sciendo

Journal of Smart Internet of Things (JSIoT)
VOL 2023, No.02 | 44-61 | 2023
DOI: 10.2478/jsiot-2023-0010

Article

Intelligent Traffic Congestion Control Using Black Widow Optimization with Hybrid Deep Learning on Smart City Environment

Sarah Hadi Shaheed[1]

[1] School of Computing, Universiti Utara Malaysia, Changlun, Malaysia

*Corresponding author: Sarah Hadi Shaheed (email: sarah-hadi@ahsgs.uum.edu.my)

Received 20 August 2023; Accepted 21 October 2023; Issue Published 15 December 2023

Abstract: Intelligent traffic congestion control is an integral aspect of making sustainable and efficient smart cities. With the increasing count of vehicles on the road and rapid urbanization, traffic congestion is a main concern nowadays that hinders the growth of the economy and affects the quality of life. In smart cities, an intelligent transportation solution (ITS) is enhance traffic flow by adjusting traffic signal timing and observing traffic patterns. Currently, one of the vital dilemmas in terms of transportation systems was traffic congestion which needs to be resolved for minimizing driver frustration, traffic jams, fuel waste, and accidents. Due to the high count of vehicles, most of the traffic interruptions in metropolitan cities arise. With the advancements in Artificial Intelligence (AI) and Machine Learning (ML), smart environments monitored in smart cities observe the influencing issues of the environment correctly, with the best control of traffic congestion, pollution, and other negative effects. Therefore, this study presents an intelligent traffic congestion control using Black Widow Optimization with Hybrid Deep Learning (ITC-BWOHDL) technique in Smart City Environment. The main aim of the ITC-BWOHDL technique is to utilize feature subset selection with parameter-tuning strategies for effective traffic congestion management. To obtain this, the ITC-BWOHDL technique primarily designs the emperor penguin optimizer-based feature selection (EPO-FS) approach for selecting a useful set of features. For the detection of traffic congestion, the ITC-BWOHDL technique makes use of the HDL model which incorporates convolutional neural network (CNN) with gated recurrent unit (GRU) approach. To improve the classification results of the HDL model, the BWO-based hyperparameter tuning process gets executed. For exhibiting the improved classification outcome of the ITC-BWOHDL system, a comprehensive range of experiments was executed. The obtained outcome described the betterment of ITC-BWOHDL method over other existing techniques.

Keywords: Smart city; Intelligent transportation system; Hybrid deep learning; Black widow optimization; Feature selection

Copyright © 2023 Journal of Smart Internet of Things (JSIoT) published by Future Science for Digital Publishing and Sciendo . This is an open access article license CC BY (https://creativecommons.org/licenses/by/4.0/)

1. Introduction

In smart cities, the intellectual transportation network has grabbed considerable attention in computer vision (CV) to avoid accidents and traffic congestion [1]. Traffic congestion may occur if the traffic increases and the speed of the object turn out to be slow. It may cause numerous difficulties like mental pressure, fuel consumption, aggravating air pollution, and time loss [2]. Different traffic stakeholders namely policymakers, government agencies, city planners and architects, road users, traffic managers, etc., receive the advantage of traffic congestion management. Traffic congestion management makes roads safe to travel, aids to minimize time travel, and increases road capacity [3]. Traffic congestion prediction removes issues like fuel consumption, air pollution, and noise pollution which further helps to maintain a sustainable environment. Hence the researcher works in the field of traffic congestion prediction so that one can contribute to society [4]. Precise Traffic data collection enabled reliable statistics which in turn produced economic advantages by assisting decision makers where to invest in safety enhancement [5]. A vital step toward constructing smart cities commenced by gathering targeted reliable data, and examining them to apply sustainable solutions. Some of the challenges of accumulating transportation data are human implications and the size of data to be processed mainly when video images are included or real-time traffic images should be processed [6].

The two major representative data-driven approaches for traffic prediction are statistical and conventional machine learning (ML) methods [7]. In time-series analysis, one of the most consolidated methods that depend on traditional statistics is the autoregressive integrated moving average (ARIMA) and its variations were broadly implemented for traffic prediction problems. On the other hand, such approaches are mostly devised for small datasets and are unsuitable to deal with dynamic and complex time series data [8]. Furthermore, as only temporal data is considered, the spatial dependency of traffic data is barely considered or ignored. Deep learning (DL) and ML methods can present insight from data captured by IoT gadgets utilizing predictive, diagnostic, perspective, and descriptive analysis methods. DL /ML approaches aid to estimate or predict traffic flow with precision [9]. This aids to resolve the issue of traffic congestion in smart cities which assists people to make decisions while travelling. Predictive analytics becomes a crucial part of smart society as it aids in making decisions in terms of outcomes [10]. DL and ML methods can do miracles to gain insights from any type of data semi-structured, structured, or unstructured. Smart cities have several applications like flow prediction or traffic congestion, parking space estimation, and so on. Where predictive analytics is beneficial to make the life of the public easier.

This study presents an intelligent traffic congestion control using Black Widow Optimization with Hybrid Deep Learning (ITC-BWOHDL) technique in Smart City Environment. The main aim of the ITC-BWOHDL technique is to utilize feature subset selection with parameter-tuning strategies for effective traffic congestion management. To obtain this, the ITC-BWOHDL technique primarily designs the emperor penguin optimizer-based feature selection (EPO-FS) approach for selecting a useful set of features. For the detection of traffic congestion, the ITC-BWOHDL technique makes use of the HDL model which incorporates convolutional neural network (CNN) with gated recurrent unit (GRU) approach. To improve the classification results of the HDL algorithm, the BWO-based hyperparameter tuning process gets executed. For exhibiting the greater classification outcome of the ITC-BWOHDL approach, a comprehensive range of experiments was executed.

2. Related Works

Qi et al. [11] present a DL structure that depends on federated learning and asynchronous GCN (FedAGCN) to forecast traffic flow precisely in real-time. To model the spatial–temporal dependence in traffic data, FedAGCN implemented asynchronous spatial–temporal graph

convolution. To minimalizing the time cost of the DL method, the author presents a graph FL method, GraphFed to train the method. In [12], the author merged prediction and graph optimization in one pipeline to inspect a new convolution graph-based NN for urban traffic flow estimation from an edge IoT platform. The resulting graph was applied for training an extended graph CNN for predicting traffic flow in the city.

Saleem et al. [13] modelled a FITCCS-VN abbreviated as fusion-related ITS control system for VNs with ML approaches which gather traffic datasets and route traffic on accessible routes for alleviating traffic congestion in smart cities. The presented structure presents new services to drivers that enabled the size of vehicles existing on the road and view traffic flow remotely, aiming to avoid traffic jams. Dai et al. [14] introduce ST-InNet, a deep spatiotemporal Inception system to predict traffic in all city regions. To be Specific, ST-InNet uses 2 Inception network to concurrently capture different spatiotemporal correlation of traffic dataset that includes spatial dependencies, temporal closeness, distant spatial dependencies, and temporal periodicity.

Hassan et al. [15] modelled a method that depends on a DL structure with a multi-layer ELM assuming congestion data at every feasible connection point to smooth a signal functioning done that recorded data. The authors [16] introduced an attention-related CNN-LSTM, a multi-step predictive technique. The presented system makes use of the spatial and time-related details of traffic data sets that can be derived through LSTM and CNN networks to enhance the method's accuracy. The attention-related method aids to find the near-term traffic details like speed that are significant to predict the future value of flow.

In [17], developed a hybrid method named boosted LSTM ensemble (BLSTME) in addition to CNNs for helping cars navigate nearby congested roads by presenting CNN great features allow for BLSTME that supports cars in dynamic atmospheres by predicting the probability of congestion. In [18], showed that noise pollution and traffic time-series data are utilized for training LSTM-RNNs that resulted in better traffic forecasting on the road. This method was utilized with pollution signs. This study addressed a novel investigation path relevant to the usage of signature profiles and AI methods to minimalize the focus on sensing structure.

3. The Proposed Model

To enhance traffic congestion control in smart cities, this paper has developed an automated ITC-BWODHL technique for congestion detection. The ITC-BWOHDL technique intends to utilize feature subset selection with parameter-tuning strategies for effective traffic congestion management. It comprises a three-step procedure such as EPO-FS-based feature selection, HDL-based congestion detection, and BWO-based hyperparameter tuning. Fig. 1 represents the working flow of the ITC-BWODHL approach.

3.1. Feature Selection using EPO-FS Algorithm

The EPO-FS algorithm is used for the identification of a useful set of features. The EPO is based on the foraging and huddling behaviors of emperor penguins (EPs) usually initiate nearby the Antarctic region [19]. The EP forages in a raft or group. This depicts the huddling action that protects them in extremely lower temperatures. For the optimizer, the optimum solution was similar to the warmest EP. The fundamental goal is to determine an effective EP that can be accomplished by calculating the distance between the EPs (X_{ep}) based on the temperature profiles. The movement of EP is directed and it is progress towards the optimal value. Firstly, the temperature profile of every EP was assessed by the following equations:

$$\theta' = \left(\theta - \frac{I_{itermax}}{C - I_{itermax}}\right) \quad (1)$$

$$\theta = \begin{cases} 0 \; if \; R > 0.5 \\ 1 \; if R \; < 0.5 \end{cases} \quad (2)$$

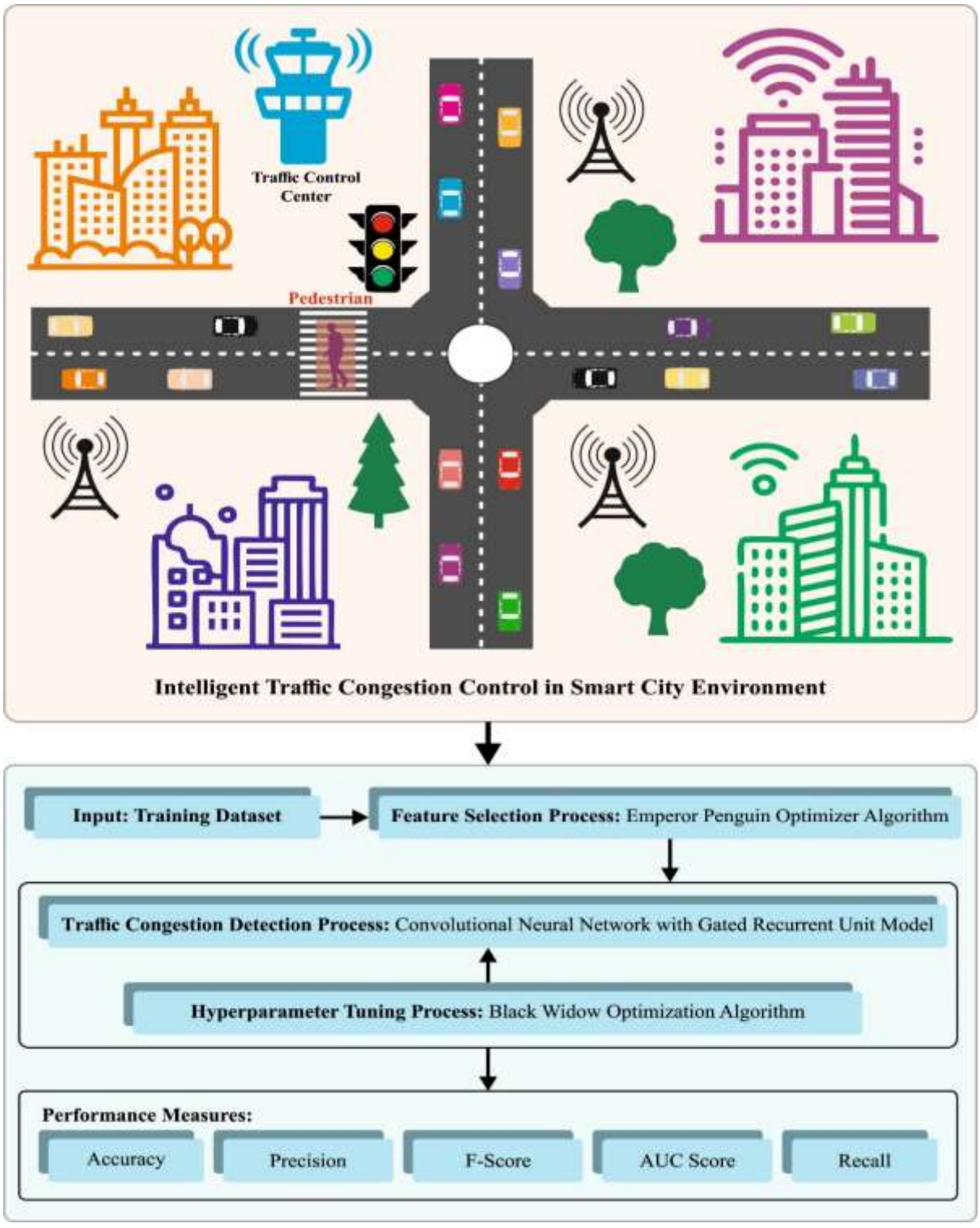


Figure 1: Workflow of ITC-BWODHL system

Where $I_{iter\max}$ denotes the maximal iteration counts, C indicates the existing iteration, and R represents the arbitrary number uniform distribution. Eq. (1) govern the exploration and

exploitation phases.

In the huddling method, there is the possibility of collision in neighboring EPs. Two vectors are introduced (reff), such as $\vec{U}$ and $\vec{V}$ to avoid this collision:

$$\vec{U} = \left\{M \times \left(\theta^{'} + X_{grid}(acc)\right) \times Rand()\right\} - \theta^{'} \quad (3)$$

$$\vec{V} = Rand(\quad) \quad (4)$$

$$X_{grid}(acc) = \left|\vec{X} - \vec{X}_{ep}\right| \quad (5)$$

In the formula, M indicates the movement parameter which is fixed to 2. $\vec{X}$ characterizes the optimal solution and $\vec{X}_{ep}$ signifies the position of other EPs. The estimation of the distance between the optimal EPs and other EPs is shown below:

$$\vec{D} = \left|\left\{S\left(\vec{U}\right).\vec{X}(x) - \vec{V}.\vec{X}_{ep}\right\}\right| \quad (6)$$

$$S\left(\vec{U}\right) = \sqrt{\left(f \cdot e^{\frac{-c}{y}} - e^{-C}\right)^2} \quad (7)$$

Where $S\left(\vec{U}\right)$ refers to the similarities to social force expert by optimum EP in other EPs. f and e are the control factors that are set at three and two correspondingly.

The location of EPs is upgraded based on the following equation:

$$\vec{X}_{ep}(x+1) = \vec{X}(x) - \vec{U}.\vec{D}_{ep} \quad (8)$$

In Eq. (8), the existing and next locations of EPs are represented as $\vec{X}(x)$ and $\vec{X}(x+1)$ correspondingly.

The FF of EPO-FS method regards the count of FS and the classifier accuracy. It decreases the set dimensional of FSs and enhances the classifier performance. Thus, the following FF was utilized for evaluating individual solutions, as follows.

$$Fitness = \alpha * ErrorRate + (1-\alpha) * \frac{\#SF}{\#All_F} \quad (9)$$

Where α is used for controlling the significance of subset length and classifier quality. $ErrorRate$ refers to the classifier rate of errors using the FS. $ErrorRate$ is estimated as the percentage of incorrectly classified to the count of classifiers made, within 0 and 1. $\#SF$ denotes the number of FS, and $\#All_F$ indicates the overall amount of attributes in the original dataset.

3.2. Traffic Congestion Detection using HDL Model

To detect the congestion in the smart city, the HDL model is used. GRU is a novel infrastructure to resolve the exploding or vanishing gradient problems. The upgraded LSTM architecture is named GRU [20]. Also, GRU has a gate structure than LSTM for regulating the data flow. Different from LSTM, GRU lacks an output gate, which allows the content that revealed completely. The update and reset gates are two different gates of GRU. In the second gate, the input and forget gates of LSTM architecture are integrated. In contrast with LSTM, GRU has lesser parameters and a simple

structure that enhances efficiency.

The equation below provides the GRU formula.

$$r_t = sigm(W_{xr}x_t + W_{hr}h_{t-1} + b_r) \quad (10)$$

$$z_t = sigm(W_{xz}x_t + W_{hz}h_{t-1} + b_z) \quad (11)$$

$$\widetilde{h_t} = tanh(W_{xh}x_t + W_{hh}(r_t \odot h_{t-1}) + b_h) \quad (12)$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \widetilde{h_t}) \quad (13)$$

Let 'x_t', 'h_t' 'r_t', and 'z_t' be the input and output vectors, reset and update gates correspondingly. 'b' refers to bias and W' denotes the weight, '$Sigm$' and '$tanh'$, correspondingly shows sigmoid and tangent activation functions. The LSTM and GRU can handle the long dependency. But there are some variations in terms of performance. For evaluating how well they categorized network traffics, we employed both architectures in this study.

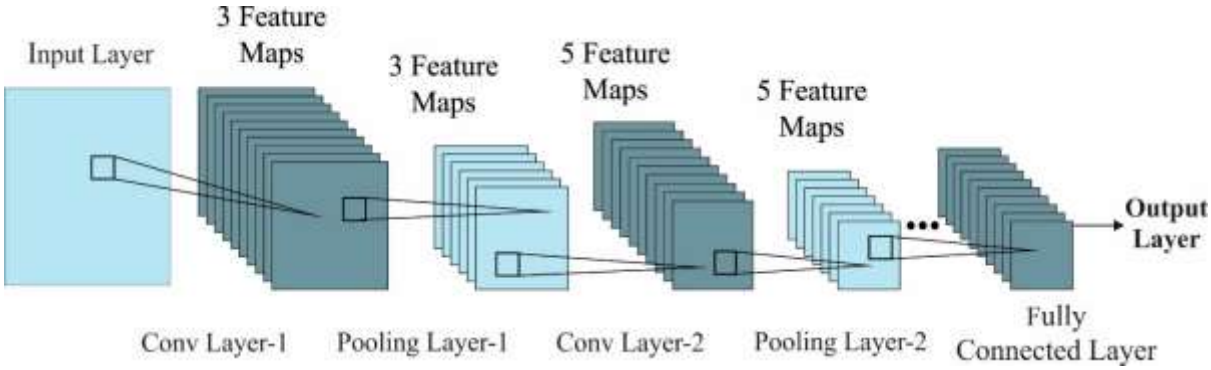


Figure 2: Architecture of CNN

The core structure of CNN consists of three principal components, such output, convolution, and pooling layers. Fig. 2 displays the infrastructure of CNN. The classical CNN architecture with three convolution layers are widely applied in image classification. It has a single input layer, multiple hidden states (namely, normalization, pooling, and convolutional layers), and a fully connected (FC) layer to the last layer known as resultant layer. Input and pooling processes are implemented to decrease the proportion of the sub-sampling. CNN classifiers receive the input image as a collection of smaller sub-section which are named receptive fields.

The CNN-GRU model includes 1 hidden layer (H), 3 convolution layers (C), and 2 GRU layers (G). The neuron for the last layer is corresponding to the labels from the database. In the final layer, the softmax ($S_\max$) activation function, providing the output based on the predictive possibility. The vector of numbers was changed as a vector of probabilities with the use of Softmax, but the probability value can be inversely proportional to the relative scale.

3.3.Modeling of BWO-based Hyperparameter Tuning

The BWO algorithm is utilized in this work for the optimum selection of hyperparameters related to the HDL model. BWO is a new optimization method based on the unique mating behaviors of BW spiders [21]. The process mimics the life cycle of BW spiders and has several benefits with respect to the avoidance of local optimality, convergence speed, and fitness optimization. BPNN is relatively mature with respect to network performance and theory, with a flexible network structure and stronger non-linear mapping abilities. But BPNN has certain shortcomings. As BPNN exploits the gradient learning model, the convergence rate can be

inevitably slower, and it simply gets trapped in the local extreme. The study found that the BWO can efficiently avoid this problem. Thus, the study applied BWO for optimizing the weight of all the layers of the BPNN and thresholds to optimize the global search capability and convergence performance BWO including the subsequent five phases.

The initial population is made up of N_{pop} BWs, where all the BWs signify a potential solution, and the BW spiders were regarded as a 1D array as follows:

$$Widow = [x_1, x_2, \cdots, x_{M_{var}}] \tag{14}$$

In Eq. (14), $x_{M_{var}}$ denotes the dimension of the optimization problem, and all the variables $(x_1, x_2, \cdots, x_{M_{var}})$ are random floating numbers. All the BWs evaluate the fitness values by the fitness function f as follows:

$$Fitness = f(widow) = f(x_1, x_2, \cdots, x_{M_{var}}) \tag{15}$$

N_{pop} BWs are generated once the population is initialized, and a $N_{pop} \times M_{var}$ BW matrix can be attained.

The procreation stage is a global search process. Firstly, according to fitness, the population is ranked, later the BWs included in procreation are evaluated based on the procreating rate (PP), and lastly, parents are chosen at random for procreation based on Eq. (16). Here, the procreating stage can be simulated by making α arrays.

$$\begin{cases} y_1 = \alpha \times x_1 + (1-\alpha) \times x_2 \\ y_2 = \alpha \times x_2 + (1-\alpha) \times x_1 \end{cases} \tag{16}$$

In Eq. (16), x_1 and x_2 represent the parents and y_1 and y_2 denote the offspring, and the procedure is reiterated $M_{VAR}/2$ times.

The process involves three different types of cannibalism: child-eat-mother, sexual, and sibling cannibalism. Child-eat-mother cannibalism refers to once the child spider consumes its mother. Sibling cannibalism is once the stronger spiderling consumes the weak sibling. Sexual cannibalism shows that female BW with higher fitness value would consume the male BW with lower fitness value after or during mating. The BWO accomplishes sibling cannibalism by destroying any children of the father and sexual cannibalism by abolishing based on the cannibalism rate (CR).

The mutation is a local search process. Based on the mutation rate (PM), the BWO chooses multiple BWs randomly, and all the BWs interchange both eigenvalues randomly in the array for completing the mutation behaviors.

BWs with maximum fitness value attained after passing over the abovementioned 4 phases are exploited as the initial population for iteration until the ending criteria are met.

Fitness selection is a key factor in the BWO approach. An encoded outcome was utilized for evaluating the goodness of candidate solutions. Presently, the accuracy value is the primary condition exploited for scheming FF.

$$Fitness = \max(P) \tag{17}$$

$$P = \frac{TP}{TP + FP} \tag{18}$$

Where TP and FP signify the true and false positive values.

4. Results and Discussion

In this section, the simulation result of the ITC-BWOHDL algorithm is tested utilizing the road traffic dataset from the Kaggle dataset [22]. It has 1879 samples with two classes namely traffic

congestion (TC) and no traffic congestion (NTC) as shown in Table 1. The dataset has 13 features such as region_id, road_category_name, region_name, road_category_description, total_link_length_km, total_link_length_miles, buses_and_coaches, pedal_cycles, cars_and_taxis, two_wheeled_motor_vehicles, all_motor_vehicles, and lgvs, all_hgvs. In addition, the EPO-FS technique has elected 6 features namely region_name, road_category_description, road_category_name, total_link_length_miles, total_link_length_km, and all_hgvs.

Table 1: Details of database

Class	**No. of Samples**
Traffic Congestion	293
No Traffic Congestion	1586
Total No. of Samples	**1879**

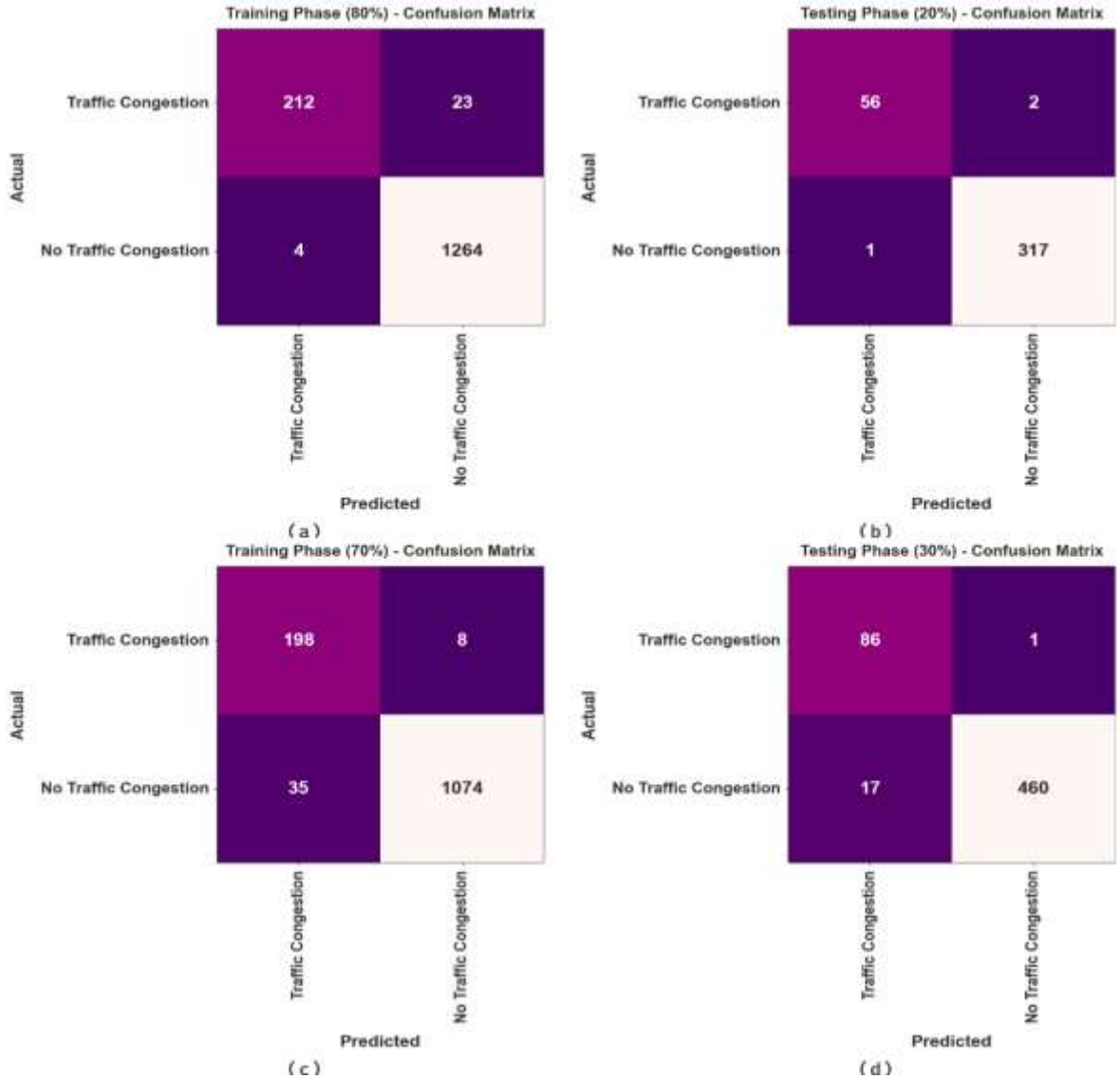


Figure 3: Confusion matrices of ITC-BWOHDL system (a-b) 80:20 of Training Phase (TRPH)/Testing Phase (TSPH) and (c-d) 70:30 of TRPH/TSPH

In Fig. 3, the confusion matrices of the ITC-BWOHDL approach are elucidated. The figure shows that the ITC-BWOHDL technique properly recognizes the TC and NTC samples.

The traffic congestion detection outcomes of the ITC-BWOHDL technique under 80:20 of TRPH/TSPH are reported in Table 2. In Fig. 4, the brief classifier outcomes of the ITC-BWOHDL system on 80% of TRPH are portrayed. The result revealed that the ITC-BWOHDL technique exhibits effectual outcomes under two cases. For instance, on TC class, the ITC-BWOHDL technique offers $accu_y$ of 90.21%, $prec_n$ of 98.15%, $reca_l$ of 90.21%, F_{score} of 94.01%, and AUC_{score} of 94.95%. Also, on NTC class, the ITC-BWOHDL approach provides $accu_y$ of 99.68%, $prec_n$ of 98.21%, $reca_l$ of 99.68%, F_{score} of 98.94%, and AUC_{score} of 94.95%.

Table 2: Traffic congestion detection outcome of ITC-BWOHDL system on 80:20 of TRPH/TSPH

Class	$Accu_y$	$Prec_n$	$Reca_l$	F_{Score}	AUC_{Score}
TRPH (80%)					
Traffic Congestion	90.21	98.15	90.21	94.01	94.95
No Traffic Congestion	99.68	98.21	99.68	98.94	94.95
Average	**94.95**	**98.18**	**94.95**	**96.48**	**94.95**
TSPH (20%)					
Traffic Congestion	96.55	98.25	96.55	97.39	98.12
No Traffic Congestion	99.69	99.37	99.69	99.53	98.12
Average	**98.12**	**98.81**	**98.12**	**98.46**	**98.12**

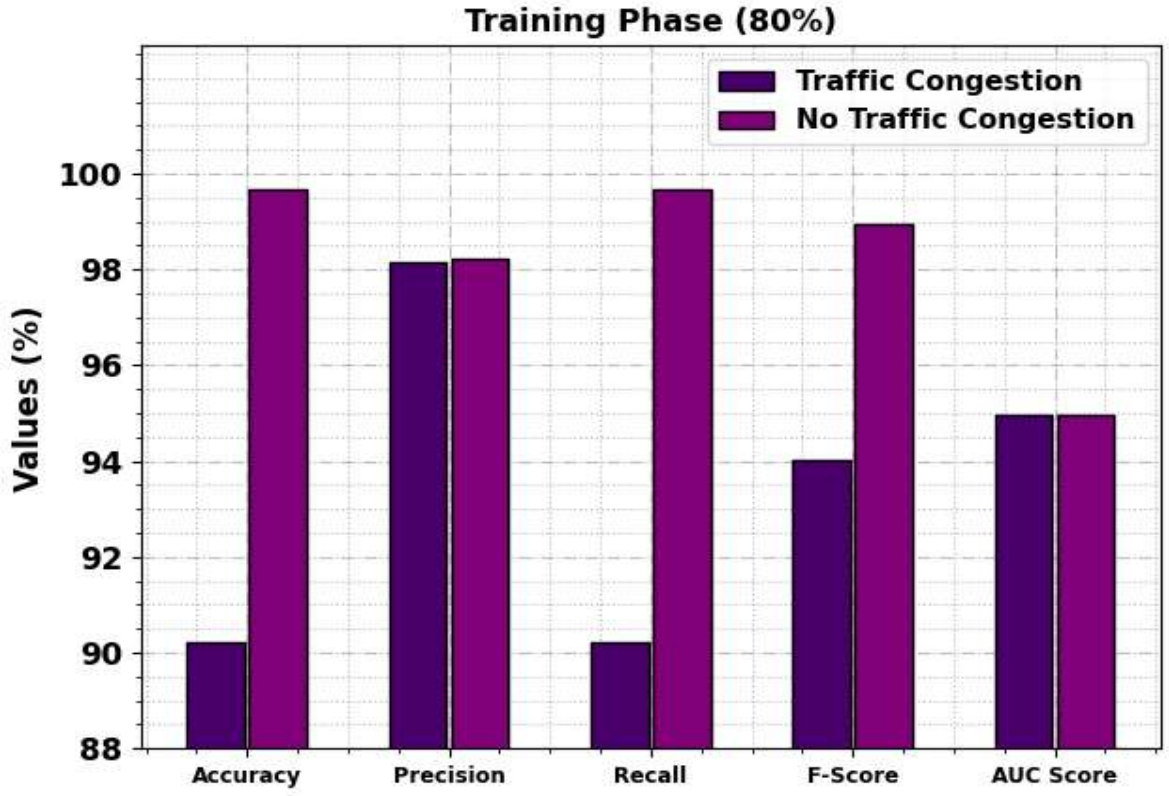


Figure 4: Traffic congestion detection outcome of ITC-BWOHDL system on 80% of TRPH

In Fig. 5, the brief classifier outcomes of the ITC-BWOHDL system on 20% of TRPH is

exposed. The outcomes depicted that the ITC-BWOHDL approach displays effective outcomes under two cases. For instance, on TC class, the ITC-BWOHDL technique offers $accu_y$ of 96.55%, $prec_n$ of 98.25%, $reca_l$ of 96.55%, F_{score} of 97.39%, and AUC_{score} of 98.12%. Also, on NTC class, the ITC-BWOHDL approach offers $accu_y$ of 99.69%, $prec_n$ of 99.37%, $reca_l$ of 99.69%, F_{score} of 99.53%, and AUC_{score} of 98.12%.

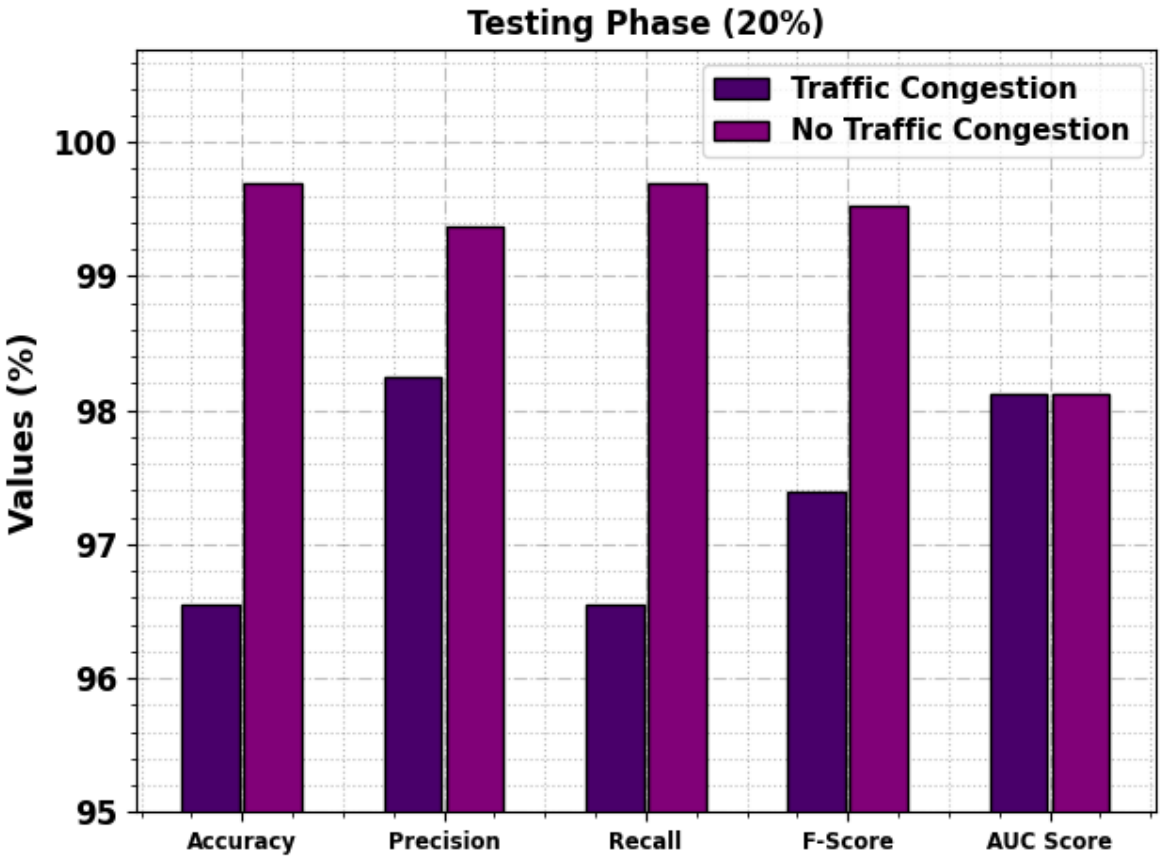


Figure 5: Traffic congestion detection outcome of ITC-BWOHDL approach on 20% of TSPH

The traffic congestion detection outcome of the ITC-BWOHDL system on 70:30 of TRPH/TSPH is described in Table 3. In Fig. 6, the detailed classifier outcomes of the ITC-BWOHDL approach on 70% of TRPH is revealed. The outcome outperformed that the ITC-BWOHDL system displays effective outcomes under two cases. For instance, on TC class, the ITC-BWOHDL methodology provides $accu_y$ of 96.12%, $prec_n$ of 84.98%, $reca_l$ of 96.12%, F_{score} of 90.21%, and AUC_{score} of 96.48%. In addition, on NTC class, the ITC-BWOHDL algorithm offers $accu_y$ of 96.84%, $prec_n$ of 99.26%, $reca_l$ of 96.84%, F_{score} of 98.04%, and AUC_{score} of 96.48%.

Table 3: Traffic congestion detection outcome of ITC-BWOHDL system on 70:30 of TRPH/TSPH

Class	$Accu_y$	$Prec_n$	$Reca_l$	F_{Score}	AUC_{Score}
TRPH (70%)					
Traffic Congestion	96.12	84.98	96.12	90.21	96.48
No Traffic Congestion	96.84	99.26	96.84	98.04	96.48
Average	**96.48**	**92.12**	**96.48**	**94.12**	**96.48**
TSPH (30%)					

Traffic Congestion	98.85	83.50	98.85	90.53	97.64
No Traffic Congestion	96.44	99.78	96.44	98.08	97.64
Average	**97.64**	**91.64**	**97.64**	**94.30**	**97.64**

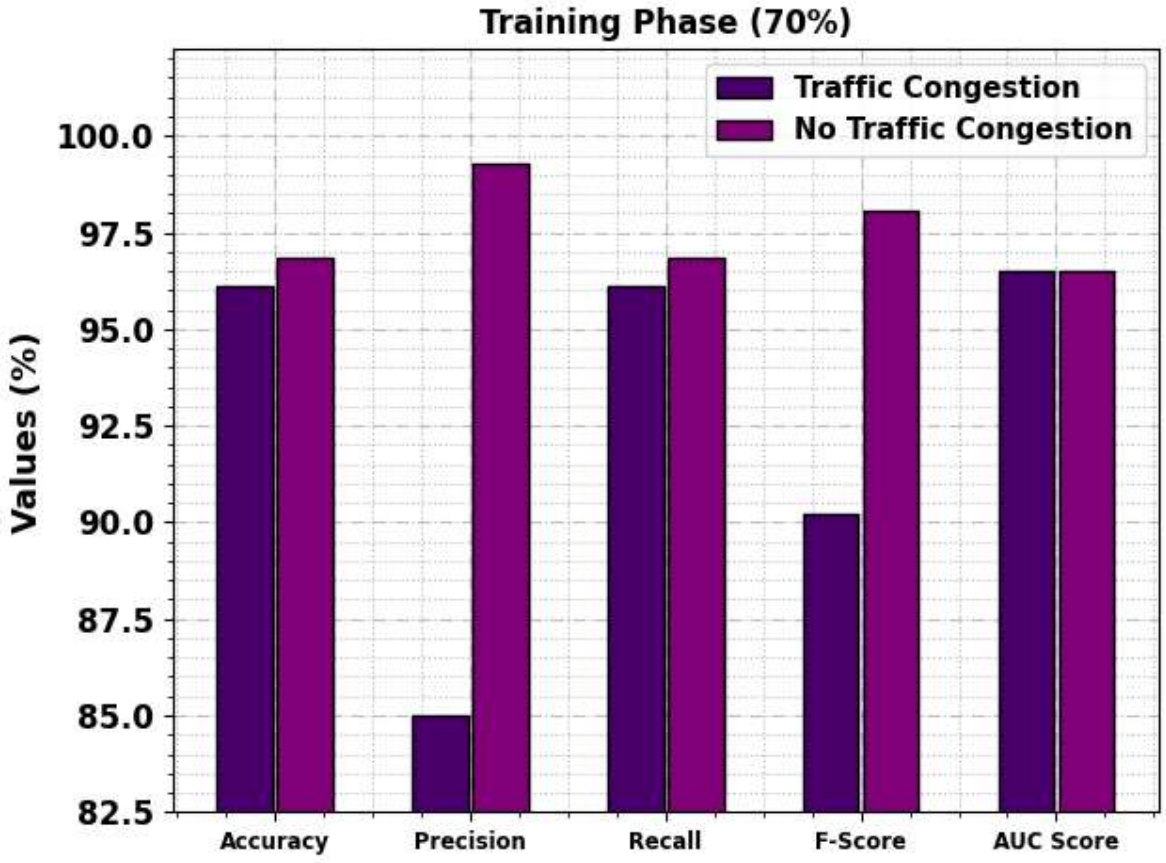


Figure 6: Traffic congestion detection outcome of ITC-BWOHDL system on 70% of TRPH

In Fig. 7, the comprehensive classifier outcomes of the ITC-BWOHDL approach on 30% of TRPH is represented. The result exhibited that the ITC-BWOHDL technique implies capable outcomes under 2 cases. For instance, on TC class, the ITC-BWOHDL method provides $accu_y$ of 98.85%, $prec_n$ of 83.50%, $reca_l$ of 98.85%, F_{score} of 90.53%, and AUC_{score} of 97.64%. Besides, on NTC class, the ITC-BWOHDL system offers $accu_y$ of 96.44%, $prec_n$ of 99.78%, $reca_l$ of 96.44%, F_{score} of 98.08%, and AUC_{score} of 97.64%.

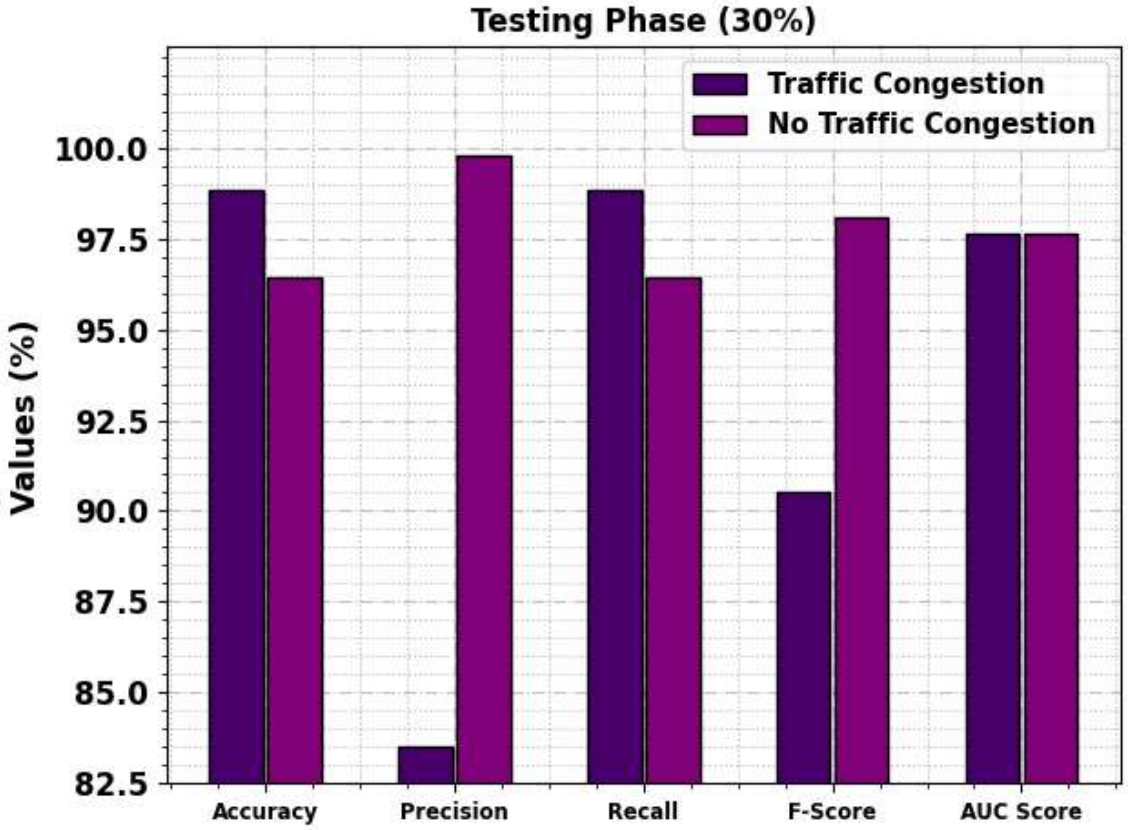


Figure 7: Traffic congestion detection outcome of ITC-BWOHDL system on 30% of TSPH

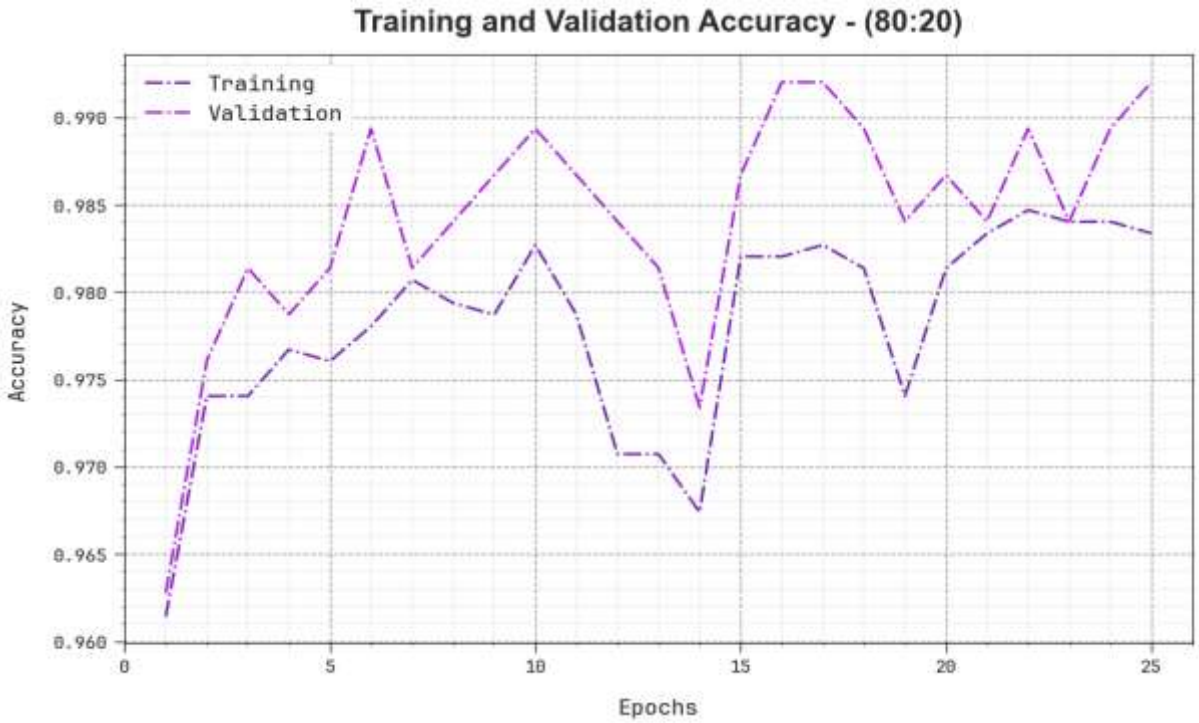


Figure **8:** $Accu_y$ curve of ITC-BWOHDL method on 80:20 of TRPH/TSPH

Fig. 8 studies the $accu_y$ of the ITC-BWOHDL system at training and validation procedure on the test data set. The result implies that the ITC-BWOHDL method obtains highest $accu_y$ values over maximal epochs. Furthermore, the enhancing validation $accu_y$ above training $accu_y$ shows that the ITC-BWOHDL algorithm obtains capably on the test data set.

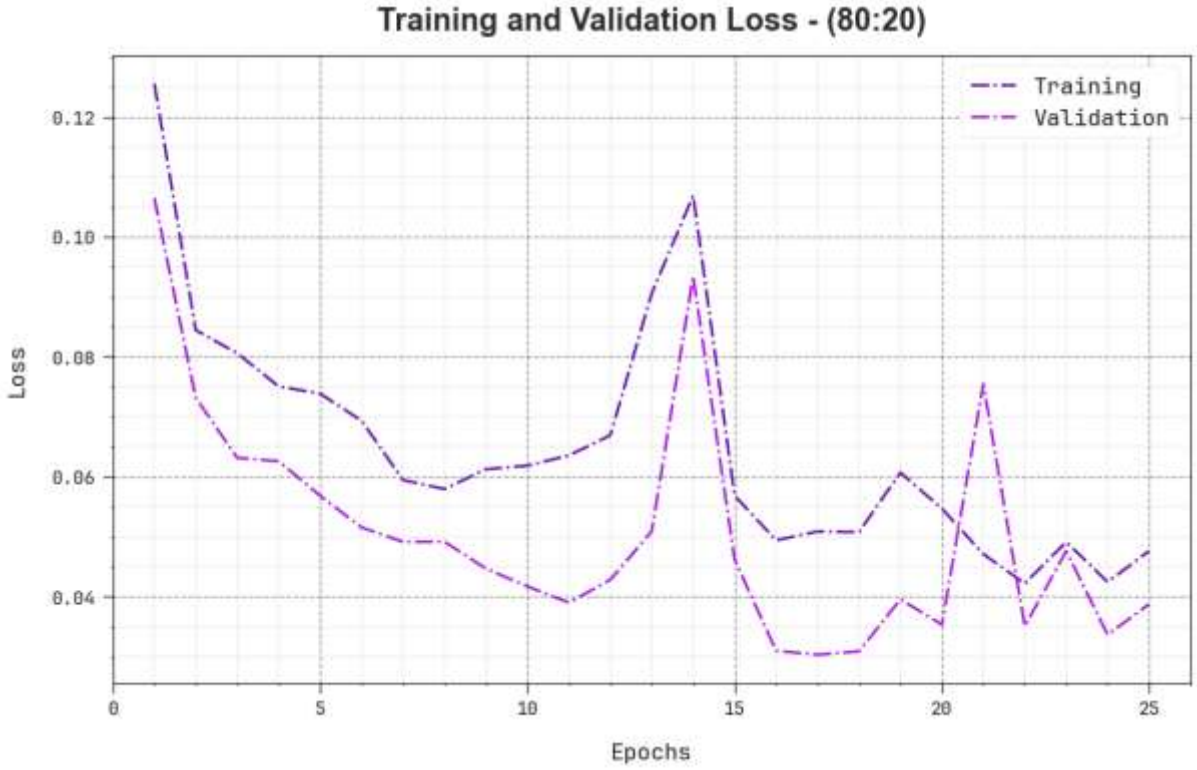


Figure 9: Loss curve of ITC-BWOHDL system on 80:20 of TRPH/TSPH

The loss curve of ITC-BWOHDL system in training and validation was displayed on the test data set in Fig. 9. The outcome inferred that the ITC-BWOHDL algorithm accomplishes nearby values of training and validation loss. The ITC-BWOHDL methodology achieves effectiveness on the test data set.

A detailed PR curve of the ITC-BWOHDL system was displayed on the test database in Fig. 10. The outcome inferred that the ITC-BWOHDL technique outcomes enhancing values of PR. Furthermore, it could be evidence that the ITC-BWOHDL method is gain enhanced PR values in 2 classes.

In Fig. 11, a ROC curve of the ITC-BWOHDL methodology can be exposed on the test database. The outcome implied that the ITC-BWOHDL algorithm leads to superior ROC values. Additionally, it can be apparent that the ITC-BWOHDL system is extend enhanced ROC values on 2 classes.

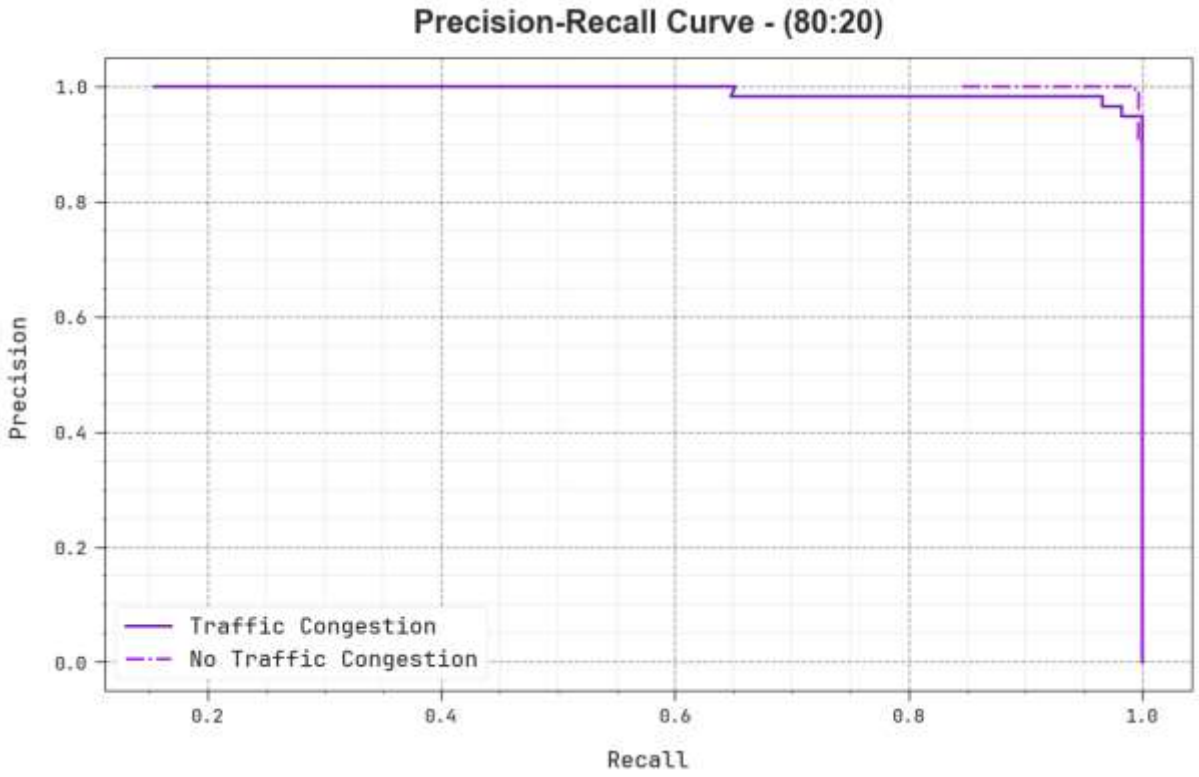


Figure 10: PR curve of ITC-BWOHDL system on 80:20 of TRPH/TSPH

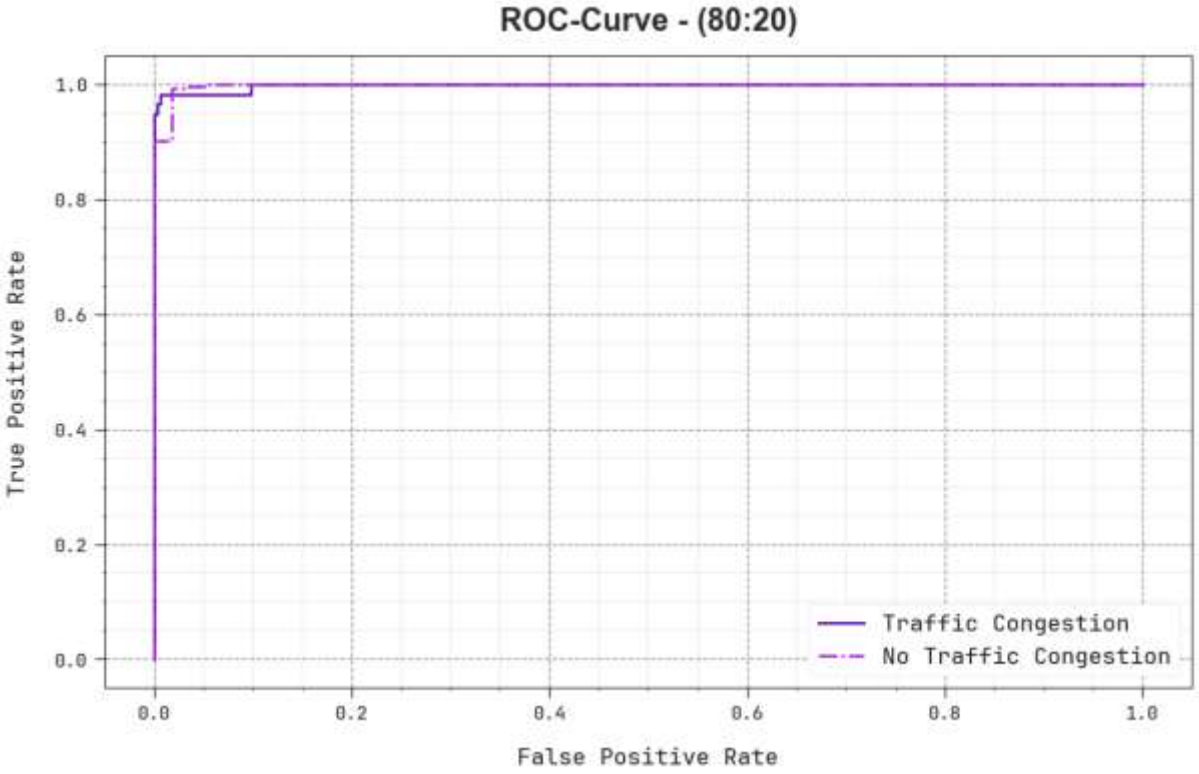


Figure 11: ROC curve of ITC-BWOHDL system on 80:20 of TRPH/TSPH

The overall comparison analysis of the ITC-BWOHDL technique with recent approaches is

made in terms of $accu_y$ and miss rate (MR) is portrayed in Table 4 [13]. Fig. 12 compares the experimental outcomes of the ITC-BWOHDL method with existing techniques in terms of $accu_y$. The result inferred that the CRTCL-VMP and CUAV-AI TCC models obtain poor performance with lower $accu_y$ values of 91.29% and 91.67% correspondingly.

Table 4: Comparative analysis of the ITC-BWOHDL method with existing techniques

Techniques	Accuracy (%)	Miss-Rate (%)
CRTCL-VMP	91.29	8.71
CUAV-AI TCC	91.67	8.33
ANN Model	95.40	4.60
SVM Model	93.50	6.50
FITCCS-VN Model	95.00	5.00
ITC-BWOHDL	98.12	1.88

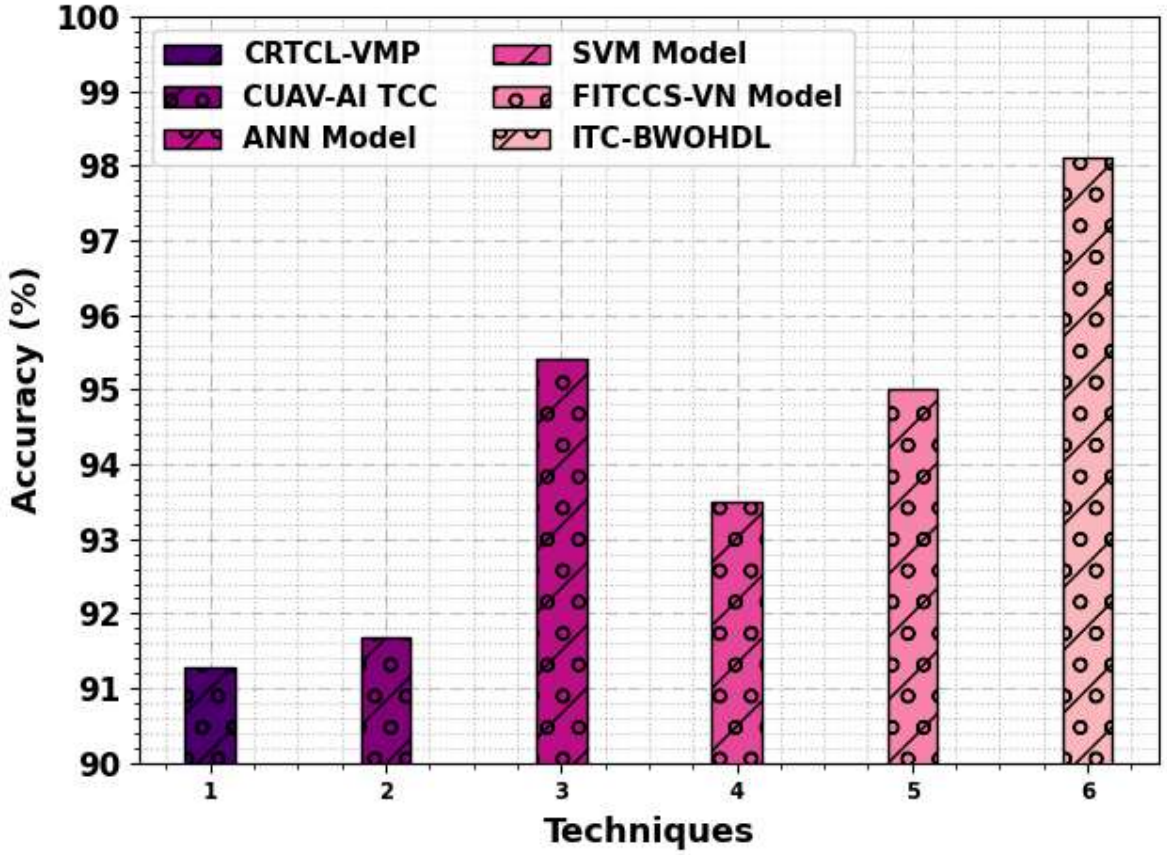


Figure 12: $Accu_y$ analysis of ITC-BWOHDL approach with recent systems

Also, the ANN, SVM, and FITCCS-VN techniques accomplish slightly improved $accu_y$ values of 95.40%, 93.50%, and 95%. Nevertheless, the ITC-BWOHDL technique demonstrated a higher $accu_y$ of 98.12%.

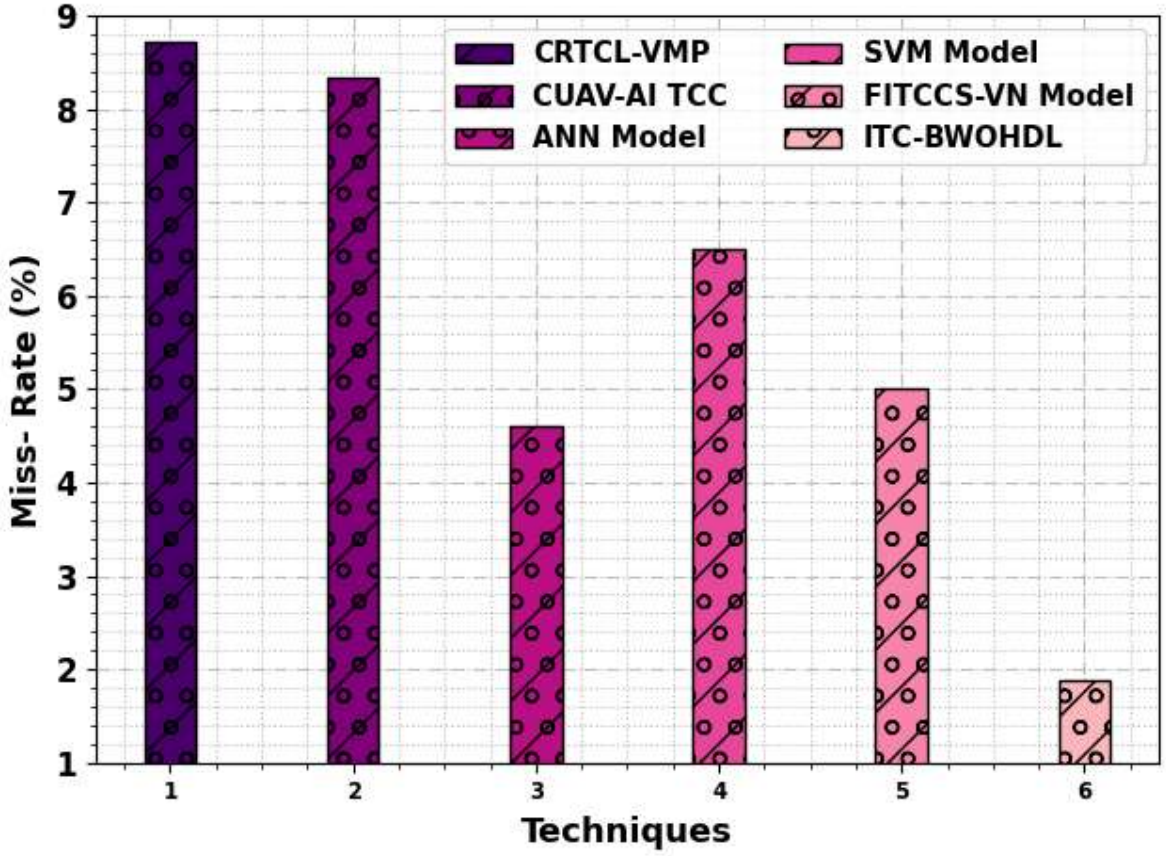


Figure 13: Miss rate analysis of ITC-BWOHDL approach with recent systems

Fig. 13 compares the experimental outcome of the ITC-BWOHDL approach with existing systems with respect to MR. The outcomes implied that the CRTCL-VMP and CUAV-AI TCC approaches gained the least performance with superior MR values of 8.71% and 8.33% correspondingly. In addition, the ANN, SVM, and FITCCS-VN systems realize somewhat lesser MR values of 4.60%, 6.50%, and 5%. However, the ITC-BWOHDL method revealed a minimum MR of 1.88%. These results shown the superior performance of the proposed technique.

5. **Conclusion**

This paper has developed an automated ITC-BWODHL technique for congestion detection to improve traffic congestion control in smart cities. The ITC-BWOHDL technique intends to utilize feature subset selection with parameter-tuning strategies for effective traffic congestion management. It comprises a three-step procedure such as EPO-FS-based feature selection, HDL-based congestion detection, and BWO-based hyperparameter tuning. The ITC-BWOHDL technique primarily designed the EPO-FS approach to reduce computation complexity and improve classification results. The BWO-based hyperparameter tuning process gets executed to enhance the classification outcomes of the HDL algorithm. For exhibiting the greater classification outcome of the ITC-BWOHDL system, a comprehensive range of experiments was executed. The achieved outcomes stated the betterment of the ITC-BWOHDL algorithm over other models. In future, real-time traffic data can be collected to examine the congestion control results of the ITC-BWOHDL technique.

References

[1]Mall, P. K., Narayan, V., Pramanik, S., Srivastava, S., Faiz, M., Sriramulu, S., & Kumar, M. N. (2023). FuzzyNet-Based Modelling Smart Traffic System in Smart Cities Using Deep Learning Models. In Handbook of Research on Data-Driven Mathematical Modeling in Smart Cities (pp. 76-95). IGI Global.

[2]Hameed, A., Violos, J., & Leivadeas, A. (2022). A deep learning approach for IoT traffic multi-classification in a smart-city scenario. IEEE Access, 10, 21193-21210.

[3]Ramana, K., Srivastava, G., Kumar, M. R., Gadekallu, T. R., Lin, J. C. W., Alazab, M., & Iwendi, C. (2023). A Vision Transformer Approach for Traffic Congestion Prediction in Urban Areas. IEEE Transactions on Intelligent Transportation Systems.

[4]Chen, G., & Zhang, J. (2022). Applying Artificial Intelligence and Deep Belief Network to predict traffic congestion evacuation performance in smart cities. Applied Soft Computing, 121, 108692.

[5]Khan, N. U., Shah, M. A., Maple, C., Ahmed, E., & Asghar, N. (2022). Traffic flow prediction: an intelligent scheme for forecasting traffic flow using air pollution data in smart cities with bagging ensemble. Sustainability, 14(7), 4164.

[6]AlZoman, R. M., & Alenazi, M. J. (2021). A comparative study of traffic classification techniques for smart city networks. Sensors, 21(14), 4677.

[7]Zhou, S., Wei, C., Song, C., Pan, X., Chang, W., & Yang, L. (2022). Short-term traffic flow prediction of the smart city using 5G internet of vehicles based on edge computing. IEEE Transactions on Intelligent Transportation Systems.

[8]Gobezie, A., & Fufa, M. S. (2020). Machine learning and deep learning models for traffic flow prediction: A survey.

[9]Razali, N. A. M., Shamsaimon, N., Ishak, K. K., Ramli, S., Amran, M. F. M., & Sukardi, S. (2021). Gap, techniques and evaluation: traffic flow prediction using machine learning and deep learning. Journal of Big Data, 8(1), 1-25.

[10]Navarro-Espinoza, A., López-Bonilla, O. R., García-Guerrero, E. E., Tlelo-Cuautle, E., López-Mancilla, D., Hernández-Mejía, C., & Inzunza-González, E. (2022). Traffic flow prediction for smart traffic lights using machine learning algorithms. Technologies, 10(1), 5.

[11]Qi, T., Chen, L., Li, G., Li, Y., & Wang, C. (2023). FedAGCN: A traffic flow prediction framework based on federated learning and Asynchronous Graph Convolutional Network. Applied Soft Computing, 138, 110175.

[12]Djenouri, Y., Belhadi, A., Srivastava, G., & Lin, J. C. W. (2023). Hybrid graph convolution neural network and branch-and-bound optimization for traffic flow forecasting. Future Generation Computer Systems, 139, 100-108.

[13]Saleem, M., Abbas, S., Ghazal, T. M., Khan, M. A., Sahawneh, N., & Ahmad, M. (2022). Smart cities: Fusion-based intelligent traffic congestion control system for vehicular networks using machine learning techniques. Egyptian Informatics Journal, 23(3), 417-426.

[14]Dai, F., Huang, P., Mo, Q., Xu, X., Bilal, M., & Song, H. (2022). ST-InNet: Deep Spatio-Temporal Inception Networks for Traffic Flow Prediction in Smart Cities. IEEE Transactions on Intelligent Transportation Systems, 23(10), 19782-19794.

[15]Hassan, M., Kanwal, A., Jarrah, M., Pradhan, M., Hussain, A., & Mago, B. (2022, February). Smart City Intelligent Traffic Control for Connected Road Junction Congestion Awareness with Deep Extreme Learning Machine. In 2022 International Conference on Business Analytics for Technology and Security (ICBATS) (pp. 1-4). IEEE.

[16]Vijayalakshmi, B., Ramar, K., Jhanjhi, N. Z., Verma, S., Kaliappan, M., Vijayalakshmi, K., ... & Ghosh, U. (2021). An attention-based deep learning model for traffic flow prediction using spatiotemporal features towards sustainable smart city. International Journal of Communication Systems, 34(3), e4609.

[17]Joseph, L. L., Goel, P., Jain, A., Rajyalakshmi, K., Gulati, K., & Singh, P. (2021, October). A novel hybrid deep learning algorithm for smart city traffic congestion predictions. In 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC) (pp. 561-565). IEEE

[18]Awan, F. M., Minerva, R., & Crespi, N. (2021). Using noise pollution data for traffic prediction in smart cities: experiments based on LSTM recurrent neural networks. IEEE Sensors Journal, 21(18), 20722-20729.

[19]Qaisar, S. M., Khan, S. I., Srinivasan, K., & Krichen, M. (2023). Arrhythmia classification using multirate processing metaheuristic optimization and variational mode decomposition. Journal of King Saud University-Computer and Information Sciences, 35(1), 26-37.

[20]Henry, A., Gautam, S., Khanna, S., Rabie, K., Shongwe, T., Bhattacharya, P., Sharma, B., & Chowdhury, S. (2023). Composition of Hybrid Deep Learning Model and Feature Optimization for Intrusion Detection System. Sensors, 23(2), 890.

[21]Wang, H., Zhang, C., & Wu, H. (2023). Shear Capacity Prediction Model of Deep Beam Based on New Hybrid Intelligent Algorithm. Buildings, 13(6), 1395.
[22]https://www.kaggle.com/datasets/arashnic/road-trafic-dataset?select=region_traffic.csv