposed model.

In this section, the confusion matrix of our proposed model is presented in Figure 8. In Figure 9, the results obtained by the ROC curve and AUC values were compared, which can help us to understand the stability of a classification model. When the ROC curve is distant from the middle-dotted line and converges to the upper left side and the AUC value is close to one, this implies an ideal model [47]. Considering this analysis, the proposed model showed better performance and stability than the other ML models in distinguishing between positive and negative classes. The AUROC value for our proposed model was 0.955, which is the highest among the other models.

Furthermore, we added a booster in the first layer of our stacking ensemble model. The comparison results of the AUROC value and curve with and without the booster are presented in Figure 10. The computed AUC value is 0.948 without using XGBoost as a booster and 0.955 with considering a booster. It confirmed obtaining better results with the booster.

XGBoost is known as a high-power predictive model for increasing the efficiency of the model. In Figure 11, the training and testing accuracies of XGBoost as our booster algorithm are shown. The training accuracy is 0.9426, and the test accuracy is 0.9405, which are close.

In the second layer, logistic regression is employed as our meta learner. We performed parameter optimization using a grid search algorithm combined with cross-validation (CV) called Grid Search CV [23], [41], in which grid search was used for parameter tuning. It considers each fusion of algorithm parameters specified in a grid. It can help to boost the performance of the model. Additionally, because CV performs oversampling, there is a special algorithm known as group k-fold cross-validation. It ensures that the same data point is not present in both testing and training. This helps us avoid overfitting.

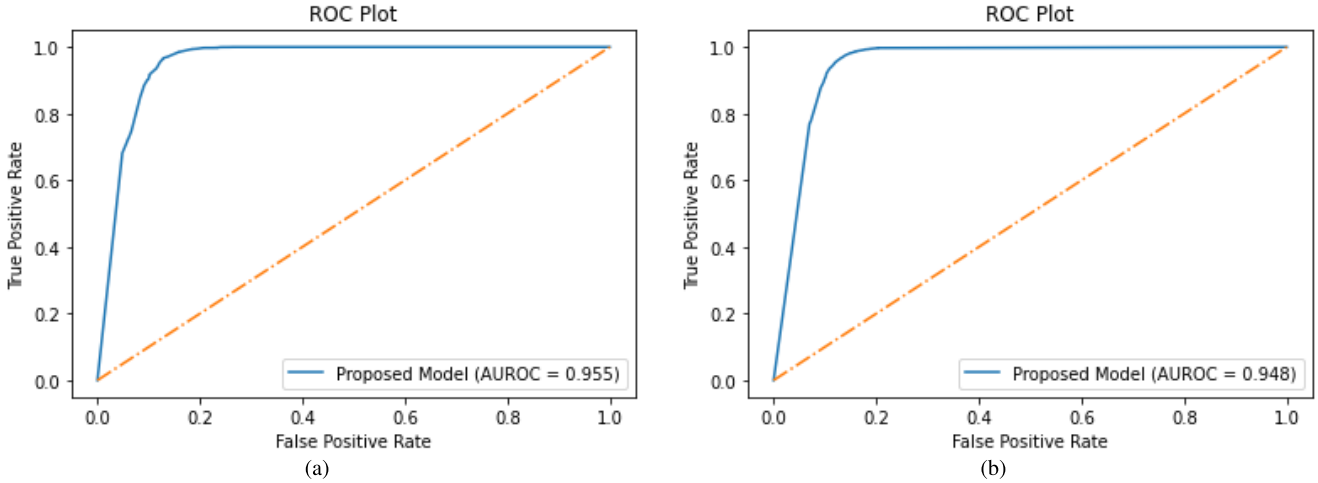

FIGURE 10. ROC curve of the proposed STK–HEM Model (a) Using a booster. (b) Without using a booster.

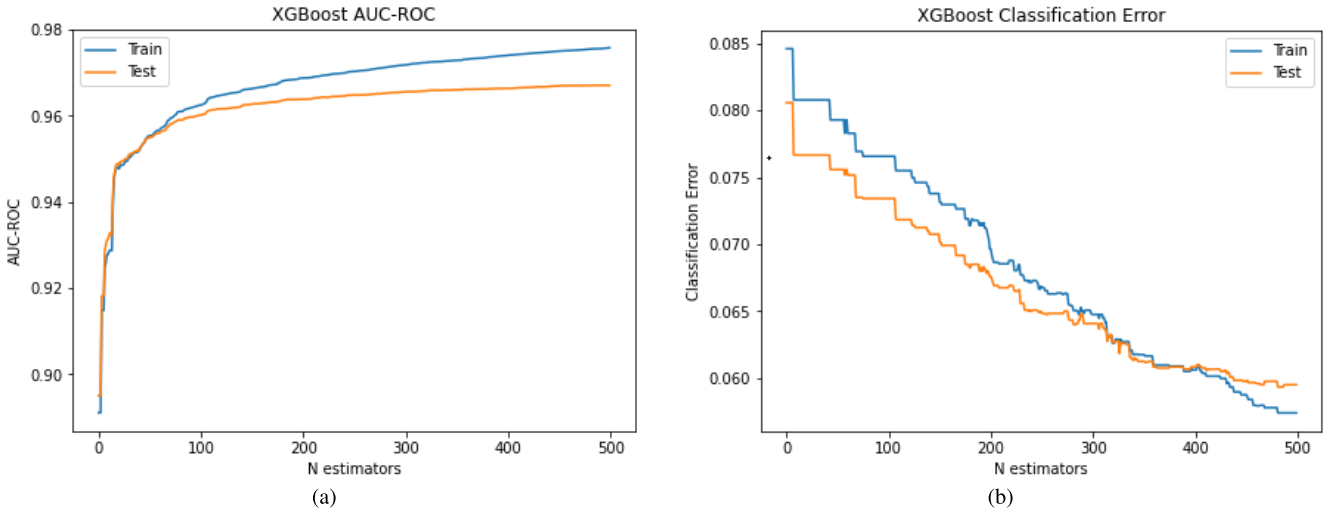

FIGURE 11. XGBoost as a booster in the first level of the proposed model. (a) AUC-ROC curve. (b) classification error.

In addition, we plotted the learning curve of the meta learner, as shown in Figure 12. The learner has mastered the learning task with a high validation loss at the beginning. However, after 10,000 training samples were trained and validated, the data converged together and stayed close to each other with a minimal gap until the end. This means that our meta-model was well-fitted. Furthermore, the learning curve can be applied as a mechanism to diagnose the machine learning model bias-variance problem, and it is possible to see the trade-off between bias-variance with our chosen super estimator model.

In summary, we built a stacking ensemble model with a booster for network traffic prediction problems. We focus on increasing overall efficiency. Moreover, the main part of our research work was selecting the best combination of the algorithm and model when constructing our model from scratch. We made many considerations to ensure that our model works well for each dataset (V2V)(V2I) and for the combination of them. Of note, because the time may vary by each execution, we considered an average value of 10 runs of the model. Moreover, the model gave us even better results for the vehicle-to-vehicle dataset, which means that when the sender and receiver were both on the move, the predictive model was stable and performed well. Finally, the proposed experimental analysis indicated that our proposed model was the winner considering every aspect. Additionally, the CPU time for training the model was similar to NB as the faster learner classifier in the baseline model. The performance analysis of the different prediction models and our proposed model with classification metrics for V2V and V2I are represented in Table 3 and Figure 13.

In summary, the efficiency of ML models relies on the size of datasets, the selected features and the type of problems. Subsequently, we identified the best-fitted ensemble model for network traffic prediction in VANETs. Our proposed model not only obtained a balance considering several aspects (including accuracy, error and time) but also showed improvement and better results.

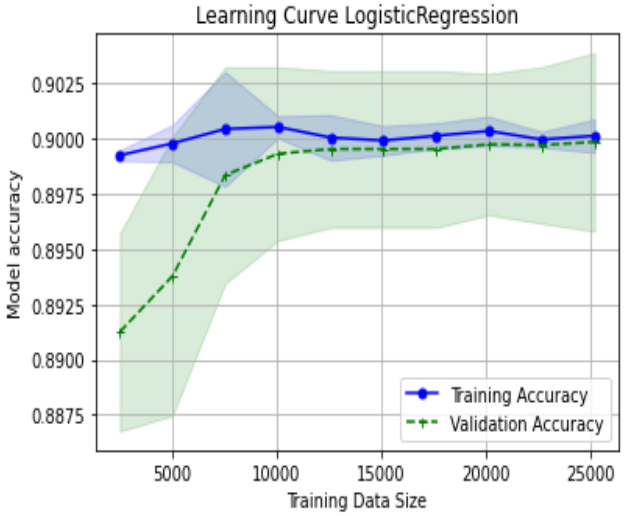

FIGURE 12. Learning curve super learner.

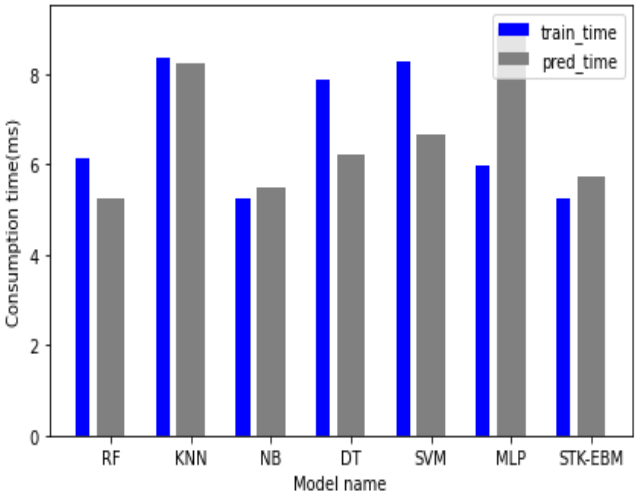

FIGURE 13. Comparison of the time consumption of different ML models and our proposed model.

TABLE 3. Performance analysis of the baseline prediction models and our proposed model with classification metrics for V2V and V2I.

Prediction Models	Precision	Recall	F1_Score	Accuracy
Random Forest	0.94	0.86	0.89	0.909
K-Nearest Neighbor	0.94	0.92	0.93	0.936
Naive Bayes	0.76	0.79	0.77	0.777
Decision Tree	0.91	0.91	0.91	0.917
Support Vector Machines	0.92	0.87	0.89	0.908
Multilayer Perceptron (MLP)	0.94	0.90	0.92	0.925
Proposed Model	0.94	0.98	0.95	0.941

3) COMPARATIVE ANALYSIS OF POPULAR ML MODELS FOR TRAFFIC PREDICTION WITH THE PROPOSED MODEL

In the experimental results, most of the common machine learning models are tested and compared, which can fully demonstrate the advantages of the proposed model. In this section, we highlight the results of the comparative analysis by considering all evaluated metrics. Regarding the confusion matrix, the proposed model maintained a good balance between both FN and FP that caused the error and incorrect traffic prediction. The other ML models showed best, worst, or not sufficiently good results in one of these factors. The ROC curve and AUC values can be considered to understand the stability of a classification model. Our model showed better performance and stability than the other ML models in distinguishing between positive and negative classes. Moreover, the AUROC value was the highest among the other models, which indicates that our model was an ideal model and winner in this metric. The accuracy of the proposed model was higher than that of the different tested ML models, which is highly important in our problem. The incorrect prediction of traffic in a network causes an error and undesirable decision in a real application. Our model also provided the highest recall value among other models, which means that the majority of the existence of traffic in the network was predicted correctly, and there is a slight probability of incorrect prediction. Furthermore, the F1 score as a weighted average score of the recall and precision also in our model was higher than that in the best standalone ML models (e.g., KNN, MLP). However, a model that can give us good accuracy while consuming more time for prediction results cannot be applicable to time-sensitive tasks such as traffic prediction. Therefore, we also consider the consumption time for prediction, and the result of the proposed model was almost near the best one, which is NB. Finally, the proposed model considers a trade-off between all popular ML models. It enhances the overall performance by increasing accuracy with minimum time and providing stability in the results.

Ultimately, a more accurate and stable prediction of traffic in the network can help to identify the failure of the network and its dependent services. It can effectively help us to predict traffic in the network and mitigate it before declining the quality of services for the users. Specifically, vehicular networks that are related to important services such as preventing congestion and accidents for road users will be more essential. Ensemble learning techniques have been investigated for decades but still attract all domain researchers with valuable advantages in different learning tasks. Of note, some challenges must be taken into consideration. For instance, identifying the best set of base learners to integrate for a given problem without much trial and error is time-consuming. However, EL provides an efficient tool to extract highly accurate and robust models, especially from dynamic, noisy, and heterogeneous data, bringing notable benefits to real-world applications. For example, traffic prediction.

V. CONCLUSION

In this paper, we proposed an ensemble-based ML model for forecasting network traffic in VANETs. We compared various most commonly used ML models that were suitable for forecasting traffic and used them in our proposed model. Moreover, to effectively evaluate the performance, we considered different classification metrics including confusion matrix, ROC-AUC curve, accuracy, precision, recall, F1 score, and time. Then, we discussed how we used the stacking ensemble strategy for building a best-fitted model for designing an

efficient network prediction model. The proposed stacking ensemble boosted model (STK–EBM), enhances the overall efficiency in all metrics and obtains stable prediction using the integration of RF, KNN, XGBoost, and LR.

The limitation of the presented model is related to DSRC access technology, which is just for short-range coverage. Additionally, there is just a basic type of communication between vehicles and roadside units. However, in a practical application, we will have various types of communication, such as vehicle to pedestrian and vehicle to a cellular network that is called vehicle to everything (V2X). Therefore, we need to provide more communication types with different access technologies, such as LTE/5G, that provide large-range coverage. In this way, we can obtain a better perception of the dynamic nature of traffic for such a prediction model in practical applications. In future work, we plan to work on different technologies, such as vehicular communication in cellular networks and V2X communications in VANETs, to design an efficient prediction model.

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] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A comprehensive survey on vehicular ad hoc network," *J. Netw. Comput. Appl.*, vol. 37, pp. 380–392, Jan. 2014.

[3] X. Huang, D. Zhao, and H. Peng, "Empirical study of DSRC performance based on safety pilot model deployment data," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 10, pp. 2619–2628, Oct. 2017.

[4] J. E. Siegel, D. C. Erb, and S. E. Sarma, "A survey of the connected vehicle landscape—Architectures, enabling technologies, applications, and development areas," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 8, pp. 2391–2406, Aug. 2018.

[5] H. Khelifi, S. Luo, B. Nour, H. Moungla, Y. Faheem, R. Hussain, and A. Ksentini, "Named data networking in vehicular ad hoc networks: State-of-the-art and challenges," *IEEE Commun. Surveys Tuts*, vol. 22, no. 1, pp. 320–351, 1st Quart., 2020.

[6] 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.

[7] D. Alekseeva, N. Stepanov, A. Veprev, A. Sharapova, E. S. Lohan, and A. Ometov, "Comparison of machine learning techniques applied to traffic prediction of real wireless network," *IEEE Access*, vol. 9, pp. 159495–159514, 2021.

[8] X. Feng, X. Ling, H. Zheng, Z. Chen, and Y. Xu, "Adaptive multi-kernel SVM with spatial–temporal correlation for short-term traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 2001–2013, Jun. 2019.

[9] F. Tang, B. Mao, N. Kato, and G. Gui, "Comprehensive survey on machine learning in vehicular network: Technology, applications and challenges," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 2027–2057, Jun. 2021.

[10] 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.

[11] I. K. Nti, A. F. Adekoya, and B. A. Weyori, "A comprehensive evaluation of ensemble learning for stock-market prediction," *J. Big Data*, vol. 7, no. 1, pp. 1–40, Dec. 2020.

[12] Y. Li and Z. Yang, "Application of EOS-ELM with binary Jaya-based feature selection to real-time transient stability assessment using PMU data," *IEEE Access*, vol. 5, pp. 23092–23101, 2017.

[13] M. Faezipour, M. Nourani, A. Saeed, and S. Addepalli, "Progress and challenges in intelligent vehicle area networks," *Commun. ACM*, vol. 55, no. 2, pp. 90–100, Feb. 2012.

[14] F. Falahatraftar, 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] G. Meena, D. Sharma, and M. Mahrishi, "Traffic prediction for intelligent transportation system using machine learning," in *Proc. 3rd Int. Conf. Emerg. Technol. Comput. Eng., Mach. Learn. Internet Things (ICETCE)*, Jaipur, India, Feb. 2020, pp. 145–148.

[16] Y.-J. Lee and O. Min, "Comparative analysis of machine learning algorithms to urban traffic prediction," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Jeju, Korea (South), Oct. 2017, pp. 1034–1036.

[17] J. Tong, X. Gu, M. Zhang, J. Wan, and J. Wang, "Traffic flow prediction based on improved SVR for VANET," in *Proc. 4th Int. Conf. Adv. Electron. Mater., Comput. Softw. Eng. (AEMCSE)*, Changsha, China, Mar. 2021, pp. 402–405.

[18] C. Chen, Z. Liu, S. Wan, J. Luan, and Q. Pei, "Traffic flow prediction based on deep learning in Internet of Vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 6, pp. 3776–3789, Jun. 2021.

[19] Y. Wu and H. Tan, "Short-term traffic flow forecasting with spatialtemporal correlation in a hybrid deep learning framework," *Comput. Vis. Pattern Recognit.*, vol. 23, no. 1, pp. 1–14, 2016.

[20] Z. Zheng, Y. Yang, J. Liu, H.-N. Dai, and Y. Zhang, "Deep and embedded learning approach for traffic flow prediction in urban informatics," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3927–3939, Oct. 2019.

[21] F. Zhao, G.-Q. Zeng, and K.-D. Lu, "EnLSTM-WPEO: Short-term traffic flow prediction by ensemble LSTM, NNCT weight integration, and population extremal optimization," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 101–113, Jan. 2020.

[22] G. Zheng, W. K. Chai, and V. Katos, "An ensemble model for short-term traffic prediction in smart city transportation system," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Waikoloa, HI, USA, Dec. 2019, pp. 1–6.

[23] 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.

[24] Z. Li, D. Cai, J. Wang, J. Fu, L. Qin, and D. Fu, "A stacking ensemble learning model for mobile traffic prediction," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Chongqing, China, Aug. 2020, pp. 542–547.

[25] F. K. Oduro-Gyimah, K. O. Boateng, P. B. Adu, and K. Quist-Aphetsi, "Prediction of telecommunication network outage time using multilayer perceptron modelling approach," in *Proc. Int. Conf. Comput., Comput. Model. Appl. (ICCMA)*, Brest, France, Jul. 2021, pp. 104–108.

[26] A. Dimara, D. Triantafyllidis, S. Krinidis, K. Kitsikoudis, D. Ioannidis, E. Valkouma, S. Skarvelakis, S. Antipas, and D. Tzovaras, "MLP for spatio-temporal traffic volume forecasting," in *Proc. IEEE Int. IoT, Electron. Mechatronics Conf. (IEMTRONICS)*, Toronto, ON, Canada, Apr. 2021, pp. 1–7.

[27] A. Mchergui, T. Moulahi, and S. Zeadally, "Survey on artificial intelligence (AI) techniques for vehicular ad-hoc networks (VANETs)," *Veh. Commun.*, vol. 34, Apr. 2022, Art. no. 100403.

[28] R. Goscien and A. Knapinska, "Efficient network traffic prediction after a node failure," in *Proc. Int. Conf. Opt. Netw. Design Model. (ONDM)*, Warsaw, Poland, May 2022, pp. 1–6.

[29] J. Jenifer and R. J. Priyadarsini, "An ensemble based machine learning approach for traffic prediction in smart city," in *Proc. Int. Conf. Advancements Electr., Electron., Commun., Comput. Autom. (ICAECA)*, Coimbatore, India, Oct. 2021, pp. 1–6.

[30] R.-C. Chen, C. Dewi, S.-W. Huang, and R. E. Caraka, "Selecting critical features for data classification based on machine learning methods," *J. Big Data*, vol. 7, no. 1, pp. 1–26, Dec. 2020.

[31] O. Aouedi, K. Piamrat, and B. Parrein, "Performance evaluation of feature selection and tree-based algorithms for traffic classification," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Montreal, QC, Canada, Jun. 2021, pp. 1–6.

[32] V. N. G. Raju, K. P. Lakshmi, V. M. Jain, A. Kalidindi, and V. Padma, "Study the influence of normalization/transformation process on the accuracy of supervised classification," in *Proc.3rd Int. Conf. Smart Syst. Innov. Technol. (ICSSIT),* Tirunelveli, India, Aug. 2020, pp. 729–735.

[33] Z. Liao, M. Su, G. Ning, Y. Liu, T. Wang, and J. Zhou, "A novel stacked generalization ensemble-based hybrid PSVM-PMLP-MLR model for energy consumption prediction of copper foil electrolytic preparation," *IEEE Access*, vol. 9, pp. 5821–5831, 2021.

[34] D. H. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, no. 2, pp. 241–259, 1992.

[35] T. Chen and C. Guestrin, "XGboost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, San Francisco, CA, USA, Jun. 2016, pp. 785–794.
[36] D. Zhang, L. Qian, B. Mao, C. Huang, B. Huang, and Y. Si, "A data-driven design for fault detection of wind turbines using random forests and XGboost," *IEEE Access*, vol. 6, pp. 21020–21031, 2018.
[37] X. Zhang and F. Ren, "Improving SVM learning accuracy with AdaBoost," in *Proc. 4th Int. Conf. Natural Comput.*, Jinan, China, 2008, pp. 221–225.
[38] Z. Jianjun, X. Yuanbiao, and F. Renhai, "Network traffic forecasting based on logistic iterative regression model," in *Proc. IEEE 3rd Int. Conf. Inf. Commun. Signal Process. (ICICSP)*, Shanghai, China, Sep. 2020, pp. 424–429.
[39] S. Xin, X. Yuanbiao, Z. Qijia, L. Zhimao, and F. Renhai, "Traffic forecasting of core network based on improved logistic regression," in *Proc. IEEE 9th Int. Conf. Inf., Commun. Netw. (ICICN)*, Xi'an, China, Nov. 2021, pp. 102–106.
[40] D. Kleinbaum, K. Dietz, M. Gail, and M. Klein, *Logistic Regression*. Berlin, Germany: Springer, 2002, p. 536.
[41] G. S. K. Ranjan, A. K. Verma, and S. Radhika, "K-nearest neighbors and grid search CV based real time fault monitoring system for industries," in *Proc. IEEE 5th Int. Conf. Converg. Technol. (ICT)*, Bombay, India, Mar. 2019, pp. 1–5.
[42] M. Awad and R. Khanna, *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*. Berlin, Germany: Springer, 2015, pp. 1–17.
[43] R. M. Fujimoto, R. Guensler, M. P. Hunter, H. Wu, M. Palekar, J. Lee, and J. Ko. (Mar. 2006). *CRAWDAD Dataset Gatech/Vehicular*. Accessed: Mar. 15, 2006. [Online]. Available: https://crawdad.org/gatech/vehicular/20060315
[44] 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.*, Sep. 2005, pp. 83–84.
[45] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, and J. Vanderplas, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, no. 10, pp. 2825–2830, Jul. 2017.
[46] G. Figueroa, Y.-S. Chen, N. Avila, and C.-C. Chu, "Improved practices in machine learning algorithms for NTL detection with imbalanced data," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Chicago, IL, USA, Jul. 2018, pp. 1–5.
[47] J. Davis and M. Goadrich, "The relationship between precision-recall and ROC curves," in *Proc. 23rd Int. Conf. Mach. Learn.*, Pittsburgh, PA, USA, 2006, pp. 233–240.
[48] A. Géron, "*Hands-On Machine Learning With Scikit-Learn, Keras, and TensorFlow*, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2019, pp. 88–100.
[49] J. Riihijarvi and P. Mahonen, "Machine learning for performance prediction in mobile cellular networks," *IEEE Comput. Intell. Mag.*, vol. 13, no. 1, pp. 51–60, Feb. 2018.
[50] B. M. Randles, I. V. Pasquetto, M. S. Golshan, and C. L. Borgman, "Using the Jupyter notebook as a tool for open science: An empirical study," in *Proc. ACM/IEEE Joint Conf. Digit. Libraries (JCDL)*, Jun. 2017, pp. 1–2.
[51] Google. (2022). *Colaboratory: Frequently Asked Questions*. Accessed: Jul. 19, 2022. [Online]. Available: https://research.google.com/colaboratory/faq.html
[52] T. Carneiro, R. V. M. Da Nobrega, T. Nepomuceno, G.-B. Bian, V. H. C. De Albuquerque, and P. P. R. Filho, "Performance analysis of Google colaboratory as a tool for accelerating deep learning applications," *IEEE Access*, vol. 6, pp. 61677–61685, 2018.

PARVIN AHMADI DOVAL AMIRI received the B.Sc. degree in computer software engineering from Islamic Azad University, Kashan Branch, Kashan, Iran, in 2009, and the M.Sc. degree in computer software engineering from Islamic Azad University, Babol Branch, Mazandaran, Iran, in 2013. She is currently pursuing the Ph.D. degree with Polytechnique Montréal, Montreal, QC, Canada. From 2009 to 2018, she was a full-time Lecturer and a Computer Laboratory Expert with Islamic Azad University, Isfahan (Khorasgan) Branch. Her research interests include forecasting traffic in vehicular networks using machine learning, deep learning, and ensemble learning models.

SAMUEL PIERRE (Senior Member, IEEE) received the Ph.D. degree (Hons.) from the Université du Quebec à Trois-Rivieres (UQTR), in May 2014, and the Ph.D. degree (Hons.) from the Université du Quebec en Outaouais (UQO), in November 2016. He is currently a Professor with the Department of Computer and Software Engineering, Polytechnique Montréal, 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. He also 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 2017, the Grand Prize for Professional Excellence from the Order of Engineers of Quebec (OIQ), in 2020, and the Gold Medal from Engineers Canada, in 2021.