International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(2) 2026 pp. 341-353
https://doi.org/10.70917/ijcisim-2026-2641
Article

Forecasting VANET Traffic Congestion Employing Extreme

Jenolin Rex M.[1]*, K. Sree Kumar[2]

[1] Department of Computing Technologies, SRM Institute of Science and Technology, Kattankulathur, Chengalpattu, Tamil Nadu, India.
Email: jenolinrex@gmail.com
[2] Department of Computing Technologies, SRM Institute of Science and Technology, Kattankulathur, Chengalpattu, Tamil Nadu, India.
Email: sreekumk@srmist.edu.in

Abstract :A vehicular Ad-hoc network (VANETS) is considered as the new age wireless communication that establishes the communication among the vehicles and exchanging data with one another about things like vehicle speed, location & direction. These networks consist of the vehicular onboard unit(OBU) as well as road side unit(RSU) which are used for the exchange of the messages between the vehicles. Though the VANETS is considered to be an intelligent data exchange between the vehicles, still it suffers from the non-prediction of traffic congestion which still creates the bottleneck for the creation of an efficient vehicular networks. Deep learning and machine learning algorithms have recently shone a stronger light on studies to improve the network's capacity to predict traffic congestion on the roadways. But these algorithms needs improvisation to be predict the traffic congestion in accordance to the dynamic variation of the vehicles speed and position. Consequently,Extreme Deep Reinforcement Learning techniques(EDRL) is proposed for an effective prediction of traffic congestion under the different scenario of trafficswere used in this work. The traffic scenario was created using the Simulation of Urban Mobility(SUMO) in which datasets for the real-time tests were gatheredand integrated with the proposed model to predict the traffic congestion. Metrics like the average travelling time delay (ATTD) as well as the average waiting time delay (AWTD) are used to assess the performance of the suggested approach. The suggested model's performance is compared to that of various deep learning models already in use, including Convolutional Neural Networks (CNN), Long Short Term Memory (LSTM), and Recurrent Neural Networks (RNN), to demonstrate its superiority .Results demonstrates that the proposed model has shown improved performance over the multiple iterations, since the EDRL is adaptive to the changes in the traffic environment and inclusion of extreme feed forward networks has shown the faster response than the existing deep learning models.
Keywords : Vehicular Adhoc Networks, Extreme Deep Reinforcement learning, SUMO, CNN, LSTM, RNN

1. Introduction

Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communications are intelligently sent over Vehicular Adhoc Networks (VANETS), a component of Intelligent Transportation Systems (ITS) . There is a variety of further possible advantages, including improved transportation safety, decreased traffic congestion, analysis of road capacity, and even a reduction in parking space needs. Due to the exponential rise in the number of automobiles on the road, traffic congestion is unavoidable and results in a significant waste of time and resources.

Traffic flow prediction is a vital element of ITS and helps the system predict and avoid traffic jams, efficiently control and manage traffic, and determine the optimal route. Artificial Neural Network (ANN) schemes, Markov process-based techniques and the k-nearest neighbor (kNN) algorithm are all examples of machine learning (ML)-based approaches [1-5]. Deep learning (DL) methods provide advantages over traditional machine learning (ML) techniques, such as reducing the data preprocessing process and surpassing other ML techniques in terms of accuracy. As a result, DL schemes have recently attracted a lot of attention in the prediction of traffic flow [6-11].To discover the crowded way in line with continuous change in the traffic characteristics, these DL systems are constrained and lack flexibility. On the other side, there are also situations where traffic jams develop after a vehicle already has entered that path and there are no other highways open at that point. It is believed that using a standard VANET alone may not be sufficient in this case. As a result, it is anticipated that deep learning algorithms would undergo a revolution, becoming more adaptable, flexible, and trustworthy.

Copyright: © 2026 by the authors

Motivated by the above problem, these research articles propose the system in which the deep learning architecture is developed by merging with extreme feed forward neural networks. This method has an agent that can figure out the best course of action in any changes to the driving environment. By coordinating quicker actions before learning about alterations in the traffic situation, it resolves the aforementioned difficult dilemma. **The following is the planned research's primary contribution:**

1. Propose the novel integration of Deep reinforcement mechanism with the Extreme Feed Forward layers to achieve the effective prediction of traffic in accordance to the adaptive changes in the traffic scenarios.
2. Exploits the usage of the Extreme Learning Machinesbased Feed Forward Networks in the prediction of traffic networks for the faster and high accurate prediction.
3. Evaluates the effectiveness of the developed algorithm using the different metrics and compared with the state-of-art deep learning models.

The remainder of the essay is structured as follows: The works by multiple writers that are linked are elaborated in Section 2. The proposed framework's traffic data collection and operation are described in Section 3 of the document. Section 4 presents the experiments, findings, and comparative analyses. The study is ended in Section 5 with a discussion on the future direction.

2. Associated Works :

Perez-MuruetaProposed an algorithm called “Entropy Balanced K Shortest Path Algorithm" to redirect the vehicles to avoid congestion. This algorithm is purely based upon Deep learning (DL) to forecast the traffic of network. Based on the previous data about congestion, this algorithm redirects the vehicle possibly and predict the congestion for nearest zones. This algorithm collects the information from a live environment by using set off probe cars so as to find noncurrent congestion. The outcome infers that, this algorithm significantly diminishes the “Average Travel Time" by 19 % as well as profiting 38 % of vehicles. However main challenge of this approach is vehicle rerouting and forecasting the condition without affecting new condition in other places. [12].

Singh, H discussed the strategies available in system for traffic condition in VANET environment. A detailed summary has given for current strategies. Based on the parameters such as latency, computation overhead throughput, and comparative analysis done on the existing strategies. From the result analysis, it is come to know that, dynamic condition control strategy outperforms all the other existing strategy in terms of above mentioned parameters. Anyhow, this work require extension to integrate few more methodologies for queue maintenance during the occurrence of congestion in dynamics strategy in order to produce better performance in the above mentioned parameters [13].

Wei Kuang Laiintroduced a network called ML- Assisted Selection "MARS". This network significantly utilizes the roadside unit for Road information maintenance. This system is able to predict the vehicle movement after that appropriate routing way is used for data transmission with the highest transmission capacity. This system perfectly provide best routing direction among road side units by using real time traffic data. The key benefit of this method is that it can deliver better results for various vehicle densities.The system is highly reliable and shows low impact when varying V2V capacity. The primary challenge of this approaches it actually has less control overhead [14].

Ahmed introduced a novel “real time route suggestion” protocol which utilizes distinctive travel times under various street conditions. For equipped and non-equipped vehicles also the congestion index precision is expanded. The travel time forecast model incorporates the driving interruption factors which makes the proposed protocol as a unique approach. For the result analysis, real time urban network scenarios has utilized to experience the better route travelling .This protocol provide best performance during the congestion hours also. The principle benefit of this work is, it has the ability to provide reduced E2E delay which almost same for no congested hours. The limitation of this work is, it provide some fluctuations during the roads are presented to the traffic congestion [15].

M. de Souze idea is to reduce the average vehicle travel cost and to improve spatial usage of intelligent road network. The authors proposed and algorithm called (CIDMERA) (“Congestion Avoidance through a Classification Mechanism and Re- Routing Algorithm”). This algorithm outperforms all the other existing algorithm with regards to average travel time travel distance, average stopped time by 31%, 8% and 70% respectively [16].

Pan et al. Introduced a centralized framework which incorporates 2 algorithms namely DSP and RKSP (Dynamic shortest path and random K shortest path) for the traffic jam detection which relies on vehicle speed, position, direction. The DSP algorithm selects the routing path with lowest travel time, RkSP algorithm selects the routing path randomly. Balancing rerouted traffic among various paths can significantly avoid switching traffic congestion from one area to other area. The primary challenge of this framework is to implement this in to real time scenario for congestion detection, because it only detecting congestion in succeeding rerouting phase [17].

Meneguette, R.I Introduced protocol known as INCIDEnT "INtelligent protocol of Congestion DETection" which is purely relies on ANN to detect and classify congestion levels. This protocol significantly manages fuel consumption, Co emissions as well as try time. Hence, it produce huge success rate during congestion. This protocol utilized real time scenarios by using OpenStreetMap. But this work does not incorporate any rerouting methodology during highly congested traffic environment [7].A. M. de Souzautilized V2V communication to introduce a methodology called " SCORTON – A Solution using Cooperative Rerouting to prevent congestion and improve Traffic cONdition" In this methodology, RSU collects all the vehicle data such as vehicle id, position, destination, route by utilizing 4G and LTE technologies. KNN is utilized for purpose of congestion classification. After the determination of traffic conditions, this methodology reroutes the vehicles via non congested area. This methodology reduces the ATT significantly by 17%. However, this methodology has few limitations such as, restricted rerouting distance, No system for re-routing mechanism for worldwide point of view [18].

Ma utilized CNN methodology for the detections and prediction of congestion. With the help of a 2D time-space matrix, this technique depicts the time and space relationships between traffic streams as pictures. CNN extracts the features from traffic and predicts the traffic speed in the network. The proposed methodology utilizes real world traffic environment in Beijing and the results were compared with existing algorithms such as KNN, ANN, ordinary least squares and RD (Random Forest). The CNN outperforms above mentioned methodologies in terms of average accuracy by 27-96% inside ATT. This model is well suited for large scale network because of reasonable training time. However, these framework simply over viewed the temporal corrections of traffic development in single area, and did not incorporates the spatial corrections of the network traffic [19].

M. Bachl utilized Reinforment learning algorithm to propose "Reactive Adaptive Experience based congestion control" mechanism which keeps optimum congestion windows by concerning current network conditions. This technique can be started with random weights to speed up convergence and increase network stability. The congestion control directly relies on apperance of affirmations that are delayed and generally received individually, the issue is not reasonable to current executions of DL. In order to overcome this issue, partial action learning in formulated. The results infers that, this methodology outperforms other works by increasing network stability and provide optimal solution within minutes. Regardless of showing excellent outcomes in terms of prediction. Main constrain of this works in excludes redirection systems in case of congestion [20].

A congestion control identification approach based on DL that simply makes use of packet arrival time was proposed by M. Bachl. For the congestion control identification this mechanism does not require knowledge about domain (or) header information during training the variants. When compared to traditional approaches, variants prominently fluctuate in behaviours like BBR. However, this methodology is tested in distinctive misfortune based congestion control when it can't discover sufficient feature characteristics. The exhibition can be mostly expanded when specific network properties are getting trained and optimised. Fundamental contain of this work is that, it necessarily need all vehicles information to understand the real traffic state [21].

Nathan joy examined internet congestion control by utilizing RL - Reinforcement Learning algorithm. This RL empowers training of deep network strategies which capture intricate patterns in traffic data and network conditions. This work additionally feature huge difficulties confronting genuine reception of RL-based blockage control, including decency, wellbeing, and speculation, which are not inconsequential to address inside traditional RL formalism. This work also presented a test suite for RL-guided blockage control dependent on the OpenAI Gym interface. Fundamental compel of this model is that it works in a single road section, which prohibits judging its competence on a global scale (the whole vehicular organization). [22].

Section-3

3.1Propoed Methodology :

To achieve an improved prediction of traffic congestion, the suggested reinforcement learning framework's working mechanism is divided into three parts: a data collection unit, feature extraction, and reinforcement learning.

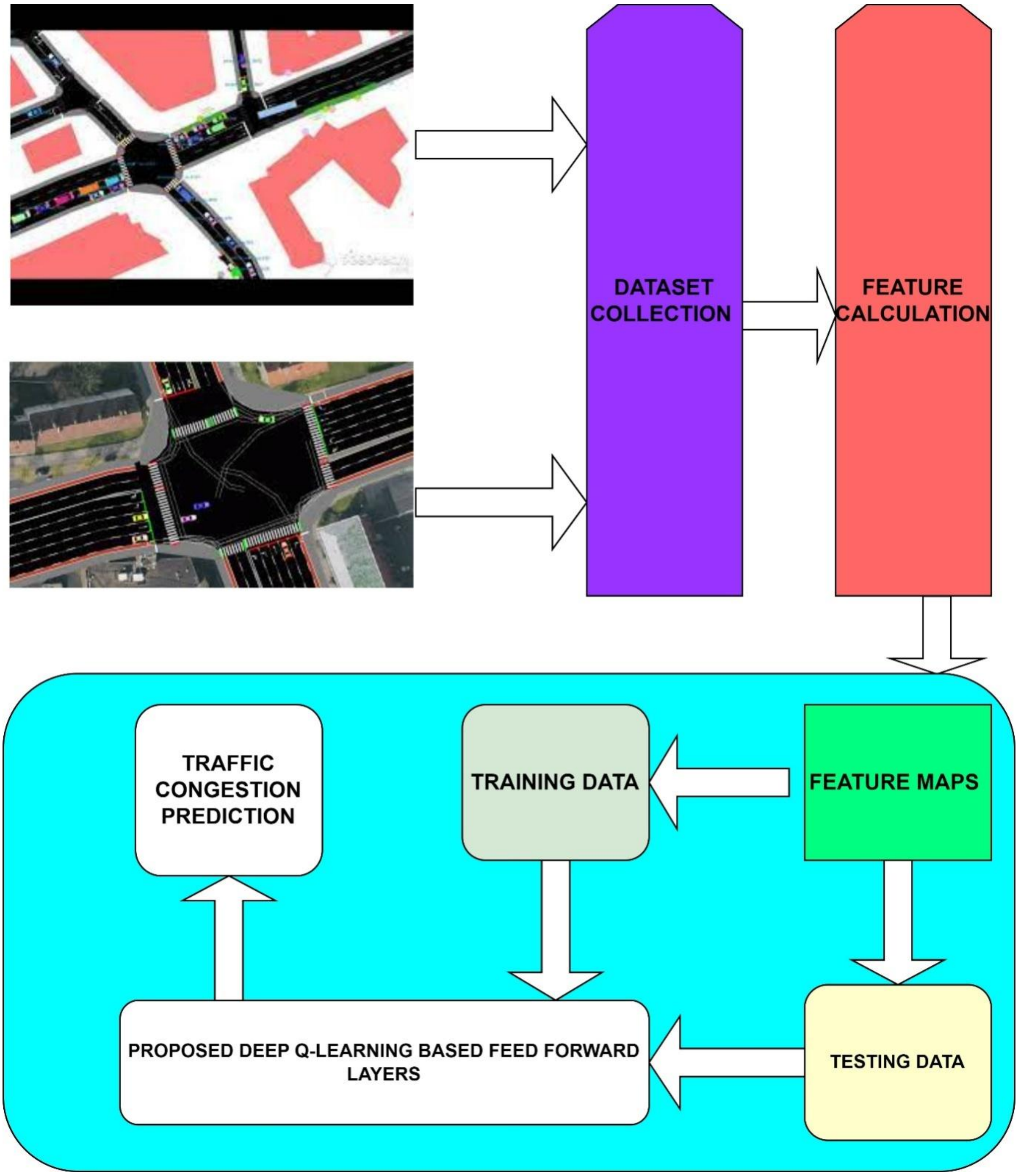

Figure 1 Overall structure of the suggested Framework used for the Traffic Congestion Prediction

3.2 Data Collection Unit :

Real-time data sets from the road scenarios on the Chennai to Salem Highways were gathered in order to construct the envisioned network in an efficient manner. SUMO is combined with the OMNET++ platform to implement the real-time data gathering mechanism. SUMO is used to build the traffic scenario, and OMNET++ is employed to extract the characteristics from the traffic scenario. The methodology is applicable to all road systems across the nation. and about ten months' worth of twelve hours (12 Hours) of traffic data, from 9 AM to 9 PM, were gathered and used for studies.Figure 2 is a road scenario developed with SUMO in real time. Additionally, all of the cars tracked by the SUMO are transformed into vehicular nodes via plug-ins made with C++ programming allowing simple implementation in OMNET++ for additional analyses and modeling.

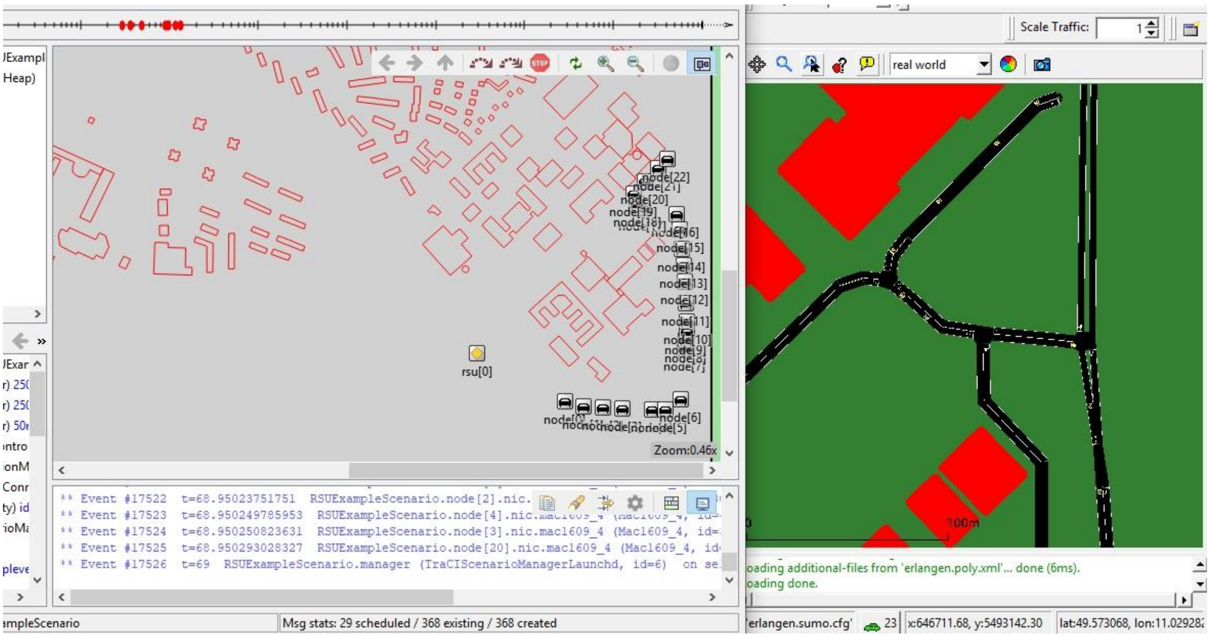

Figure 2 Real-time Data Collection Module Scenario with SUMO-OMNET++ Interfaces

3.3 Feature Extraction :

Our recently developed plug-ins, which were covered in the previous section, were able to extract the various features, including the quantity of TCP packets transmitted, the quantity of packets received, signal strength, latency, vehicle speed, direction, throughput, time stamp, input message ID, and layer ID. Additionally, labeling is carried out based on the quantity of data collected for each as well as every moment so that it may be fed into the prediction model we've suggested. The features that were retrieved are an overall reflection of the movement of traffic on the road network.The features such as throughput, number of vehicles, speed of the vehicles, latency between the information , no of packets transmitted and delivered from the source to destination.To train the model, almost 9745 datasets were gathered and utilised.

3.4 Proposed Framework:

To create the traffic congestion model, the suggested system uses deep reinforcement Q-learning and Feed Forward Extreme learning machines.

3.4.1 Deep Reinforcement Q-Learning Model :

The Q-learning process is primarily used to construct the routing protocol. The Q-learning technique is chosen due to the several drawbacks of the current routing strategy. The Q-learning paradigm facilitates the acquisition of features pertinent to dynamic situations. As a result, Markov Decision Process is being used to generate Q learning like a reinforcement learning approach . This MDP specifies (S,A,P,R) and $P^{a}_{zz'}$ as the parameters for states action, reward, and probability of occurrence. Assume that the current state is z and the next state, z', has an action value of ‘a’.

$$P^{a}_{zz'} = Prob\{z_{i+1} = z' | z_t = z, a_t = a\} \qquad (1)$$

The reward function is provided as $R^{a}_{z_t z_{t+1}}.t$ for states z and z'. Current state's overall reward function is

$$R_t = \sum_{z_{t+1} \epsilon Z} P^{a_t}_{s_t s_{t+1}} R^{a_t}_{z_t z_{t+1}} |_{z_t=z, a_t=a}$$
(2)

Let the utility function with a policy variableω,$Q_{\omega}(z,a)$ be defined.

$$Q_{\omega}(z,a) = \{R_t + \gamma \sum_{z_{t+1} \epsilon Z} P^{a_t}_{z z_{t+1}} Q_{\omega}(z_t, a)\} \qquad (3)$$

γ →e reduction factor [0,1] & practically γ values primarily be [0.5,0.99].

$$Q^{*}(z,a) = max\{Q_{\omega}(z,a)\}$$

$$= \left\{ R_t + \gamma \sum_{z_{t+1} \epsilon Z} P^{a_t}_{z_t z_{t+1}} max\{Q_\omega(z_{t+1}, a)\} \right\} |z_t = z, a_t = a$$

$$=\{R_t + \gamma \sum_{z_{t+1} \epsilon Z} P^{a_t}_{z_t z_{t+1}} Q^*(z_{t+1}, a)\}|z_t = z, a_t = a \quad (4)$$

3.4.2Feed Forward Extreme Learning Networks :

The planned BORN currently employs Extreme learning machines created by G.B. Huang[22], which include a hidden units, speed, velocity, and accuracy with considerable speculation/exactness, as well as the ability to approximate any function [22].

This technique states that even though the output layer is assumed to be in line, the "L" neurons of the convolution layer are linked with a consistent activation function (for example, the sigmoid function). In ELMs, it is not essential to adjust the concealed layer separately. The hidden layer's weights are fixed at random (counting the bias loads).

The models are equated before training data are taken into account. The system yield for a mono layer ELM is provided by eqn (5)

$f_l(n) = \sum_{i=1}^{l} \omega_i g_i(n) = g(n)\omega$ (5)

Where n→ input

Output weight vector's expression is ω

$\omega = [\omega_1, \omega_2, \dots \dots \dots \dots . \omega_l]^T$(6)

G(n)→Concealed layer outputas

$g(n) = [g_1(n), g_2(n), \dots \dots \dots \dots . . g_l(n)$ (7)

For estimating the target vector O, the hidden layers were established by eqn (8)

$$G = \begin{bmatrix} g(n_1) \\ g(n_2) \\ \vdots \\ g(n_N) \end{bmatrix} \quad (8)$$

The ELM are accomplished using the least squares model's marginal non-linear characteristic and are represented as

$\omega' = G^* O = G^T(GG^T)^{-1}O$ (9)

Where G∗→Moore−Penrose inverse (inverse of H)

Eqn (9) can be rewritten as

$\omega' = G^T(\frac{1}{C}GG^T)^{-1}O$ (10)

The output function is thus estimated as

$f_l(n) = g(n)\omega = g(n)G^T(\frac{1}{C}GG^T)^{-1}O$ (11)

ELM employs the kernel function to produce high accuracy and improved performance. Limited training errors and better approximation are the ELM's main benefits. ELM finds use in prediction and classification values due to the automatic-tuning of the weight biases with non-zero activation functions. We may find a thorough explanation of ELM's equations in [22], [23]. Algorithm 1 depicts the ELM's pseudo code.

In step 1, collect "N" set of data for training with n Hidden neurons and an Activation Function. In step two, biases and input weights are assigned. Calculate the hidden matrix in step three. Next, H concealed matrix has been calculated and finally classify or predict the values in Step 5.

3.4.3 Proposed Hybrid Integrated Model :

As mentioned in Section 3.4.1, proposed model has been developed based on the following parameters such as Environment and agent models. The world in which an agent operates is referred to as the environment.In the environmental model, features that are extracted from the SUMO using OMNET++ are used . After getting the data , multi-dimensional arrays are constructed using the python libraries The agent's responsibility is to decide based upon rewards and observations.In this case, Extreme Feed Forward Networks are built inside the agent which receives the different vehicular matrices as mentioned in **Section 3.4.2.**The agent chooses its course of action based on the incentives it receives.This model's input layer is made up of six nodes with the throughput, vehicular speed, and position, latency, no of data transmitted and no of data received. The Q-value is the output layer. The parameters used to train the suggested model are shown in Table I. The proposed model replaces the traditional ReLu with the parameterized ReLu to prevent the data losses during the model training.

Table I Variables that were used to train the proposed model

Sl.no	Various Network Parameters	Particulars
01	No. of Input Layer	06
02	No. of Output Layer	03
03	No. of Hidden layer	200
04	Bias Weights	Auto-Tuned
05	Activations Functions-	PReLU-Parameterized Relu

Each set consisting of a state and an action generates the reward function R(N). The proposed network's reward function is provided as follows:

$$R(N) = < D(S) \qquad (12)$$

Where D(S) is the latency of the vehicles to move from one source and destination. Then based on the equation () ,The training process for the suggested deep extreme machines iterates and converges to obtain numerous Q-Values for each node. The proposed model's operational mechanism is shown in Algorithm-1.

Steps	**The proposed algorithm's pseudo code, Algorithm-1**
1	Inputs : Traffic Data Collected From the SUMO
2	Outputs : Traffic Congestion Prediction
3	Label the Data and Pre-processed for the Applications
4	Formulate the Q-Based Extreme Feed Forward Networks
5	Formulate the Reward Functions using Equation(12)
6	Based on the reward function, calculate the outputs using Equation(11)
7	If Output ===1
8	//No Traffic Congestion
9	Else if Output==2
10	//High Traffic congestion
11	Else if Output==3
12	//Medium Traffic Congestion
13	Else if Output==4
14	//Low Traffic Congestion
15	Else
16	Go to Step 6
17	Stop
18	Stop
19	Stop
20	Stop

Section -4

4.1 Execution Process:

The suggested model is executed using the real time traffic road scenario of Chennai ,Tamilnadu,India. As stated in Section-3, real time traffic datasets information were collected from SUMO and employed to train the model.

To perform the continuous simulation, a distinct Python API has been established to communicate with the data collection unit running on the OMNET++ platforms. Nevertheless, the simulation study was carried out utilizing the Python Tensor Flow API on an Intel i8 with a 2GB NVIDIA GPU running at 3.0 GHz and 8 GB of RAM.The rerouting mechanism is executed in OMNET++ plugins.

4.2Training Models

The aforementioned proposed model underwent 0.001 learning rate training on real-world traffic data. Table III is a list of the trainable parameters for the suggested training model.

Table 2Training variables employed in the study

Parameters	Standards
No of Hidden layer	30
Dropout	0.2
Batch Size	30
Epochs used	220

4.3 Performance metrics :

Metrics including accuracy, precision, recall, specificity, and F1-score are generated during the performance study and contrasted with those of other current deep learning frameworks. The mathematical formula used to determine how well the suggested model performed in predicting traffic congestion is shown in Table 3**.**

Table 3Mathematical Expression for Performance Metrics Calculation

SL.NO	Performance Metrics	Mathematical Expression
01	Accuracy	$\frac{Tp + Tn}{Tp + Tn + Fp + Fn}$
02	Sensitivity or recall	$\frac{Tp}{Tp+Fn}$ x100
03	Specificity	$\frac{Tn}{Tn + Fp}$
04	Precision	$\frac{Tn}{Tp + Fp}$
05	F1-Score	$\frac{Precison * Recall}{Precision + Recall}$

Tp&Tn are True Positive & Negative

Fp&Fn are False Positive & Negative

Table 4Performance Indicators for the Suggested Framework in Each Time Zone

Time Hrs	Performance metrics				
	Accuracy	**Precision**	**Recall**	**Specificity**	**F1-score**

9 am to 10	0.92	0.91	0.89	0.112	0.92
10 to 11	0.92	0.90	0.88	0.120	0.89
11 to 12	0.94	0.892	0.892	0.110	0.89
12 to 3	0.912	0.88	0.845	0.164	0.85
3 to 5	0.93	0.89	0.885	0.114	0.876
5to 8	0.9435	0.90	0.893	0.112	0.893
8 to 9 pm	0.94	0.92	0.91	0.101	0.90

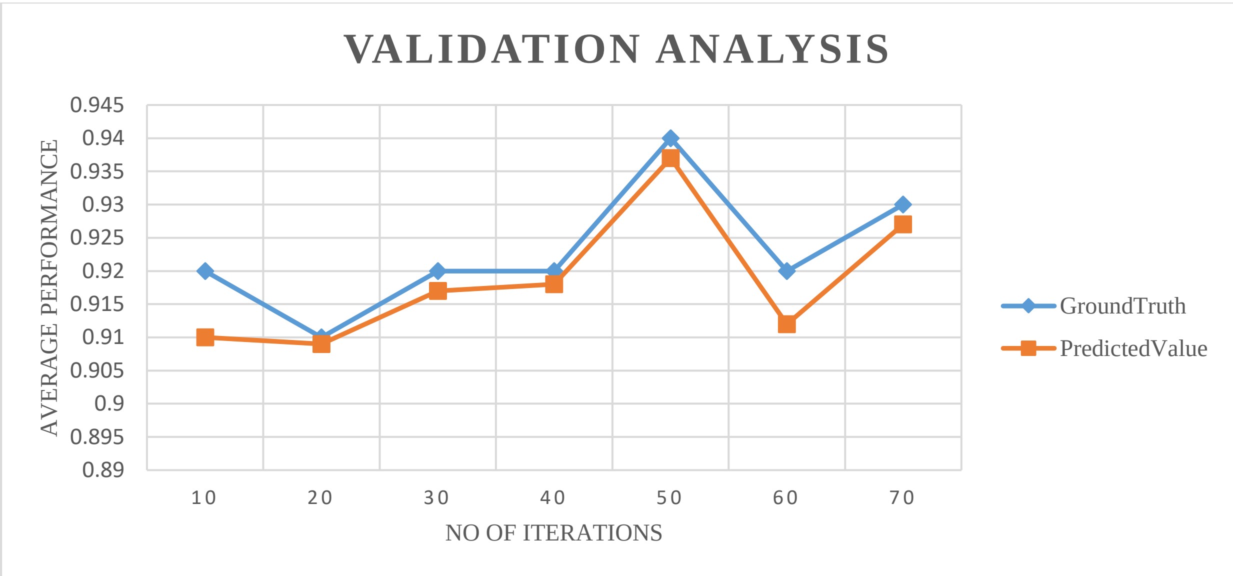

Figure 3Performance Validation of the Proposed Framework for the Various Iterations

Table 5 Performance Indicators for the Current CNN in Each Time Zone.

Time Hrs	**Performance Indicators**				
	Accuracy	**Precision**	**Recall**	**Specificity**	**F1-score**
9am to 10	0.73	0.72	0.70	0.303	0.70
10 to 11	0.72	0.71	0.70	0.302	0.72
11 to 12	0.73	0.715	0.685	0.320	0.70
12 to 3	0.72	0.710	0.674	0.324	0.685
3 to 5	0.723	0.72	0.645	0.367	0.703
5 to 8	0.723	0.70	0.68	0.32	0.702
8 to 9	0.73	0.68	0.684	0.332	0.689

Figure 4 The CNN Framework's Validation Performance for the Various Iterations

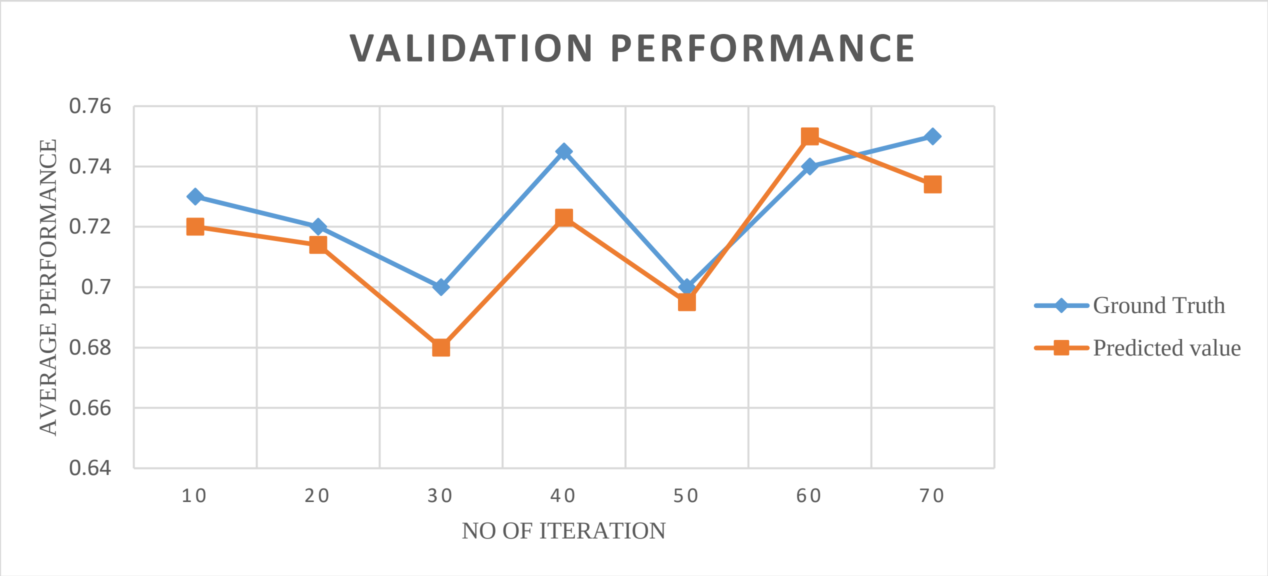

Table 6Performance Measurements of the Current LSTM in the various time zones.

Time Hrs	Performance Indicators				
	Accuracy	Precision	Recall	Specificity	F1-score
9 to 10	0.78	0.763	0.70	0.330	0.732
10 to 11	0.783	0.774	0.70	0.332	0.745
11 to 12	0.783	0.773	0.75	0.332	0.734
12 to 3	0.775	0.73	0.76	0.340	0.745
3 to 5	0.765	0.754	0.74	0.323	0.732
5 to 8	0.774	0.744	0.70	0.304	0.703
8 to 9	0.79	0.75	0.70	0.323	0.743

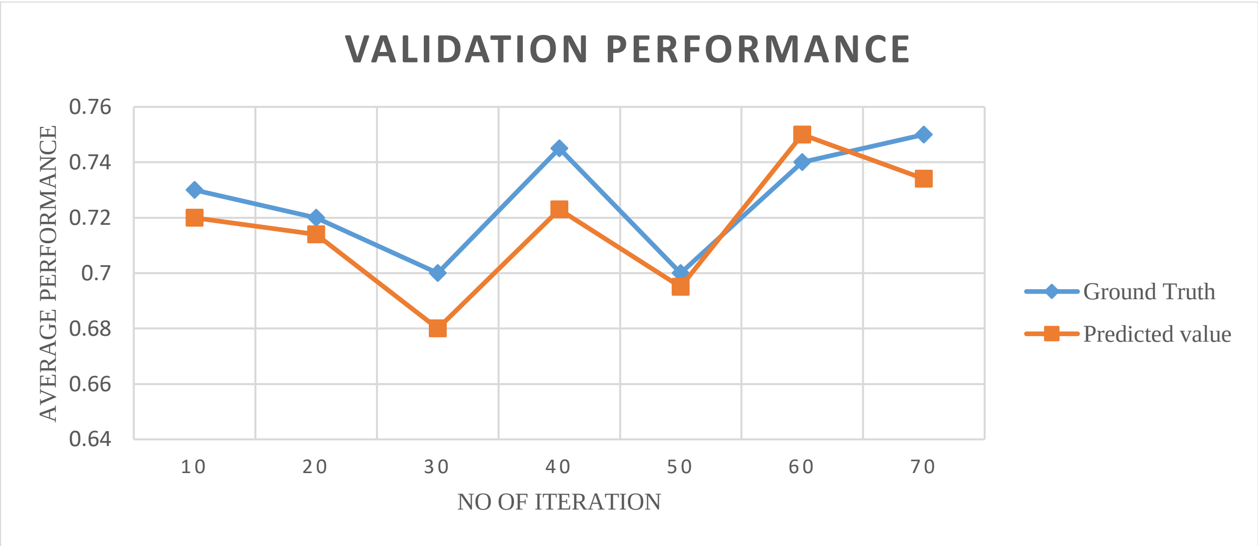

Figure 5 Validation Performance of the Existing LSTM for the various iterations

Table 7 Performance Measurements of the Current RNN in the various Time Zones

Time Hrs	Performance Indicators				
	Accuracy	Precision	Recall	Specificity	F1-score
9 to 10	0.81	0.80	0.79	0.290	0.75
10 to 11	0.82	0.813	0.78	0.220	0.73
11 to 12	0.80	0.785	0.77	0.292	0.743
12 to 3	0.82	0.80	0.79	0.230	0.79
3 to 5	0.82	0.802	0.784	0.204	0.800
5 to 8	0.82	0.814	0.79	0.23	0.80
8 to 9	0.83	0.82	0.81	0.202	0.812

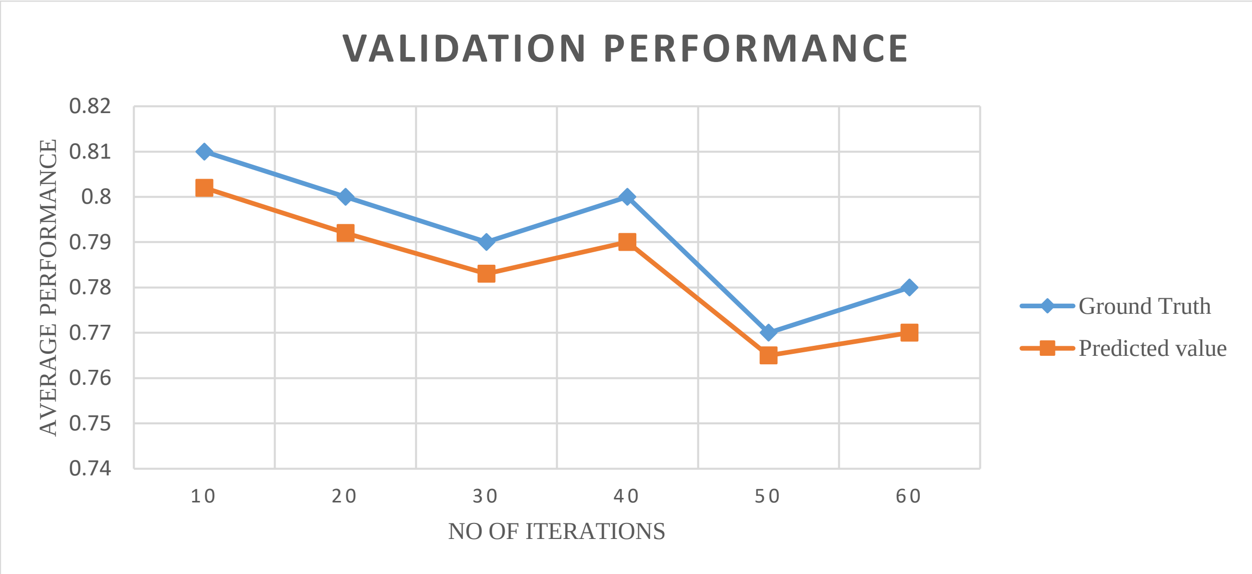

Figure 6 Validation Performance of the Existing RNN for the different iterations

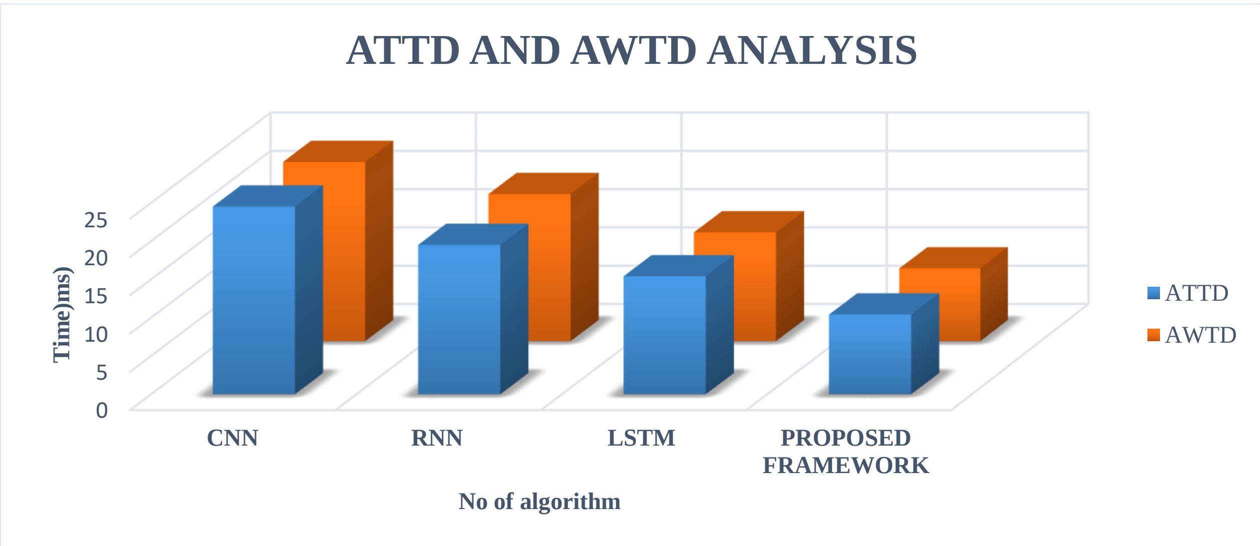

Figure 7 ATTD and AWTD Analysis for the Different Algorithms at VANET Scenario

4.4 Findings :

Table (4) and Figure (3) illustrates the performance and validation of the suggested algorithm at the different time zones. From the Table (4) , it is evident that the developed framework has produced the uniform average efficiency of 0.92 accuracy in detecting the different traffic levels .Also, validation error of the proposed model is even less than 0.01 for different iterations. Table(5) and Figure(4) presents the performance and validation of the q-based Convolutional neural networks(Q-CNN) in which the average performance is found to be 0.74 and the validation error is found even greater than the proposed framework. Table(6) presents the average performance of the Q-Recurrent Neural Network while figure (5) presents the validation performance of the Q-RNN in predicting the traffic congestions . It is found that the average performance is slightly increased than CNN but the error is found to be 0.293 while validating with the ground truth values. Table (7) and figure(6) presents the performance of the Q-LSTM (Long Short Term Memory) in predicting the congestions. From the observations, it is found that the LSTM has produced the better performance than CNN and RNN but lesser performance than the proposed framework. Overall, the proposed framework has produced the best performance which is 20% greater than CNN, 18% than RNN and 16% than LSTM. Moreover, ATTD and AWTD was very low compared with the other deep learning architectures which is also evident from Figure 7.

Section-5

Conclusion and Future Scope :

In this work, deep reinforcement learning have been developed using the Extreme Feedforward networks to forecast the traffic congestion in the road side scenario. The suggested framework has undergone real-time testing, traffic scenario using the SUMO-OMNET++ framework. Nearly 9745 datasets(12 Hrs) are collected at the different time zones of particular road scenarios. Using the obtained dataset, significant experimentation has been conducted, and performance measures including accuracy, precision, recall, specificity, and F1-score are assessed and analyzed. Additionally, ATTD and AWTD are also measured for the analysis of proposed model. The performance of the suggested model is also contrasted with those of other deep learning architectures currently in use. Results show that the suggested model performed better than any other deep learning models, establishing its superiority in the process of forecasting traffic jams.As the future enhancement, proposed architecture still needs improvisation to handle the larger road datasets also considering the varying climatic conditions.

References:

1. “IEEE Standard for Local and metropolitan area networks, Amendment 6: Wireless Access in Vehicular Environments,” IEEE Std 802.11p-2010, pp. 1–51, 2010.
2. C.K.MoorthyandB.G.Ratcliffe,“Shorttermtrafficforecastingusingtimeseriesmethods,”Transportation PlanningandTechnology,vol.12,no.1,pp.45–56,1988.
3. J. Ma, X.-D. Li, and Y. Meng, “Research of urban traffic flow forecasting based on neural network,” Acta Electronica Sinica, vol.37,no.5,pp.1092–1094,2009.
4. B. L. Smith and M. J. Demetsky, “Short-term traffic flow prediction: neural network approach,” Transportation Research Record,vol.1453,1994.
5. M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, and L. D. Han, “Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions,” Expert Systems with Applications,vol.36,no.3,pp.6164–6173,2009.
6. Chen A, Leung MT: Regression neural network for error correction in foreign exchange forecasting and trading. ComputOper Res 31(7):1049-1068 (2004).
7. Nag AK, Mitra A: Forecasting daily foreign exchange rates using genetically optimized neural networks. Journal of Forecasting 21(7):501-511 (2002).
8. Sermpinis G, Stasinakis C, Theofilatos K, Karathanasopoulos A: Modeling, forecasting and trading the EUR exchange rates with hybrid rolling genetic algorithms—Support vector regression forecast combinations. Eur J Oper Res 247(3):831-846(2015).
9. Sermpinis G, Theofilatos K, Karathanasopoulos A, Georgopoulos EF, Dunis C: Forecasting foreign exchange rates with adaptive neural networks using radial-basis functions and particle swarm optimization. Eur J Oper Res 225(3):528-540 (201)
10. Yu L, Wang S, Lai KK: A novel nonlinear ensemble forecasting model incorporating GLAR and ANN for foreign exchange rates. ComputOper Res 32(10):2523-2541 (2005). 13. Yu L, Wang S, Lai KK: Forecasting crude oil price with an EMD-based neural network
11. A.R.Mohamed,G.Dahl,andG.Hinton,“Deepbeliefnetworksforphonerecognition,”inProceedingsoftheNIPSWorkshoponDeepLearningforSpeechRecognitionandRelatedApplications, 2009
12. Perez-Murueta, Pedro; Gómez-Espinosa, Alfonso; Cardenas, Cesar; Gonzalez-Mendoza, Miguel, Jr., "Deep Learning System for Vehicular Re-Routing and Congestion Avoidance," *Applied Sciences*, Vol.9, No. 13, 2019. https://doi.org/10.3390/app9132717.

13. Singh, H., V. Laxmi, A. Malik and IshaBatra. “Current Research on Congestion Control Schemes in VANET: A Practical Interpretation.” International Journal of Recent Technology and Engineering, Vol.8, No.4, pp. 4336-4341, 2019.
14. Wei Kuang Lai, Mei-Tso Lin, Yu-Hsuan Yang, “A Machine Learning System for Routing Decision-Making in Urban Vehicular Ad Hoc Networks,” International Journal of Distributed Sensor Networks, Vol.11, No.3, pp.1-13, 2015.
15. Ahmed, Muhammad & Iqbal, Saleem& Awan, Khalid &Sattar, Kashif& Khan, ZuhaibAshfaq&Sherazi, Husnain, “A Congestion Aware Route Suggestion Protocol for Traffic Management in Internet of Vehicles,”. Arabian Journal for Science and Engineering, 2019. https://doi.org/10.1007/s13369-019-04099-9
16. A. M. de Souza, R. S. Yokoyama, G. Maia, A. Loureiro and L. Villas, "Real-time path planning to prevent traffic jam through an intelligent transportation system," 2016 IEEE Symposium on Computers and Communication (ISCC), Messina, Italy, 2016, pp. 726-731, doi: 10.1109/ISCC.2016.7543822.
17. Pan, J.; Khan, M.A.; Popa, I.S.; Zeitouni, K.; Borcea, C. Proactive Vehicle Re-Routing Strategies for Congestion Avoidance. In Proceedings of the 2012 IEEE 8th International Conference on Distributed Computing in Sensor Systems, Hangzhou, China, 16–18 May 2012; pp. 265–272.
18. Meneguette, R.I.; Ueyama, J.; Krishnamachari, B.; Bittencourt, L. Enhancing Intelligence in Inter-Vehicle Communications to Detect and Reduce Congestion in Urban Centers. In Proceedings of the 20th IEEE symposium on computers and communication (ISCC), Larnaca, Cyprus, 6–9 July 2015; pp. 662–667.
19. A. M. de Souza, R. S. Yokoyama, L. C. Botega, R. I. Meneguette and L. A. Villas, "Scorpion: A Solution Using Cooperative Rerouting to Prevent Congestion and Improve Traffic Condition," 2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, Liverpool, UK, 2015, pp. 497-503, doi: 10.1109/CIT/IUCC/DASC/PICOM.2015.71.
20. Ma, Xiaolei& Dai, Zhuang & He, Zhengbing& Ma, Jihui& Wang, Yong & Wang, Yunpeng. “Learning Traffic as Images: A Deep Convolutional Neural Network for Large-Scale Transportation Network Speed Prediction,” Sensors, Vol.17.No. 818, 2017. 10.3390/s17040818.
21. M. Bachl, T. Zseby and J. Fabini, "Rax: Deep Reinforcement Learning for Congestion Control," ICC 2019 - 2019 IEEE International Conference on Communications (ICC), Shanghai, China, 2019, pp. 1-6, doi: 10.1109/ICC.2019.8761187.
22. M. Bachl, T. Zseby and J. Fabini, "Rax: Deep Reinforcement Learning for Congestion Control," *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, Shanghai, China, 2019, pp. 1-6, doi: 10.1109/ICC.2019.8761187.
23. Nathan Jay, NogaRotman, Brighten Godfrey, Michael Schapira, Aviv Tamar, “A Deep Reinforcement Learning Perspective on Internet Congestion Control,” International Conference on Machine Learning, Proceedings of Machine Learning Research, Vol.97, pp.3050-3059, 2019.