sciendo

Transport and Telecommunication, 2022, volume 23, no. 4, 293–310
Transport and Telecommunication Institute, Lomonosova 1, Riga, LV-1019, Latvia
DOI 10.2478/ttj-2022-0024

DYNAMIC TRAVELING ROUTE PLANNING METHOD FOR INTELLIGENT TRANSPORTATION USING INCREMENTAL LEARNING-BASED HYBRID DEEP LEARNING PREDICTION MODEL WITH FINE-TUNING

Shridevi Jeevan Kamble[1], Manjunath R. Kounte[2]

School of ECE, REVA University
Bangalore, India
[1]Shridevi.havanoor@gmail.com, [2]Manjunath.kounte@gmail.com

Predicting the most favorable traveling routes for Vehicles plays an influential role in Intelligent Transportation Systems (ITS). Shortest Traveling Routes with high congestion grievously affect the driving comfort level of VANET users in populated cities. As a result, increase in journey time and traveling cost. Predicting the most favorable traveling routes with less congestion is imperative to minimize the driving inconveniences. A major downside of existing traveling route prediction models is to continuously learn the real-time road congestion data with static benchmarking datasets. However, learning the new information with already learned data is a cumbersome task. The main idea of this paper is to utilize incremental learning on the Hybrid Learning-based traffic Congestion and Timing Prediction (HL-CTP) to select realistic, congestion-free, and shortest traveling routes for the vehicles. The proposed HL-CTP model is decomposed into three steps: dataset construction, incremental and hybrid prediction model, and route selection. Firstly, the HL-CTP constructs a novel Traffic and Timing Dataset (TTD) using historical traffic congestion information. The incremental learning method updates the novel real-time data continuously with the TDD during prediction to optimize the performance efficiency of the hybrid prediction model closer to real-time. Secondly, the hybrid prediction model with various deep learning models performs better by taking the route prediction decision based on the best sub-predictor results. Finally, the HL-CTP selects the most favorable vehicle routes selected using traffic congestion, timing, and uncertain environmental information and enhances the comfort level of VANET users. In the simulation, the proposed HL-CTP demonstrates superior performance in terms of Mean Square Error (MSE), Mean Absolute Error (MAE), and Root Mean Square Error (RMSE).

Keywords: Vehicular Ad-Hoc Network (VANET), Traffic and Timing Dataset (TDD), Incremental Learning, Hybrid Learning Model, Real-time Traveling Route Prediction

1. Introduction

Due to the urbanization acceleration and the vehicle population, predicting real-time traffic congestion is a prominent issue in developing countries where human lives and economic status are highly suffered (Brockmann *et al.*, 2006). Also, traffic congestion poses several economic, social, and environmental problems. Road traffic congestion is defined as the amount of traffic flow density that exceeds the capacity of roads (Afrin *et al*., 2020). The road congestion creates an excess delay in the travel time of vehicles when they select highly congested routes for travel. In other words, the vehicles necessarily have to wait for a long time to select highly congested and longer travel routes, resulting in increased travel costs and time. Hence, it is crucial to predict and forecast the travel time information estimated based on real-time traffic congestion for the vehicles to reach the destination without delay (Xu *et al.*, 2017). Additionally, the poor weather condition also affects the comfort level of VANET users. In real-time, several factors like dynamic traffic congestion, unexpected events, and poor weather conditions impact traveling route selection. The traffic congestion and travel time prediction models with weather data assist the travelers in selecting the best routes based on their requirements and minimize travel costs (Lana, *et al.*, 2018; Shridevi *et al.*, 2022).

Accurate travel time prediction based on the dynamic traffic congestion level is crucial to direct the vehicles through appropriate paths. Numerous traffic congestion and travel time prediction methods are proposed in the literature (Afrin *et al.*, 2020). The conventional works are classified into statistical models (Wagner-Muns *et al.*, 2017) and machine learning-based models (Liu *et al.*, 2017). The statistical models are generally designed for pre-defined smaller datasets and they lack to handle the issues associated with complex time series vehicular data. To rectify such issues, machine learning-based models with realistic datasets are employed later in which the uncertainty of complex large-sized VANET information is addressed effectively (Tedjopurnomo *et at.*, 2020). However, a major drawback of

conventional methods is the lack of continuity in the learning process with real-time road information. Performing learning with a static dataset may not be appropriate, and they lack to accomplish high real-time prediction performance. The incremental learning or active learning model can alleviate such issues by updating the novel incoming real-time data with existing learning data optimally. Research works fuse the new data with a learning model in which the re-learning process takes long computation resources and time. Therefore, an efficient learning model is essential to predict the vehicle traveling routes realistically with more roadside parameters and enhance the VANET efficiency.

The main contributions of the proposed HL-CTP are described as follows.

- The prime intention of HL-CTP is to optimize the real-time traveling route prediction accuracy of vehicles without increasing the model complexity by utilizing incremental learning on a hybrid prediction model.
- The HL-CTP fuses the prediction results of various independent predictors of the hybrid model using Ameliorate Bayesian Combination Method (ABCM) to achieve a more reliable route prediction.
- To optimize the efficiency of the prediction model, the HL-CTP builds a novel TDD with benchmarking and synthetic information. The incremental learning algorithm updates the TDD with real-time road information in an optimal way.
- Instead of incrementally updating the real-time traffic data with massive TDD learning, the HL-CTP applies incremental learning on the n^{th} TDD subset during prediction and fine-tuning the prediction accuracy, resulting in high real-time performance.
- Finally, the experimental results demonstrate the superiority of the proposed HL-CTP in terms of various prediction error metrics, such as MSE, MAE, and RMSE.

2. Literature Survey

The works in (Yin *et at.*, 2020; Nagy *et al.*, 2018) offer a comprehensive survey on conventional traffic event prediction methods. Another survey in (Boukerche *et al.*, 2020; Li *et al.*, 2018) comprehensively explains different traffic prediction strategies and provides suggestions for machine learning-based traffic prediction for research. The research work in (Elfar *et al.*, 2018) presents three machine learning strategies employed for short-term congestion prediction based on the connected trajectory information of vehicles. Such work develops two different prediction methods that are offline and online. The offline strategies exploit historical information, whereas the online models are constructed using real-time vehicle information. An improved model in (Hosseinzadeh *et al.*, 2014) detects traffic congestions using the learning methods that utilize environmentally assisted vehicle mobility patterns and vehicle trajectory for prediction. The improvements are accomplished by appending preprocessing steps that neglect short-tracked sequences. The work in (Lanka *et al.*, 2014) rectifies diverse traffic-related issues based on deep learning neural network that exploits the GPS trajectory information of floating cars in congestion prediction. Consequently, the work in (Chen *et al.*, 2011) identifies the most popular traveling routes among the two locations. For that, the historical traveling behavioral patterns of multiple users are observed in advance.

The maximum probability product algorithm is utilized to discover the most popular routes efficiently. A novel floating car data analysis strategy (Xu *et al.*, 2013; Shridevi *et al.*, 2022) utilizes data cubes to explore the congestion pattern. The analysis model deviates from other conventional systems by utilizing the numerical statistics of the traffic information in scrutinizing the congestion pattern. It appends the event view or spatial-temporal progressions in traffic congestion measurements. Firstly, it discovers the traffic congestion event based on the multi-dimensional analysis in which the slow-speed road segments Spatio-temporal data is utilized. Secondly, it obtains different traffic patterns on diverse Spatio-temporal dimensions by employing the aggregated cluster style. Finally, the congestion pattern is predicted based on the aggregated location clustering data and timing information.

The research work in (Zhang *et al.*, 2016) presents a comparative study on various traffic forecasting methods based on multivariate short-term freeway traffic flow with missing data. A novel travel speed prediction mechanism has been introduced in (Tang *et al.*., 2017). The prediction model utilizes evolving fuzzy neural network in which the RTMS information collection method is used to gather the travel speed. The learning model uses the Takagi-Sugeno fuzzy inference strategy in which the fuzzy rules are employed to estimate the appropriate output vectors to the corresponding input travel speed data. The work in (Ran *et al.*, 2016) presents a tensor completion framework to compute the missing traffic state information. In the framework, the placed tensor straightly derives the entries from the floating cars, and it models the traffic states using the retrieved information.

Furthermore, the framework decides that the unobserved traffic states of tensors are missing entries. Deep learning-based multitask learning Gated Recurrent Unit (MTL-GRU) has been proposed in (Zhang *et al.,* 2020) with residual mappings. The MTL-GRU incorporates feature engineering to choose the most informative features related to forecasting, and it maximizes its performance efficiency. A multi-step prediction model named Spatial-Temporal Attention Wavenet (STAWnet) has been proposed for effective traffic condition forecasting (Chenyu Tian *et al.,* 2021). It manages the long-time sequences by effectively capturing the complex Spatio-temporal dependency information based on a self-attention network. Artificial intelligence-based traffic congestion models are reviewed in (Akhtar *et al.,* 2021). The work in (Kong *et al.,* 2016) utilizes the floating car trajectory data to predict the urban road traffic congestion efficiently. Such work models the floating cars as mobile sensors and can probe the realistic urban road traffic information.

A pattern matching-based highway travel time prediction algorithm has been proposed in (Zhao *et al.,* 2015; Shridevi *et al.,* 2019). Initially, it establishes a pattern library by selecting the traffic volume and speed information. Further, the prediction algorithm estimates the mean speed and travel time by exploiting the two-dimensional linear interpolation strategy. Consequently, the research work in (Servos *et al*., 2019) evaluates the long-term forecasting-based machine learning to compute the multimodal freight travel time. The transport goods have some tracking sensors for data generation. Finally, the prediction model constructs many travel time prediction models based on various features obtained from the tracking sensors. A novel method named Traffic Pattern centric Segment Coalescing Framework (TP-SCF) has been introduced (He *et al.,* 2020). For predicting the travel time of bus journeys, the TP-SCF relies on learned traffic conditions disparate patterns of diverse bus line segments. Especially the TP-SCF consists of two stages for travel time prediction that is training and prediction. During the training stage, the TP-SCF separates the bus lines into bus line segments. The Non-negative Matrix Factorization (NMF) is employed to explore the common travel time patterns of bus line segments. The TP-SCF clusters the bus line segments with the same patterns and extracts the data records from the coalesced clusters to train the Long Short-Term Memory (LSTM) model and predict the bus journey time. A system in (Petersen *et al.,* 2019) proposes a bus travel time prediction model that uses the dynamic Spatio-temporal correlation information of urban bus networks for travel time prediction. It is a hybrid model, as the neural network employs both convolutional and LSTM for prediction. A novel framework in (Wu *et al.,* 2020) presents a hybrid data-driven travel time prediction method for speculating the bus journey travel time. Adopting convolutional LSTM and self-attention strategy effectively predicts the time based on the waiting time of junctions. Thus, the model is highly robust as it can capture long-range information easily.

The short-term traffic flow prediction model in (Zhang *et al.,* 2019) utilizes the Spatio-temporal analysis model to select the features and constructs a prediction model using CNN deep learning model. The research work (Wen *et al.,* 2019; Helen *et al.,* 2022) identifies the traffic environments based on rules designed using genetic algorithm-based temporal association. Further, the classification algorithm categorizes the traffic congestion level according to the rule mining. The work in (He *et al.,* 2019) determines the traffic congestion at turn-level intersections using low-frequency probe vehicle data. The road network is divided into various meter-scalar cells to correctly identify the turn-level road intersections and analyze the traffic level at each intersection. A traffic congestion prediction model in (Zhao *et al.,* 2019) determines the traffic congestion level using bus driving time and the LSTM model. It exploits the bus station data and GPS-based trajectory information to calculate the actual and standard bus driving time data. The diverse traffic conditions such as morning, evening, and daytime are considered (Chen *et al.,* 2014). The research work in (Chen *et al.,* 2020) uses the RSUs to direct the EVs through minimum travel time routes by escalating the average speed to avoid congested areas. The RSUs continuously sent congestion information to the network simulator, instructing EVs to follow the shortest routes. The research work in (Chen *et al.,* 2019) proposes an improved deep belief network (DBN) for bus travel time prediction. The DBN is constructed based on Gaussian–Bernoulli restricted Boltzmann machines.

Further, the classical DBN is updated with continuous information. Finally, a back-propagation (BP) neural network is employed for performance enhancement. A discrete and continuous combined analysis method has been presented to accomplish an enhanced long-term travel time prediction (LTTP) of connected vehicles (Qi *et al.,* 2021). For that, the LTTP integrates the non-negative tensor factorization to retrieve the dynamic speeds of vehicles effectively. Also, it employs the Gaussian mixture regression model to handle the continuous VANET factors. Thus, the LTTP combines various discrete and continuous computation factors and incorporates multi-criteria factors such as type of vehicle, days, time, weather conditions, driver differences, and travel distance in travel time prediction. Consequently, the research work in (Sun *et al.,* 2021; Naveen *et al.,* 2021) presents LSTM neural networks based on two

different deep learning algorithms, such as hybrid LSTM and sequential LSTM with a self-attention strategy. Such models effectively obtain the location patterns and time trajectory information with their dependencies for simultaneously predicting the next location and travel time of vehicles. Similarly, the work in (Adewale *et al.*, 2020) develops two various neural network strategies referred to as multi-layer perceptron and LSTM for travel time prediction. The work in (Bogaerts *et al.*, 2020) exploits deep neural networks and LSTM to predict the long and short-term traffic flow predictions. The deep neural network continuously extracts the spatial traffic features from the corresponding environment, and the LSTM is used to extract the temporal traffic features. Moreover, such a model predicts long-short traffic flows and mitigates congestion.

2.1. Motivation and Problem Statement

Traffic congestion in urban areas is undesirable as it escalates the driving inconveniences, vehicular emissions, fuel consumption level, and vehicle operating costs by slower downing the public and private vehicles. By navigating the vehicles through alternate traveling routes, the transportation system minimizes the negative effects of congestion caused due to the high vehicle population. Efficient route prediction models are prominent to increase the driving comfort level of vehicular users by predicting the road status in advance. Therefore, most existing research utilizes machine learning-based models to predict less congested and shortest traveling routes. However, there are some drawbacks to the existing works. Firstly, most of the existing works use a limited number of prediction parameters to discover efficient traveling routes, and they lack in considering dynamically varying auxiliary information that has a high impact on vehicle route selection. Secondly, the conventional works do not update real-time traffic data with static learning efficiently and diminish the real-time prediction performance efficiency. By heeding the issues associated with existing solutions, this paper proposes an efficient route prediction model that exploits the incremental learning model to real-time data update and a hybrid learning model to predict the status of the approaching route.

3. System Model

The VANET is modeled as a two-dimensional network graph G (V, R). The term V denotes the number of VANET entities which may be normal vehicles and Road Side Units (RSUs). The term R defines r number of regions of graph G. Generally, the road network is wide and comprises numerous streets with many obstacles. Initially, the HL-CTP represents mth vehicle as targeted vehicle v among the V, where v€{V = V1, V2, ….Vm}. For modeling the traveling route prediction-based VANET system, the HL-CTP divides the road network into multiple sub-regions and collects the spatial and temporal related data from all regions through various monitoring devices like roadside and vehicle sensors to construct the TDD. The TDD is represented with i number of rows and j number of columns, TDD = $\{x_i, y_j\}$ and the data of TDD is collected over a certain period (T). The TDD comprises both spatial and temporal related information like road width (RW), height (H), and uncertain features like weather (W) and unexpected Collisions (C). The road congestion of TDD is defined in tensor form $TDD(C) \in RC^{T*FL*RW*H*W*C}$, where T is the time, and FL refers to the flow type (in and out). To predict the most appropriate route among the source S and destination D, the HL-CTP considers the traffic and timing data with uncertainty information and predicts the best routes using a hybrid and incremental learning algorithm. The symbols used by HL-CTP are listed in Table 1.

3.1. Problem Formulation

Traffic congestion and timing predictions are key elements in building the vehicle route planning system. Generally, the travel time of the road is a static element, and it becomes a dynamic element when it is calculated based on the road congestion parameter. It is not easy to estimate the road travel time rapidly and accurately based on the variances in road congestion. The scale of road congestion information is large, and it has complex characteristics in real-time situations. The prediction model is also suffered from various external uncertainty factors of the road like sudden changes in weather, unexpected events, and peak hours. The rapid changes in traffic congestion and uncertain factors greatly affect the prediction performance of the learning algorithms. Integrating the current traffic data with benchmarking datasets necessitates high hardware requirements and computation time. The traffic data of benchmarking dataset is a large scale, and it is very expansive to append the current data. To handle such an issue, the proposed HL-CTP divides the

constructed TDD dataset into multiple subsets based on the time interval and updates the incremental data with such sub-datasets, resulting in optimized real-time prediction accuracy. In the congestion and timing-based traveling route prediction problem, the historical congestion data at time t-n $C_{t-n}^{t-1} = \{C_{t-1}, C_{t-2}, \ldots . C_{t-n}\}$ and current congestion data at time t, $C_t^{t-k+1} = \{C_{t-k+1,} C_{t-k+2,} \ldots . C_t\}$ is used to predict the future congestion data at time t+n, $C_{t+n}^{t+1} = \{C_{t+1}, C_{t+2}, \ldots . C_{t+n}\}$. Using the tensors of historical and current traffic flows at t-n and t time intervals, the HL-CTP formulates the future traveling route prediction problem at time t + 1.

$$C_{t+n}^{t+1} = \delta\left(\theta \middle| \left(C_{t-n}^{t-1} * C_t^{t-k+1}\right)\right). \quad (1)$$

In equation (1), the term $\delta(.)$ represents the real-time congestion prediction model, and the term θ is the model parameter. By solving the C_{t+n}^{t+1}, the HL-CTP predicts the most favourable realistic routes for the vehicles.

Table 1. Symbols used by HL-CTP

Symbols	Definition
G (V, R)	Two-dimensional network graph
V	Number of Vehicles
T	Time Period
RW	Road Width
H	Height
W	Weather
C	Unexpected Collision
FL	Flow Type (Vehicle in and out)
S	Source
D	Destination
C_{t-n}^{t-1}	Historical congestion data at time t-n
C_t^{t-k+1}	current congestion data at the time t
C_{t+n}^{t+1}	Future congestion data at t+n
θ	Model parameter
$\delta(.)$	Real-time congestion prediction model
σ	Pearson correlation coefficient
x_t	Input variable set
y_t	Output variable set
ε_t	Unobserved noise variable
z_t	Update gate
r_t	Reset gate
c_t	Traffic congestion at the time t
o	Hadamard product of two matrices
φ	Standard logistic sigmoid function
c_{t-1}	Previous candidate hidden states of GRUNN
$\bar{c}_t$	Current candidate hidden states of GRUNN
b_o	Bias
F	Number of Most Informative Features as Neurons
$\varphi(X_t, c_i)$	Radian basis function
ω_i	Weight of i^{th} feature
p_t^k	k^{th} predictor weight at a time t
K	Number of sub predictors
T_p	Travel time prediction
F_R	Best route position
T_a	Actual time value
T_d	Delay time value
M	Memeplexes
R	Number of routes among S and D
R_b	Route with the best position
R_w	Route with worst position
R_g	Global best traveling route
A	Time boundary of EV

4. Principle and Proposed Methodology overview

The principle of the HL-CTP is to plan dynamic optimal traveling paths with minimum traffic congestion and short travel time for vehicles. The HL-CTP considers the GPS historical and realistic traffic flow pattern as the main contributing input factors in a hybrid learning-based dynamic traveling path prediction model. The historical traffic patterns are collected using historical traffic flows, and the realistic traffic flows are collected using the incremental data update model. Figure 1 sketches the block diagram of the proposed methodology. Initially, the HL-CTP constructs a TDD by collecting the current and historical congestion and timing data from the GPS satellite through the internet. The TTD consists of numerous informative and non-informative features related to traffic congestion and timing prediction. Taking the retrieved raw TTD with a hybrid learning model increases the learning time and minimizes the effectiveness of the machine learning algorithm. The proposed work preprocesses the raw TTD and selects the most informative features for prediction to rectify such issues. The correlation analysis method identifies the correlation between the current and historical data of TTD. Consequently, the TTD with the most informative features is provided as input to the hybrid prediction model that employs three different predictors, VARMAX, GRUNN, and RBFNN, for effective prediction. Finally, the Ameliorate Bayesian Combination Method (ABCM) combines the output of three predictors to take an effective prediction-based decision. The incremental data of current traffic flows with uncertain data are exploited to fine-tune the hybrid model with ABCM and achieve high real-time performance efficiency. Thus, the hybrid model avoids the issues of inaccurate forecasting of a single learning algorithm with noisy traffic information and maximizes the prediction accuracy by taking advantage of different learning algorithms. Moreover, the Improved Shuffled Frog Leaping (ISFL) algorithm selects the best optimal traveling route for less traffic congestion and the shortest timing from the available routes for efficiently navigating the vehicles to the desired destination.

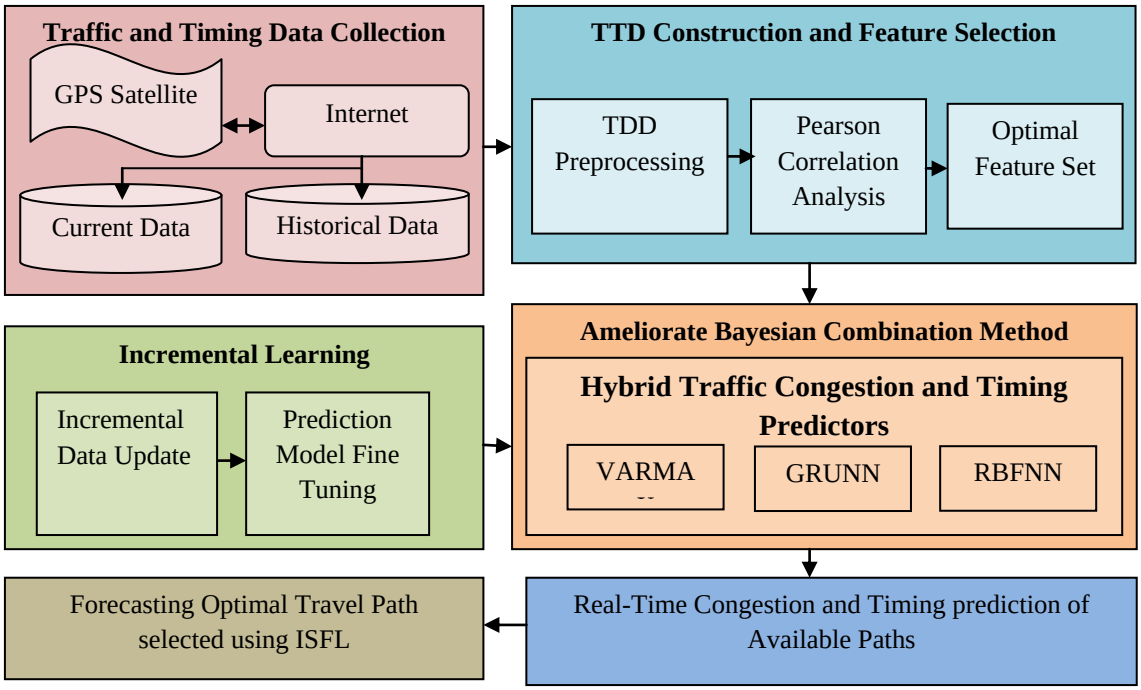


Figure 1. Overview of Proposed Methodology

4.1. TTD Construction and Feature Selection

Generally, many issues are associated with employing a benchmark dataset in traffic congestion prediction. Firstly, most of the dataset comprises little traffic-related information, and there is a minimum number of most informative features. Utilizing such datasets for traffic congestion prediction leads to inaccurate traffic prediction. Secondly, the benchmark dataset mainly utilizes long time intervals of 5 minutes to 10 minutes for traffic data collection. The long-time interval-based dataset desperately affects

the accuracy of the prediction model, resulting in ineffective traveling path selection. Therefore, the prediction model accuracy mainly depends on the advantages of the dataset. It is crucial to gather large traffic information within short time intervals. The proposed work constructs a novel dataset TTD is constructed using multiple traffic congestion parameters. The TTD construction is performed by collecting the two-way road traffic information of 308 urban roads of Xi'an city observed within 2 minutes time interval (Hou *et al.,* 2021). The proposed work uses the python language to get a series of JSON formatted traffic congestion and timing information since the data is updated every two minutes. Further, regular expressions are employed to process the original JSON format, and the TTD is constructed based on the obtained useful traffic data. The content of traffic information in TTD is extensive, encompassing traffic flow, traffic facilities, and uncertain data.

The collection of fundamental traffic information, traffic flow, and speed are the most informative data for predicting traffic congestion and timing. Except for the most informative features, the redundant and non-informative information is neglected through preprocessing and feature selection. Moreover, a complete CSV file for the proposed prediction model is generated, and the TTD is constructed successfully. After the construction of TDD, the data of TDD is incrementally updated over a specific t time interval based on the realistic observation of road congestion information using GPS.

Preprocessing: A significant amount of erroneous and redundant information is collected during TTD construction. Employing the collected raw information directly in the prediction model increases the learning time and shrinks the accuracy level of predictors. Also, the collected data consists of numerous informative and non-informative features. Hence, the non-informative features increase the burden of predictors, which are not essential for predicting traffic congestion and timing. Therefore, data preprocessing and feature selection are necessary before inputting the TTD features into the learning model.

Pearson Correlation Analysis Method: The retrieved raw TTD comprises the attributes like city, emergency area, street name, traffic status, weather conditions, driving angle, speed, and time. Hence, all features do not impact traffic and timing prediction. Therefore, it is essential to choose the informative features that highly impact the prediction model. The proposed HL-CTP utilizes the Pearson correlation analysis method, a familiar similarity measurement for finding the dependency level between two features. If the dependency between two features is high, the Pearson correlation coefficient value is high. Otherwise, it is low. For effective feature selection, the HL-CTP utilizes the Pearson correlation coefficient (σ) threshold of 0.5. If the σ value is 1, the two features are highly correlated. If it is zero, then the features are not correlated. Otherwise, the Pearson correlation coefficient value is above 0.5, and the features are partially correlated. The proposed work selects the features with the Pearson correlation coefficient value above 0.5 as the most informative features.

4.2. HL-CTP Model Learning

The HL-CTP exploits three types of predictors, VARMAX, GRUNN, and RBFNN, to predict traffic congestion. Further, it utilizes the ABCM method for taking the final traffic congestion and timing prediction results based on the results of the three predictors and timing data. To effectively predict the optimal traffic congestion and timing results, the HL-CTP exploits three sub predictors that are VARMAX, GRUNN, and RBFNN, and decides the final prediction using the ABCM model. The Prediction and route selection Model of HL-CTP and ISFL Route Selection is shown in Figure 2.

VARMAX: The VARMAX is the vector autoregressive moving-average model with exogenous variables. In VARMAX, the term y_t is the output variable set, $y_t = (y_{1t}, \ldots . y_{kt})'$ and the term x_t is the input variable set $x_t = (x_{1t}, \ldots . x_{rt})'$ that influences the output variable set y_t. The variables are determined outside of the system. Since the input variable set refers to independent, input, predictor, and exogenous variables, the output variable set refers to dependent and endogenous variables. Moreover, the unobserved noise variable set, $\varepsilon_t = (\epsilon_{1t}, \ldots . \varepsilon_{kt})'$ is estimated using the vector white noise process.

GRUNN: The GRUNN is a gated recurrent unit neural network capable of learning long-term traffic information and finding possible future prediction values. The GRUNN model structure comprises three layers such as input, output, and hidden. The input layer denotes a set of input variables x_t and the output layer consists of the set of output variables y_t. The basic unit of the hidden layer is a memory block. The memory block includes two types of gates such as update z_t and reset r_t. The GRUNN comprises three types of layers that are input, hidden, and output to predict the congestion information.

The fundamental units of the hidden layer are the GRU memory block, which consists of update and reset gates. The update gate handles the updated hidden layer information, and the reset gate manages the information that can be ignored. The GRUNN in TTD speculates the road traffic using the following equation (2).

$$C_t^{t-k+1} = \varphi\left(W_o\left((1 - Z_t)\ o\ c_{t-1} + z_t\ o\ \bar{c}_t\right) + b_o\right). \quad (2)$$

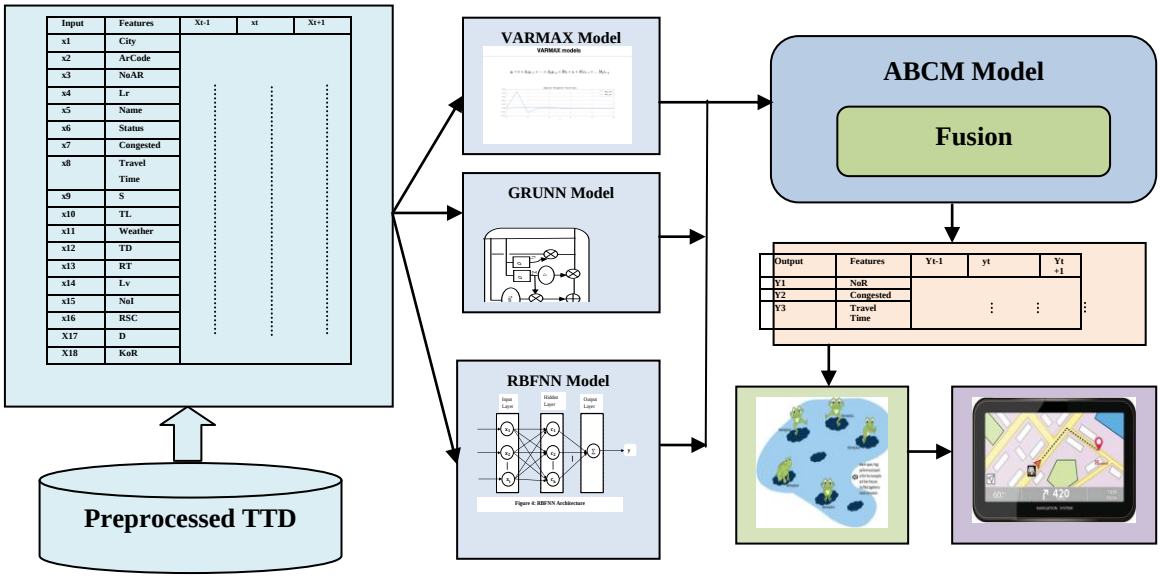


Figure 2. Hybrid Predictors and ISFL Route Selection

In equation (2), the term C_t^{t-k+1} is the traffic congestion at t time, and the symbol o refers to the Hadamard product of two matrices. The term φ is the standard logistic sigmoid function and the term W_o is the weight matrix. The term b_o is bias. Finally, the terms c_{t-1} and $\bar{c}_t$ are previous and current candidate hidden states of GRUNN, respectively. Finally, the prediction value of GRUNN is the second predictor input to the ABCM of TTD.

RBFNN: The RBFNN is the radial basis function neural network, a multi-layer machine learning algorithm used for traffic congestion prediction. The layers of the RBFNN are input, hidden with a non-linear RBF activation function and a linear output. The output of the RBFNN N C_t^{t-k+1} is a scalar function of the input vector estimated using equation (3).

$$C_t^{t-k+1} = \sum_{i=1}^{F} \omega_i * \varphi(X_t, c_i), \quad (3)$$

where F – Number of Most Informative Features as Neurons,
$\varphi(X_t, c_i)$ – Radian basis function,
ω_i – Weight of i^{th} feature.

According to the RBFNN structure, the inputs are coupled with a hidden layer of neurons. Further, the hidden layer is connected with an output layer for computing the final prediction value. Furthermore, the output of RBFNN is provided as the third input to the ABCM of TTD. The TDD instructs the ABCM to combine the results of VARMAX, GRUNN, and RBFNN and take the final traffic congestion and timing prediction results.

ABCM-based Prediction The proposed ABCM-based prediction model utilizes three sub predictors, VARMAX, GRUN, and RBFNN, to optimize the vehicle path prediction efficiency. The ABCM fuses the road traffic predictions of three sub-predictors based on the conditional probability and Bayes rule. Since the VARMAX and RBFNN models efficiently capture the non-linear road characteristics within a short-term time interval. Further, the GRUNN model deals with the massive road traffic information collected using satellite-based GPS networks. Thus, the ABCM maximizes the prediction accuracy of single predictors by

considering the prediction output of three various sub predictors. Also, the ABCM includes an error calibration coefficient with the fundamental BCM to solve the issues associated with the noisy traffic data of the training set. For fusing the sub predictor results and taking efficient prediction decisions, the ABCM model employs posterior probability to select the best k^{th} sub predictor results as the final output, which is very powerful in road traffic decision making. The following equation (4) demonstrates the posterior probability value of the k^{th} sub predictor at time interval t.

$$p_t^k = \frac{P(c_t, Y=k|\, C_{t-n}^{t-1}, \ldots\ldots C_t^{t-k+1})}{\sum_{a=1}^{K} P(c_t, Y=a|\, C_{t-n}^{t-1}, \ldots\ldots C_t^{t-k+1})} * T_p, \quad (4)$$

where p_t^k – k^{th} predictor weight at a time t,
K – Number of sub predictors,
c_t – Congestion event at time t,
T_p – Travel time prediction.

The ABCM estimates the T_p based on the actual time value and the tome delay due to weather, current traffic information, and unexpected events. After getting the weights of the sub-predictors based on the congestion event at time t, the final output of traffic congestion and travel time prediction at time interval t+n along a traveling route is speculated using the following equation (5).

$$C_{t+n}^{t+1} = \lambda_t * p_t^K\left(\sum_{a=1}^{K}(\, C_{t-n}^{t-1})_a * C_t^{t-k+1}{}_a * T_p\right). \quad (5)$$

In equation (11), the term λ_t refers to the error calibration coefficient value lies between -1 to 1. The error calibration coefficient factor minimizes the errors due to noise during training set construction and elaborates the prediction accuracy of sub predictors. The term p_t^K is dynamically adjusted based on the historical prediction performance of the sub-predictors, resulting in better performance.

4.2.1. Incremental Learning on HL-CTP Prediction

After the pre-learning process of HL-CTP, incremental learning is applied to hybrid prediction with ABCM to fine-tune the model parameters. In HL-CTP, incremental learning is adjusting the learning parameters of the pre-learned model with novel data. The incremental learning iteratively optimizes the accuracy of the hybrid model by obtaining typical samples of real-time road data with normal and uncertain information, as shown in Figure 3. Generally, the proportion of real-time data with different uncertain characteristics contributes to the prediction model, especially unexpected events like weather and collisions on the roadside. The incremental learning model continuously updates such typical samples related to real-time road congestion status, weather data, traffic flow type, and unexpected events with the current prediction model. The incremental learning model initially updates all road-related information through active learning, and the HL-CTP obtains the most valuable real-time prediction data to fuse them with TDD. VARMAX is a statistical model among the sub predictors, and the other two are deep learning models. The incremental learning algorithm in HL-CTP only extends the prediction knowledge output of the VARMAX. It actively learns the VANET environmental data such as road congestion, weather, and uncertain data and incrementally adds the data with the already learned TDD subset. Thus, it creates changes in the dataset records only instead of the VARMAX process. Further, valuable real-time data is used for the incremental model update. Selection of the most valuable real-time incremental data minimizes the dataset size and re-learning time while meeting high real-time prediction performance.

Model Fine-Tuning: The deep learning models with only historical traffic flows may minimize the real-time performance of the predictor. Also, it increases the computation resources and learning time with long-term historical traffic flows. Hence, it is crucial to fine-tune the learning model with realistic traffic data. The HL-CTP uses incremental learning based on current traffic flow updates to fine-tune the model performance C_{t+n}^{t+1} of predictors and enhances the real-time performance efficiency. Fusing the current traffic data with massive historical data increases the learning time and builds a complex structure. The incremental data is updated with n^{th} sub TDD to optimize the model performance. The HL-CTP fuses the current data update incrementally with the nth TDD subset minimizes the learning complexity, and also increases the efficiency of the model. Finally, the model parameters of C_{t+n}^{t+1} are efficiently adjusted based on active learning data and achieve better real-time prediction performance.

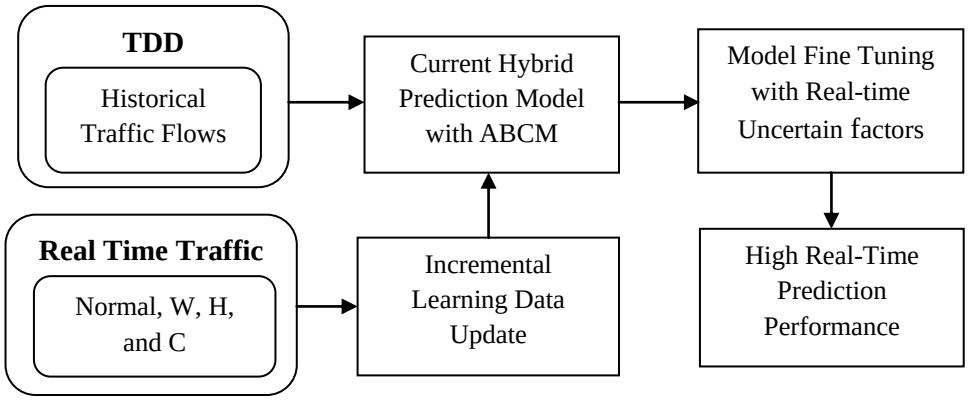


Figure 3. Incremental Learning on HL-CTP Prediction Model

4.3. ISFL based Optimal Travelling Route Selection

It is necessary to select the shortest and most reliable traveling routes for travelers to prevent the impairments of traveling costs and inconveniences. The ABCM exploits the ISFL algorithm in which the congestion prediction results of R traveling routes are considered as frogs of ISFL. Further, the best frog is selected as the final optimal route for the journey. According to the ISFL algorithm, the ABCM takes the required journey time of vehicles as an acceleration factor α of ISFL. Thus, the term α has a major contribution to selecting the best routes without incurring high traveling delays and complexity. The steps involved in ISFL-based optimal route selection are as follows.

Step 1: Generation of virtual road population: Construct a D-dimension space by considering the number of traveling routes among a source and destination as frogs. Each road has a position p in the feasible space.

Step 2: Route fitness estimation: Estimate the fitness value of each frog based on the road traffic congestion level and timing information obtained using ABCM. Sort the routes in descending order by taking into account the fitness value. Find out the best frog position as the best route position among the available routes.

$$F_R = C_{t+n}^{t+1} * (T_a + T_d). \quad (6)$$

In equation (6), the term C_{t+n}^{t+1} is the predicted congestion level of a route R at t+n time. The terms T_a and T_d represent actual and delay time values of route R.

Step 3: Partitioning of Roads: Divides the entire road population into multiple memeplexes (m) in which each m has several routes r between S and D, where R = m*r. In this step, rank 1 goes to m 1, and rank 2 goes to m 2. Similarly, rank r goes to m, m, and rank r+1 goes to m 1.

Step 4: Memetic estimation of best routes: The ABCM evolves each m over a specific iteration to determine the shuffled m. Further, the routes R with best (R_b) and worst (R_w) positions are identified. The global best traveling route (R_g) is identified for further improvements. The ABCM algorithm improves the best route position based on the acceleration factor. The acceleration factor is actual travel requirement of travelers to reach the destination. The position of R_b is adjusted using the following equation.

$$\Delta R = \alpha * \text{rand}\,(0,1) * (R_g - R_w), \quad (7)$$

where α is the time boundary of vehicle arrival and the term $\text{rand}\,(0,1)$ is a random number that lies between 0 to 1. The new position is determined as follows.

$$N = R_b + \Delta R. \quad (8)$$

Likewise, the position of routes is changed based on the road congestion level and travel time of routes.

Step 5: Shuffle meme flexes for optimal route selection: In this step, the ABCM shuffles the meme flexes after the optimal number of evolutionary steps. The routes are rearranged in descending order according to the fitness value. Further, the ABCM selects the R_b as the best final route for EVs. Moreover, the utilization of ISFL in ABCM effectively solves the dynamic path planning of the vehicles

and selects the optimal route by considering the road traffic and travel time information with uncertain factors. The entire process of HL-CTP is described in the algorithm (1).

//HL-CTP Design Algorithm//

Function: Less congested Shortest Traveling Path Selection
Input: Historical Traffic and Timing Data, Incremental data, Hybrid Predictors, and ISFL path selection Algorithm
Output: Real-time route selection for Vehicles.

```
EVs do {
    Initialize the HL-CTP;
    Collect the historical and current traffic and timing data from GPS satellites;
    Constructs the Raw TTD;
    TTD Preprocessing based on Euclidean distance measurement;
    Correlation Analysis based Optimal feature Selection;
    Feeds the features to the ABCM;
    ABCM do {
        Inputs the feature set with data points to three sub predictors;
        VARMAX do {
            Estimates the traffic and timing prediction to the input variable set;
            Feeds the predicted result as first input to the ABCM;
            }
        GRUNN do {
            Computes the prediction value with multiple layers using equation (2);
            Feeds the result as second input to the ABCM;
            }
        RBFNN do {
            Predicts the traffic congestion using equation (3);
            Feeds the predicted result as third input to the ABCM;
            }
        Computes the final prediction results using equation (4) and (5);
        Applies incremental learning to optimize the HL-CTP prediction model parameters;
        Active learning with uncertain information optimizes the predictor results;
            }
    ISFL based optimal route selection;
    }
```

Algorithm 1. Protocol Design of HL-CTP

5. Experimental Evaluation

The experiments are carried out to analyze the efficacy of the proposed HL-CTP. The HL-CTP is an intelligent application layer protocol, and there is no effective benchmarking dataset with traffic and timing data combination. To rectify the issues of inaccurate and unconvincing results with existing datasets, the proposed HL-CTP develops a novel dataset TTD. The TTD is an extension of the Xi'an Road Traffic Dataset (Hou, Y *et al*., 2021), whereas it includes more auxiliary information than the Xi'an Road Traffic, which is very powerful in maximizing the prediction performance. The dataset used in this paper is TTD captured from the GPS satellite system every two minutes interval. In HL-CTP, the temporal resolution depends on the orbital characteristics of the GPS satellite to obtain real-time vehicular data. The proposed TDD is a small data set that comprises thousands of data. The features of the TTD are nearly equal to 27, which are listed in table 2. The features observed by the incremental learning in HL-CTP are similar to those defined in table 2. The HL-CTP exploits 70% of TDD data for learning and 30% for testing. The HL-CTP utilizes incremental learning-based data augmentation method to increase the size of TDD. The incremental model actively learns the real-time VANET data and improves the prediction accuracy of learning models. The proposed HL-CTP takes the prediction horizon as 15 minutes. It normalizes the TDD data into [0,1] before feeding the TDD to the predictors. It exploits long forecasting horizons to measure the error metrics that have a high impact on route prediction. The proposed model decides the rolling window size as half of the time series horizon for accuracy estimation. It makes the accuracy estimation smooth by taking the rolling window size as large. The HL-CTP obtains a series of traffic and timing data from the Xi'an Road Traffic dataset to construct TDD and preprocesses the TTD using python libraries. The experiment is implemented on a computer with Intel i5 2.5GHZ CPU

and 16 GB memory. The python libraries are used to evaluate the performance of VARMAX, GRUNN, and RBFNN. Further, the predictor results are fused by ABCM, and ABCM obtains the final prediction output. Consequently, the ISFL algorithm selects the dynamic suitable traveling routes for vehicles. The parameters of ISFL are frog group scale, group number, and the number of iterations.

Table 2. Features of TTD

S.No	Fields	Description	Possible Values
1	City	City Name	XiAn
2	ArCode	Area Code	610100 to 610200
3	NoAR	Number of available roads among a source and destination	2 to 10
4	Name	Road segment ID	1,87,18,81,89,190
5	Status	Road Congestion Level	Congestion free-0; Less Congestion-1; Medium Congestion - 2; Severe Congestion-3
6	Time	Time taken to cross a road with maximum speed	Varied based on road length and vehicle speed
7	S	Vehicle speed	45 to 60 Km/hr
8	RL	Road Length	1 to 2 Km
9	Weather	Weather phenomenon	Sunny, Rainy, and Snow
10	Temp	Realistic Temperature	10 to 20
11	WS	Wind Speed	10 to 20 m/s
12	WD	Wind Direction	Same or Opposite
13	Hum	Humidity of Air	60 to 100
14	Speed	Speed of Vehicles	Slow, Medium, and High
15	Angle	Driving Angle	30 to 45 degree
16	Polyline	Road Coordinates	108.94725
17	TD	Traveling Direction	Same or Opposite
18	DCI	Data collection Interval	2 minutes
19	NoTTR	Number of Traffic and Timing Records	263815
20	RT	Road Type	Simple or Complex
21	AS	Application Scenario	City or Highway
22	NoV	Number of Vehicles on the road	Small, Medium, High
23	NoI	Number of intersections	Small, Medium, High
24	RSC	Road Surface Conditions	Dry, Wet, Snow, and Icy
25	D	Distance between Vehicles	Short, Medium, and Long
26	TD	Type of the day	Working Days or weekend
27	KoR	Kind of Roadways	Traffic Circle, One-way, Two-way, and Single or Multi-lane

5.1. Prediction Error Metrics

The experimental results of HL-CTP are obtained in terms of error metrics such as MSE, MAE, and RMSE.

MSE: It is the mean or average of the square of the difference between actual and predicted output values.

$$\mathrm{MSE} = \frac{\Sigma(y_a - y_p)^2}{n}. \tag{9}$$

The terms y_a and y_p refer to the actual and predicted output values. The term n denotes the number of observations.

MAE: It is the ratio of errors to the number of observations. The errors are the difference between actual and predicted values.

$$\mathrm{MAE} = \frac{|y_a - y_p|}{n}. \tag{10}$$

RMSE: It is the root value of MSE. The RMSE is the numeric value of the prediction performance efficiency of a predictor.

$$\mathrm{RMSE} = \sqrt{\frac{\Sigma(y_a - y_p)^2}{n}}. \tag{11}$$

5.2. Experimental Results

Figure 4 compares prediction performance results in terms of MSE, MAE, and RMSE of three predictors such as VARMAX, GRUNN, and RBFNN, with and without Incremental Learning (IL). Each predictor accomplishes different error rates during prediction. The error rate results are tabulated in Table 3. Figure 4(a) demonstrates that the VARMAX attains 0.52, 0.63, and 0.72 of MSE, MAE, and RMSE values before fine-tuning the hybrid model. However, the prediction error rate of VARMAX is minimized via model fine-tuning with IL. For instance, the VARMAX attains 0.18, 0.33, and 0.46 error rates of MSE, MAE, and RMSE with IL. Similarly, Figures 4(b) and 4(c) show that the MSE, MAE, and RMSE of GRUNN and RBFNN are 0.07, 0.21, 0.26, and 0.06, 0.16, 0.23, respectively, for the scenario of prediction without IL. Hence, the GRUNN and RBFNN outperform the VARMAX under all situations, as they exploit deep learning structures to predict traffic congestions. The deep learning model utilizes multiple hidden layers and reveals optimized prediction output values. For example, the MSE, MAE, and RMSE results of ABCM are 0.04, 0.12, and 0.16, respectively, for prediction without the IL scenario. The ABCM accomplishes the error rates and improves accuracy by fusing the results of three predictors. However, the prediction models without IL minimize the prediction performance efficiency, as they perform predictions only with static TDD. The IL fine-tunes the model learning by updating the incremental data through active learning. Thus, it optimizes the prediction accuracy considerably. For example, the prediction error rates of MSE, MAE, and RMSE of GRUNN and RBFNN are 0.03, 0.09, 0.14, and 0.02, 0.08, 0.14, respectively, when applying IL on prediction. Moreover, the RBFNN outperforms the GRUNN and VARMAX in terms of MSE, MAE, and RMSE from the results of Figure 4. The reason is that the RBFNN model has strong tolerance against input noise, and it is flexible with large size data when compared with GRUNN and VARMAX. For example, the RBFNN accomplishes 0.16 and 0.08 MAE error rates for without and IL scenarios. Thus, it increases the prediction accuracy of HL-CTP and neglects the vehicles to enter into high bottleneck segments. It previously instructs the vehicles to dynamically select another route for traveling and avoids the highly congested bottlenecked routes through advance prediction. Based on the MSE, MAE, and RMSE results of sub predictors with IL, the ABCM achieves 0.015, 0.06, and 0.09 of MSE, MAE, and RMSE, respectively, as depicted in Figure 4 (d).

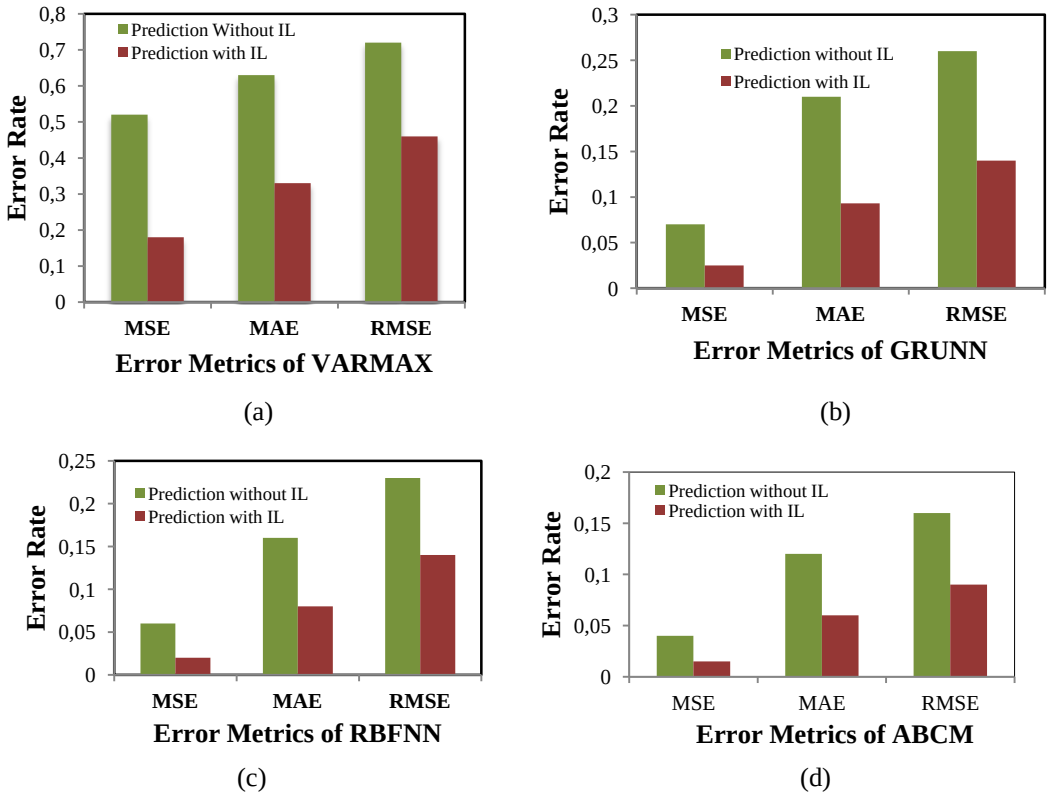
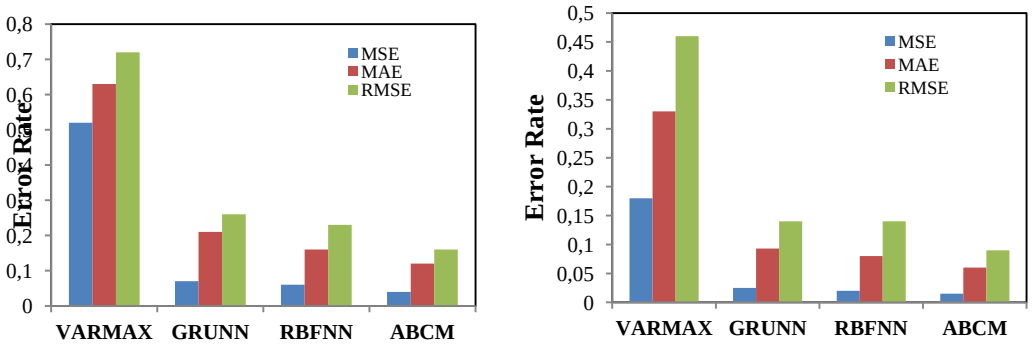


Figure 4. Prediction Performance Results of VARMAX, GRUNN, RBFNN, and ABCM

Table 3. Error Rate Results of VARMAX, GRUNN, and RBFNN

Error Rate	Predictor Results							
	VARMAX		GRUNN		RBFNN		ABCM	
	Prediction without IL	Predic-tion with IL	Prediction without IL	Prediction with IL	Prediction without IL	Prediction with IL	Prediction without IL	Prediction with IL
MSE	0.52	0.18	0.07	0.03	0.06	0.02	0.04	0.015
MAE	0.63	0.33	0.21	0.09	0.16	0.08	0.12	0.06
RMSE	0.72	0.46	0.26	0.14	0.23	0.14	0.16	0.09



(a) (b)

Figure 5. Comparison of Error Rate Results of ABCM

Table 4. Error Rate Results of ABCM with IL

Comparative Models	Error Metrics					
	MSE	MSE	MAE	MAE	RMSE	RMSE
	Prediction without IL	Prediction with IL	Prediction without IL	Prediction with IL	Prediction without IL	Prediction with IL
VARMAX	0.52	0.18	0.63	0.33	0.72	0.46
GRUNN	0.07	0.025	0.21	0.093	0.26	0.14
RBFNN	0.06	0.02	0.16	0.08	0.23	0.14
ABCM	0.04	0.015	0.12	0.06	0.16	0.09

Figure 5 demonstrates the comparative error rate results of ABCM obtained for prediction without IL and IL scenarios. The ABCM accomplishes 0.04, 0.12, and 0.16 of MSE, MAE, and RMSE error rates, respectively, for prediction without the IL scenario. The ABCM minimizes the error rates compared with single predictors like VARMAX, GRUNN, and RBFNN, as shown in Figures 5 (a) and (b). The main reason is that the ABCM fuses the prediction results of three sub-predictors based on the conditional probability and the Bayes rule. However, the error rates of ABCM prediction without IL are higher than those of ABCM prediction with IL. The reason is that the prediction model improves the efficiency nearly equal to real-time by updating the current data with the assistance of IL. Moreover, the ABCM accomplishes 0.015, 0.06, and 0.09 error rates of MSE, MAE, and RMSE, respectively, for prediction with the IL scenario.

Figure 6 illustrates the MSE, MAE, and RMSE comparison results of simple RNN, LSTM, and ABCM models, tabulated in Table 5. The results of figure 5 depict that the ABCM model has the best MAE performance among all models, as the ABCM uses three different sub-predictors such as VARMAX, GRUNN, and RBFNN and provides an optimal result based on the sub-predictor results. For instance, the ABCM accomplish 0.04, 0.11, and 0.2 of MSE, MAE and RMSE values in Figure 6(a), (b), and (c). Unlike the benchmark simple RNN and LSTM models, the ABCM fuses the results of three

different single models and diminishes the prediction error with the assistance of hybrid information. Also, the HL-CTP provides the most informative features to the ABCM prediction model and escalates the prediction accuracy by inventing a novel traffic and timing dataset, TTD. Thus, the ABCM neglects the erroneous prediction data and enhances the EV path planning decision accuracy. The ABCM prediction results fine-tuning with real-time data through incremental learning also maximizes the accuracy level of the predictors. For example, the MSE, MAE, and RMSE prediction performance of ABCM is reduced by 59.2%, 57.5%, and 35.5% when compared with the simple RNN model but is varied by 55.6%, 54.2%, and 33.3% for the LSTM model. Finally, the comparative results of MSE, MAE, and RMSE are shown in Figure 6(d).

Figure 7 represents the optimal solution values of the ISFL algorithm during the EV route selection. The HL-CTP exploits the random integer coding method to model the ISFL for solving the best route selection issue. Further, the congestion and timing information with uncertain real-time information is used to search the optimized EV route and obtain a better path with less congestion and travel time. Figure 6 shows that the ISFL shows good performance in route selection optimization and selects the best-optimized route based on the memes of ISFL. Moreover, the actual solution and ISFL optimal solutions are represented in Figure 7.

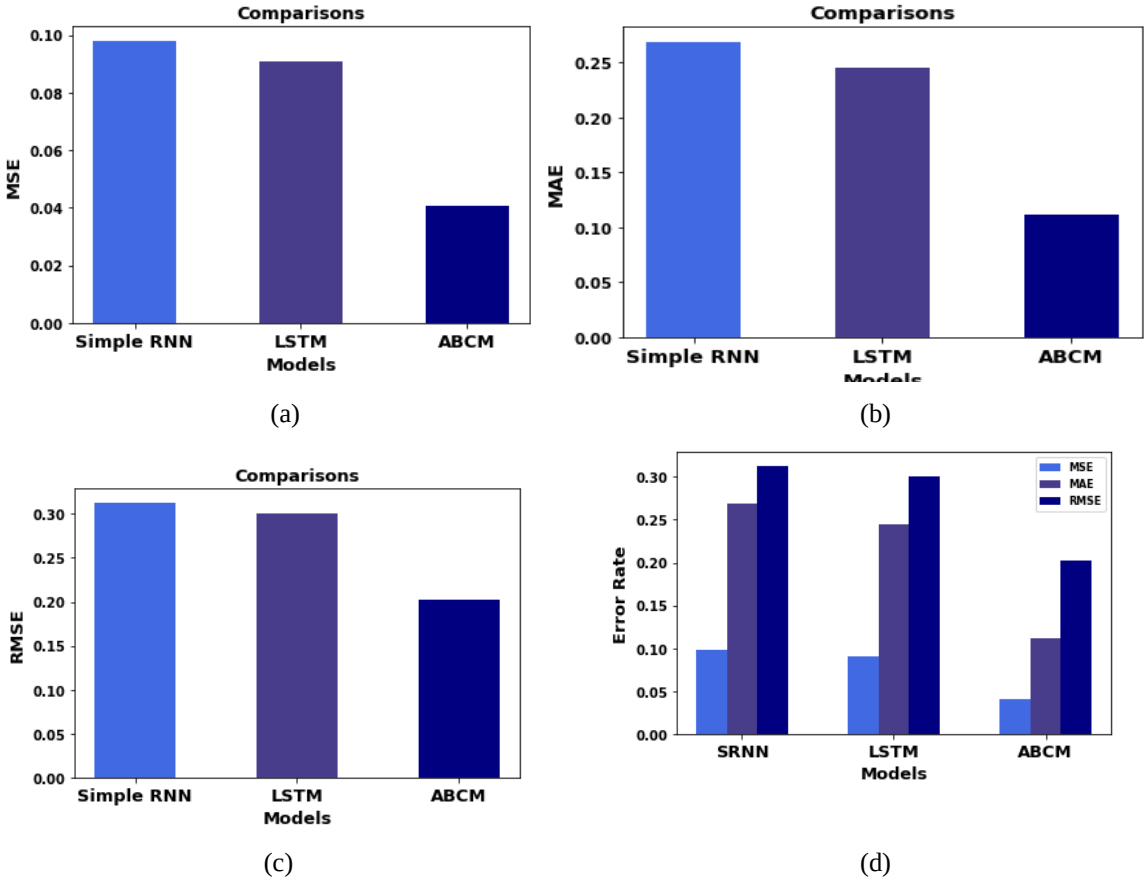


Figure 6. Illustrates the MSE, MAE, and RMSE comparison results of simple RNN, LSTM, and ABCM models

Table 5. Error Rate Comparison Results of ABCM with Benchmark Models

Error Rate	Comparative Models		
	Simple RNN	LSTM	ABCM
MSE	0.0980476	0.0907399	0.0407005
MAE	0.2668887	0.2447706	0.1117087
RMSE	0.3131255	0.3012306	0.2017436

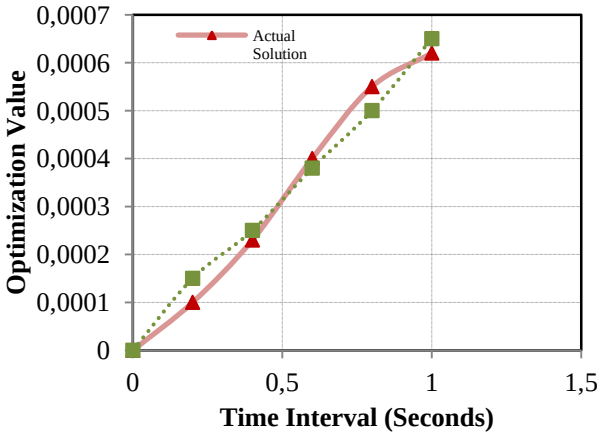


Figure 7. Optimal Solution of ISFL during EV Route Selection

6. Conclusion

Hybrid and incremental Learning enabled Traffic congestion, and a timing-aware real-time traveling route selection strategy named HL-CTP for vehicles has been proposed in this paper. The main intention of HL-CTP is to identify minimum congested shortest traveling routes for vehicles and increases the driving comfort level of VANET users. The HL-CTP incorporates the most informative features in the prediction model and maximizes the accuracy by developing traffic and timing dataset TTD. By taking the most informative features as input with various sub predictors, the ABCM of HL-CTP neglects the erroneous prediction forecasting information and improves the accuracy level of the path planning decision. Updating the current data incrementally with pre-learning enhances the real-time performance of the route prediction algorithm. Further, the final traveling route is selected based on the ISFL algorithm, and the vehicles arrive on time without traveling delay at the desired destination. By employing the advantages of both hybrid and incremental learning, the proposed HL-CTP minimizes the update model cost and accomplishes high real-time route prediction performance. Furthermore, the simulation results show the efficiency of HL-CTP with multiple error rates performance metrics such as MSE, MAE, and RMSE.

Acknowledgements

The authors would like to thank Dr P Shyamaraju, Chancellor, REVA University for all the facilities provided to carry out the research activities. Also, school of ECE, REVA University for their continuous support and encouragement.

References

1. Brockmann, D., Hufnagel, L., & Geisel, T. (2006) The scaling laws of human travel. *Nature*, 439(7075), 462-465.
2. Afrin, T., & Yodo, N, (2020) A survey of road traffic congestion measures towards a sustainable and resilient transportation system, *Sustainability*, 12(11).
3. Xu, T., Li, X., & Claramunt, C, (2017) Trip-oriented travel time prediction (TOTTP) with historical vehicle trajectories, *Frontiers of Earth Science*, 12(2), 253–263.
4. C Lana, I., Del Ser, J., Velez, M., & Vlahogianni, E. I, (2018) "Road Traffic Forecasting: Recent Advances and New Challenges", IEEE Intelligent Transportation Systems Magazine, Vol. 10, No. 2, pp. 93–109
5. Afrin, Tanzina, and Nita Yodo. (2020) A survey of road traffic congestion measures towards a sustainable and resilient transportation system. *Sustainability*, 4660.
6. Wagner-Muns, I., Guardiola, I., Samaranayke, V. and Kayani, W. (2017) A functional data analysis approach to traffic volume forecasting. *IEEE Transactions on Intelligent Transportation Systems*, 19(3), 878–888.

7. Kamble, Shridevi Jeevan, and Manjunath R. Kounte (2021) SG-TSE: Segment-based Geographic Routing and Traffic Light Scheduling for EV Preemption based Negative Impact Reduction on Normal Traffic. *International Journal of Advanced Computer Science and Applications*, 12.12.
8. Liu, Y. and Wu, H. (2017) Prediction of Road Traffic Congestion Based on Random Forest. *10th International Symposium on Computational Intelligence and Design (ISCID)*, 361-364.
9. Tedjopurnomo, D. A., Bao, Z., Zheng, B., Choudhury, F., & Qin, A. K, (2020) A Survey on Modern Deep Neural Network for Traffic Prediction: Trends, Methods and Challenges. *IEEE Transactions on Knowledge and Data Engineering.*
10. Yin, X., Wu, G., Wei, J., Shen, Y., Qi, H., & Yin, B. (2020) A comprehensive survey on traffic prediction. *arXiv preprint arXiv*:2004.08555.
11. Nagy, A. M. and Simon, V. (2018) Survey on traffic prediction in smart cities. *Pervasive and Mobile Computing*, 50, 148–163.
12. Boukerche, A., & Wang, J. (2020) Machine Learning-based traffic prediction models for Intelligent Transportation Systems. *Computer Networks*, 181, 107530.
13. Li, Y., & Shahabi, C. (2018) A brief overview of machine learning methods for short-term traffic forecasting and future directions. *Sigspatial Special*, 10(1), 3-9.
14. Elfar, A., Talebpour, A., & Mahmassani, H. S. (2018) Machine learning approach to short-term traffic congestion prediction in a connected environment. *Transportation Research Record*, 2672(45), 185-195.
15. Hosseinzadeh, A., & Safabakhsh, R. (2014) Learning vehicle motion patterns based on environment model and vehicle trajectories. *Iranian Conference on Intelligent Systems (ICIS)*, 1-5. IEEE.
16. Lanka, S., & Jena, S. K. (2014) Analysis of GPS based vehicle trajectory data for road traffic congestion learning. *Advanced Computing, Networking and Informatics*, 2, 11-18. Springer, Cham.
17. Chen, Z., Shen, H. T., & Zhou, X. (2011) Discovering popular routes from trajectories. In: *2011 IEEE 27th International Conference on Data Engineering*, 900-911. IEEE.
18. Xu, L., Yue, Y., & Li, Q. (2013) Identifying urban traffic congestion pattern from historical floating car data. *Procedia-Social and Behavioral Sciences*, 96, 2084-2095.
19. Zhang, Y., & Zhang, Y. (2016) A comparative study of three multivariate short-term freeway traffic flow forecasting methods with missing data. *Journal of Intelligent Transportation Systems*, 20(3), 205-218.
20. Tang, J., Liu, F., Zou, Y., Zhang, W., & Wang, Y. (2017) An improved fuzzy neural network for traffic speed prediction considering periodic characteristic. *IEEE Transactions on Intelligent Transportation Systems*, 18(9), 2340-2350.
21. Ran, B., Song, L., Zhang, J., Cheng, Y., & Tan, H. (2016) Using tensor completion method to achieving better coverage of traffic state estimation from sparse floating car data. *PloS one*, 11(7), e0157420.
22. Zhang, K., Wu, L., Zhu, Z., & Deng, J. (2020) A multitask learning model for traffic flow and speed forecasting. *IEEE Access*, 8, 80707-80715.
23. Tian, C., & Chan, W. K. (2021). Spatial-temporal attention wavenet: A deep learning framework for traffic prediction considering spatial-temporal dependencies. *IET Intelligent Transport Systems*, 15(4), 549-561.
24. Akhtar, M., & Moridpour, S, (2021) A review of traffic congestion prediction using artificial intelligence. *Journal of Advanced Transportation.*
25. Kong, X., Xu, Z., Shen, G., Wang, J., Yang, Q. and B. Zhang. (2016) Urban traffic congestion estimation and prediction based on floating car trajectory data. *Future Generation Computer Systems*, 61, 97–107.
26. Zhao, J., Xu, F., Liu, W., Bai, J., & Luo, X. (2015) Travel time prediction based on pattern matching method. *International Journal on Smart Sensing and Intelligent Systems*, 8(1).
27. Servos, N., Liu, X., Teucke, M., & Freitag, M. (2019) Travel time prediction in a multimodal freight transport relation using machine learning algorithms. *Logistics*, 4(1), 1.
28. He, P., Jiang, G., Lam, S. K., & Sun, Y. (2020) Learning heterogeneous traffic patterns for travel time prediction of bus journeys. *Information Sciences*, 512, 1394-1406.
29. Petersen, N. C., Rodrigues, F., & Pereira, F. C. (2019) Multi-output bus travel time prediction with convolutional LSTM neural network. *Expert Systems with Applications*, 120, 426-435.
30. Wu, J., Wu, Q., Shen, J., & Cai, C. (2020) Towards attention-based convolutional long short-term memory for travel time prediction of bus journeys. *Sensors*, 20(12), 3354.
31. Zhang, W., Yu, Y., Qi, Y., Shu, F. and Y. Wang. (2019) Short-term traffic flow prediction based on spatio-temporal analysis and CNN deep learning. *Transport metrica A: Transport Science*, 15(2), 1688–1711.

32. Wen, F., Zhang, G., Sun, L., Wang, X. and X. Xu. (2019) A hybrid temporal association rules mining method for traffic congestion prediction. *Computers & Industrial Engineering*, 130, 779–787.
33. He, Z., Qi, G., Lu, L. and Y. Chen. (2019) Network-wide identification of turn-level intersection congestion using only low-frequency probe vehicle data. *Transportation Research Part C: Emerging Technologies*, 108, 320–339.
34. Shridevi, Jeevan Kamble, Manjunath R. Kounte. (2022) A Survey on Emergency Vehicle Preemption Methods Based on Routing and Scheduling. *International Journal of Computer Networks and Applications (IJCNA)*, 9(1), 60-71. DOI:10.22247/ijcna/2022/211623.
35. Zhao, H., Jizhe, X., Fan, L., Zhen, L. and L. Qingquan. (2019) A peak traffic Congestion prediction method based on bus driving time. *Entropy*, 21(7), 709.
36. Chen, Y. Z., Shen, S. F., Chen, T., & Yang, R. (2014) Path optimization study for vehicles evacuation based on Dijkstra algorithm. *Procedia Engineering*, 71, 159-165.
37. Chen, Z., Jiang, Y., Sun, D. and X. Liu. (2020) Discrimination and prediction of traffic congestion states of urban road network based on spatio-temporal correlation. *IEEE Access*, 8, 3330–3342.
38. Chen, C., Wang, H., Yuan, F., Jia, H., & Yao, B. (2020) Bus travel time prediction based on deep belief network with back-propagation. *Neural Computing and Applications*, 32(14), 10435-10449.
39. Qi, G., Ceder, A. A., Zhang, Z., Guan, W., & Liu, D. (2021) New method for predicting long-term travel time of commercial vehicles to improve policy-making processes. *Transportation Research Part A: Policy and Practice*, 145, 132-152.
40. Sun, J., & Kim, J. (2021) Joint prediction of next location and travel time from urban vehicle trajectories using long short-term memory neural networks. *Transportation Research Part C: Emerging Technologies*, 128, 103114.
41. Adewale, A. E., & Hadachi, A. (2020) Neural Networks Model for Travel Time Prediction Based on ODTravel Time Matrix. *arXiv preprint arXiv*:2004.04030.
42. Bogaerts, T., Masegosa, A. D., Angarita-Zapata, J. S., Onieva, E. and P. Hellinckx. (2020) A graph CNN-LSTM neural network for short and long-term traffic forecasting based on trajectory data. *Transportation Research Part C: Emerging Technologies*, 112, 62–77.
43. Hou, Y., Chen, J., & Wen, S. (2021) The effect of the dataset on evaluating urban traffic prediction. *Alexandria Engineering Journal*, 60(1), 597-613.
44. Kamble, Shridevi Jeevan, and Manjunath R. Kounte. (2019) Routing and scheduling issues in vehicular ad-hoc networks. *International Journal of Recent Technology and Engineering*, 8.3: 4272-4283.
45. Joy, Helen K., Kounte, Manjunath R. (2022) Deep CNN Based Video Compression with Lung Ultrasound Sample. *Journal of Applied Science and Engineering*, 26(3), 313-321.
46. Naveen, Soumyalatha, Kounte, Manjunath R., Ahmed, Mohammed Riyaz. (2021) Low Latency Deep Learning Inference Model for Distributed Intelligent IoT Edge Clusters. *IEEE Access*, 9, 160607-160621. DOI: 10.1109/ACCESS.2021.3131396.