INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

TCC-HDL: A Hybrid Deep Learning Based Traffic Congestion Control System for VANET

VM Niaz Ahamed[1]*, **Arul Prakash**[2], **Mohamed Ziyath**[1]

1 Assistant Professor, Department of CSE, B.S Abdur Rahman Crescent Institute of Science and Technology, India
2 Assistant Professor, Department of CSE, Sathyabama Institute of science and Technology, India

OPEN ACCESS

Received: 31-05-2023

Accepted: 17-07-2023

Published: 29-08-2023

Citation: Ahamed VMN, Prakash A, Ziyath M (2023) TCC-HDL: A Hybrid Deep Learning Based Traffic Congestion Control System for VANET. Indian Journal of Science and Technology 16(32): 2548-2559. https://doi.org/10.17485/IJST/v16i32.1319

* **Corresponding author**.

msajce.niazahamed@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: This Study is centered on developing suitable method to reduce road accidents and improve individual traffic management as a part of smart cities development. **Methods:** A new hybrid deep learning-based model which uses a hybrid deep learning technique (TCC-HDL) is proposed to collect data on traffic patterns and send vehicles along the most efficient routes. The data are collected from kaggle about 8,000 roadside of 12-hour manual counts. From the extracted data, traffic congestion is predicted by new hybrid deep learning approach such as Recurrent capsule networks (CapsRNN), Fuzzy Interface System (FIS) and Optimized Bi-LSTM (O-Bidirectional Long short Memory). The proposed model TCC-HDL has been analyzed in terms of Accuracy, Precision, F-Measure and Recall with the standard algorithms like Bi-LSTM, CapsRNN, GRU, and LSTM. The information comes from the Highway Traffic Crash Dataset. Statistical features, higher-order statistical features, correlation-based features, and database features are used to extract information from the collected data. **Findings:** The work achieved 0.0102 to 0.1043% improvement in terms of accuracy, 0.0088% to 0.2133% of Precision, 0.039% to 0.2364% of Recall and 0.0056% to 0.083% of F-Measure. **Novelty:** New hybrid deep learning approach for predicting the situation of heavy traffic CapsRNN algorithm which has the better action recogoization and Bi-LSTM is the long term prediction of data which optimized using RSOA can fused together and it is fed as input to Fuzzy Interface System (FIS).

Keywords: Hybrid Deep Learning; Traffic Congestion Control System; VANET; Database Features

1 Introduction

Data shows 3500 automobile accidents each day worldwide. According to the UN, India has the most road fatalities. The drivers' lack of information contributed to this tragedy [1]. Road traffic accidents are more common at crossroads [2]. Traffic congestion happens when a metropolitan road's capacity is exceeded [3] Smart city congestion affects air pollution, fuel consumption, traffic violations, noise pollution, accidents, and wasted time. Traffic often waits for other routes to clear, delaying emergency vehicles. Fixed signal timings worsen congestion when significant traffic flows in one direction[4]. Intelligent transportation has grown with the smart town. Smart villages also control traffic[5]. ITS (Intelligent Transport technologies) was created in response to traffic congestion and poor wireless communication technologies for traffic management[6]. ITSs are becoming more popular as car traffic causes roadblocks, bottlenecks, and accidents. ITSs integrate ICTs with users, vehicles, and transportation networks to improve network management and safety. ITSs provide real-time weather, traffic regulation, and effective planning[7]. VANET (Vehicular Ad Hoc Networks) can achieve the key ITS aims of traffic congestion restraint, accident prevention, and optimal infrastructure utilization[8]. Control frameworks and traffic administration increasingly use VANETs[9]. VANETs enable vehicle-network frame communication, helping smart urban communities reduce traffic.Access VANETs in smart city districts reduce obstruction, accidents, malfeasance, and cost overhead. VANETs allow cars and displacement control constructions to communicate in structure-based and remote networks without context[10,11]. VANETs prioritise security, entertainment, and environmental protection[12]. VANETs communicate data across vehicle nodes without infrastructure or central management[13]. These networks optimise data communication protocols and wireless communication transmission efficiency to improve traffic monitoring and network efficiency[14].

The objective of the paper is:

- To establish a new hybrid deep learning approach for predicting the situation of heavy traffic.
- To select the Optimized Bi-LSTM (O-Bidirectional Long short Memory)to forecast traffic congestion.
- To enhance the traffic detection accuracy, the weight of Bi-LSTM is tuned using the new hybrid optimization model.

The remainder section is ordered as: Section 2 of the paper examine about the existing works that has been accomplished. Section 3 portrays information on the proposed methodology for Traffic Congestion Control System For VANET. The results gained with the proposed model is discussed thoroughly in Section 4. In Section 5, this paper is concluded.

1.1 Litrature Review

In 2018, Jain et al.[15] have suggested city lane and cross-sectional simulationto accomplish the goal of dominant vehicle mobility. It displays actual communication between cars and between cars and traffic infrastructure. Network Simulator 2 was used to conduct the experiment. 3 units all needed to be modelled for the implementation. In the simulation, factors like traffic volumeand packet loss were led intoaccount. These variables guarantee effective signal-to-signal communication. As the vehicles would be not reported to access the intersection area and given data pertaining to other vehicles, this improves traffic control and road safety.

In 2019, Liu et al.[16] have presented a thorough analysis of VANET congestion control strategies. Withal determined the pertinent Measurable variables and indicators of performance that could use to assess these strategies, and evaluated each strategy based on a variety of factors, including the nature of traffic, whichever it was energetic or sensitive, and the method for reducing blockage.

In 2019, Mohanty et al.[17] have created new model using the SUMO simulator for simulating a realistic traffic scheme in a big city is a major problem. In a city, there was many factors that contribute to traffic congestion, including the population's rapid growth, the number of four-wheelers, the condition of the roads, the lack of a physical plan to guide development, and the construction of more lanes, overpasses, underpasses, and overbridges at intersections.

In 2018, Sharma et al.[18] have created and validated ESPM framework to forecast the likelihood that an accident wasoccurred on an Indian four-lane highway. To foresee an emergency situation in advance was the main goal of ESPM. So, it works to reduce the number of fatalities and injuryinduced by traffic collision. Three stages—reporting, monitoring, and prediction—was involved in ESPM's emergency situation prediction process.

In 2017, Ravikumar et al.[19] have enhanced congestion control by using heuristic techniques to decrease traffic announcement channels while taking into account the reliability of submission supplies in VANETs. The simulation results have showed that meta-heuristic techniques significantly outperform other blocking control procedures in VANETs.

In 2020, Abdelatif et al.[20]have suggested VANET-Cloud layer to improve the interconnected system operation and manage traffic during busy periods. The proposed layer incorporates the advantages of the CSN for collecting the information on traffic and the cloud-based infrastructure for delivering automatic and on-demand cloud services for traffic management. The information content was propagated using a fuzzy aggregation process by traffic services using the mechanism for exchanging data. As part of the assessment phase, simulation conclusions showed how the carried out VANET-Cloud layer could significantly boost network performance and traffic safety when compared to more recent efforts.

In 2017, Mallah et al.[21] have suggested a context for the up to date distributed classification of gridlock into its constituent parts using VANET On a miscellaneous urban road network. Then offered models that were trained on artificial data extrapolated from a real-world case study of Cologne and built on knowledge of the structural and temporary causality estimates. By creating efficient strategies for congestion mitigation while being aware of the underlying causes of congestion, this framework could help transportation agencies reduce urban congestion.

In 2020, Choe et al.[22] have suggested a cooperative RL-powered intelligent channel access algorithm in which vehicles fully decentralise channel admittance coordination. In order to improve the V2V safety broadcast in congested, infrastructure-free VANETs, also took into consideration an appropriate interaction scheme between vehicles. Agreeing to different levels of traffic congestion, presented evaluation results from in-depth simulations. Additionally, the algorithm meets both the short- and long-run communication fairness requirements as well as the wait time essential of VANET secure application.

In 2018, Ullah et al.[23] have presented urgency message dispersalschedules that was based on VANET situation of preventing congestion and the use of vehicular Fog computing. A VANET architecture that usedFoG assistance was investigated in order to competentlydeal with message congestion scenarios. Also outlined a taxonomy of strategies for preventing message bottleneck. To highlight the favors and shortcomings of different congestion avoidance strategies, included a comparison discussion.

In 2021, Kothai et al.[24] have suggested a new hybrid BLSTME and CNN model that combined the robust CNN features with BLSTME to deal with the flexible actions of the vehicle and anticipate traffic congestion on roads successfully. CNN removes the features from the traffic representations, and the suggested BLSTME coaches and improves the weak classifiers to predict congestion. The suggested model was created with the help of the TensorFlow Python libraries and it was put to the test in a real-world traffic simulation using SUMO and OMNeT++. The model was evaluated using the performance metrics recall, precision, and likely prediction accuracy after conducting extensive experimentation. Table 1 shows the research gaps by various authors reviews about Traffic Congestion Control System for VANET.

1.2 Research Gap

Table 1. Various Authors reviews in research gaps

Author	Aim	Research Gaps
Ngo et al.[2]	enhancing the safety of intersections on roads	Minimum waiting time per vehicle performance metric is not improved using a genetic algorithmic approach.
Zheng et al.[5]	to install the bare minimum of RSUs necessary to distinguish and cover all rush-hour traffic.	Vehicle flow's spatiotemporal characteristics are not examined.
D'Andrea et al.[7]	real-time GPS monitoring for traffic jams and incidents	A dynamic routing service is not integrated with the system.
Elhoseny et al.[13]	Utilising a Clustering Model for Energy Efficient Optimal Routing in VANET Communication	low cost, energy efficiency remains unchanged.
Qureshi et al.[14]	A dynamic congestion monitoring programme for vehicular ad hoc network safety applications	Techniques for probabilistic traffic prediction are not investigated.
Jobaeret al.[8]	VANET-Based UAV-helped Hybrid Scheme for safety on city roads	Amount of RSU is not reduced.
Choe et al.[22]	Congested vehicular networks require a robust channel access method that uses cooperative reinforcement learning.	Performance is not improved.
Kothaiet al.[24]	Prediction of Severe Traffic Congestion in Smart Cities	• No larger datasets are available. • Prediction accuracy is not improved.

2 Methodology

2.1 Training Phase

Step1: The data is collected from: https://www.kaggle.com/datasets/arashnic/road-trafic-dataset?select=region_traffic.csv

Step 2: Then, the data features like statistical features, higher order statistical features, correlation-based features and database features is extracted from the collected data.

Step 3: The traffic congestion is predicted using the new hybrid deep learning approach (proposed) that includes the Recurrent capsule networks (CapsRNN)[25], Fuzzy Interface System (FIS) and Optimized Bi-LSTM (O-Bidirectional Long short Memory) (proposed). The RSOAis trained using the extracted statistical features. In addition, to enhance the traffic detection accuracy, the weight of Bi-LSTM is tuned using the new hybrid optimization model (RSOA). It is the association of the standard SWOA and RO. Outcome from Recurrent capsule networks (CapsRNN) and Optimized Bi-LSTM is fused together, and it is fed as input to Fuzzy Interface System (FIS), which makes the final detection regarding the network congestion. The data acquired from FIS is stored in the cloud. Figure 1 demonstrates the general layout of the proposed work.

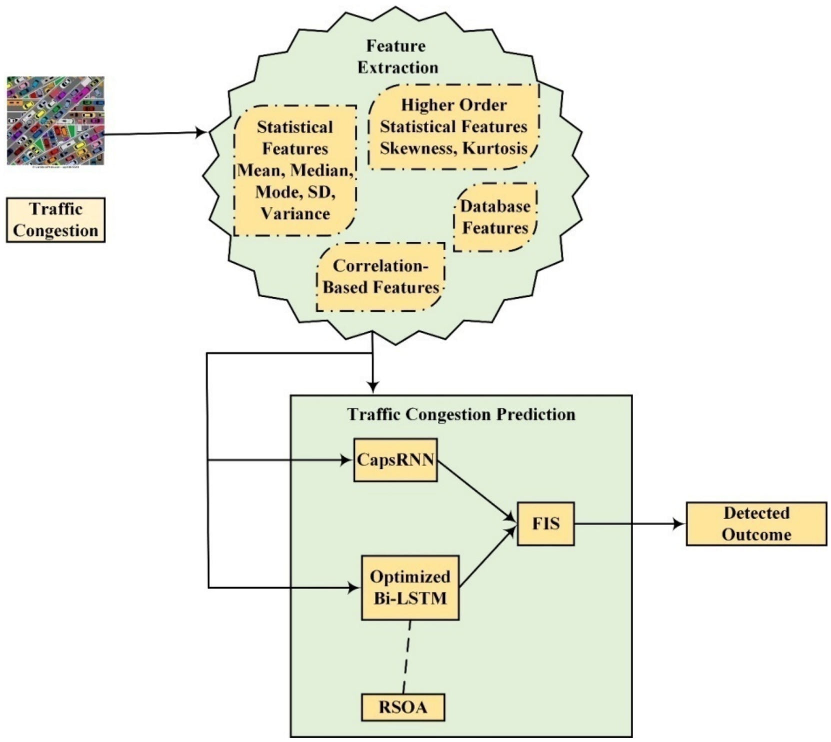

Fig 1. Overall Framework of the proposed Methodology

2.2 Data Collection

In this research work, information is collected from the Traffic Road Dataset. For each junction-to-junction link on the motorway and 'A' road network, as well as for few tiny roads in Great Britain, road traffic open data provides street-level information. Vehicle miles, which combine the quantity of vehicles present on the road and the distance they travel, are the most common units of measurement for annual statistics. Details from about 8,000 roadside 12-hour manual counts, continuously updated traffic counters knowledge, and information on road lengths are used to create annual traffic statistics. Every 10 years or so, the road traffic information group conducts asmall route traffic benchmarking exercise with the goal of increasing the precision of traffic projections for small paths. The exercise's findings from 2018 to 2019 have been made public, and the minor road has been modified.

2.3 Feature Extraction

In this research work, the collected raw data are extracted using Statistical Features like mean, median, mode, standard deviation, variance also higher order statistical features, correlation-based features and database features.

2.3.1 Statistical Features

• **Mean**-The "mean" is the standard "average" one might arrive at by adding up all the numerals then dividing by the sum. Mathematical expression is shown in Eq. (1).

$$\bar{v} = \frac{v}{no} \tag{1}$$

Where, $\bar{v}$ is the means of set of $v$standards.

$\sum v$ is the entirety of all the $v$estimates

no is the quantity of v values.

• **Median**-The core value in a numerical list is termed "median."

• **Mode:**The Worth that is widely used "mode". There is no mode for the list if no number on the repeats in the list.

• **Standard Deviation**-The standard deviation is assessing of how widely apart or differently a set of amounts. While a high standard deviation indicates that the values are distributed across a broad array, a low standard deviation suggests that the values have a tendency to be nearby to the set mean. The sigmaσ is most deployed a lot in mathematics texts and equations to represent standard deviation, which also be referred to as SD. The mathematical expression is shown in Eq. (2).

$$\sigma = \sqrt{\frac{\sum (e-\mu)^2}{S}} \tag{2}$$

σ is the population Standard Deviation, $e$every population-based value, S is area and, μ is the population mean

• **Variance**- The statistical measurement of the variation in numbers within a data set is known as variance. In more detail, variance assesses how far apart each number in the set is from the meanfrom each other. This icon describe variation usually: 2 . It is utilised by analysts and traders to evaluate market volatility and privacy. The mathematical model is shown in Eq. (3).

$$\sigma^2 = \frac{\sum_{i=1}^{n} (a-\mu)^2}{n} \tag{3}$$

a are the elements of the available data.

2.3.2 Higher Order Statistical Features (skewness, kurtosis)

• **Skewness**- The difference between the mean and median is multiplied by three, and the standard deviation is divided by this result to determine the skewness.The Eq. (4) defines skewness (γ_1)

$$(\gamma_1) = W\left[\left(\frac{b-\bar{b}}{\sigma}\right)^3\right] \tag{4}$$

$\bar{b}$ is the sample mean.

• **Kurtosis**- The "tailedness" of actual values random variable's probability distribution is measured by kurtosis. Kurtosis, like skewness, describes specific feature of a probability distribution. For a theoretical distribution, there are various ways to measure kurtosis, and corresponding methods of estimating it from a sample of a population also exist. Kurtosis measurements can be interpreted in various ways. The fourth standardised moment, kurtosis (β_2) is described in Eq. (5).

$$\beta_2 = W\left[\left(\frac{b\_\bar{b}}{\sigma}\right)^4\right] \tag{5}$$

$\bar{b}$ is the sample mean.

• **Correlation**- A statistical analysis known as correlation expresses how linearly related two variables are to each other. It's a widely used method for describing basic connections not expressing the relationship of cause and effect.

• **Database features**

• **region_id :** This refers to an identifier for a particular region or area. It could be a numerical or alphanumeric code used to uniquely identify a specific geographic region.

• **region_name :** This refers to the name of the region or area corresponding to the region_id. For example, "Northwest Region" or "London Boroughs."

• **road_category_id :** This refers to an identifier for a particular type or category of road. It could be a numerical or alphanumeric code used to uniquely identify a specific road category.

• **road_category_name :** This refers to the name of the road category corresponding to the road_category_id. For example, "Motorway" or "A Road."

• **road_category_description :** This refers to a description or definition of the road category, which may include information on the characteristics, speed limits, and usage restrictions of the roads in that category.

• **total_link_length_km :** This refers to the total length of road links (or segments) in kilometres within a particular region and road category.
• **total_link_length_miles :** This refers to the total length of road links (or segments) in miles within a particular region and road category.
• **pedal_cycles :** This refers to the number of pedal cycles (i.e., bicycles) that have travelled on the roads.
• **lgvs :** This refers to the number of light goods vehicles (e.g., vans) that have traveled on the roads in the region and road category.
• **two_wheeled_motor_vehicles :** This refers to the number of motor vehicles with two wheels (e.g., motorcycles, scooters) that have travelled on the roads.
• **cars_and_taxis :** This refers to the number of cars and taxis that have travelled on the roads pertaining to the region and roads.
• **buses_and_coaches :** This refers to the number of buses and coaches that have travelled on the roads in the region and road category.
• **all_hgvs :** This refers to the number of heavy goods vehicles (e.g., trucks, lorries) that have traveled on the roads in the geographical and road class.
• **all_motor_vehicles :** This relates to the overall number of all motor vehicles that have traveled on the roads in the location and road type.

2.4 Traffic congestion prediction

In this research work, the extracted features are predicted using the new hybrid deep learning approach (proposed) that includes the Recurrent capsule networks (CapsRNN), Fuzzy Interface System (FIS) and Optimized Bi-LSTM. Novel hybrid deep learning approach is trained using the extracted statistical features. In addition, to enhance the traffic detection accuracy, the weight of Bi-LSTM is tuned using RSOA. It is the combination of the standard SWOA and RO. The outcome from Recurrent capsule networks (CapsRNN) and Optimized Bi-LSTM is fused together, and it is fed as input to Fuzzy Interface System (FIS), which makes the final detection regarding the network congestion. The data acquired from FIS is stored in the cloud.

2.4.1 Recurrent capsule networks (CapsRNN)

The input layer, recurrent layer, and capsule layer are the three main parts that the network integrates.

2.4.1.1 Embedding layer. In a sentence related to coronary arteriography$X = (c]_1^S$, which is a sequence of S words, mapping independent language symbols is the fundamental step, together with the words and associated categories of entities, to vectors that are distributed. Lookup word embedding $c'_i \in R^{f_c}$from word embedding matrix $A_c$for every individual wordc_i, where $i \in (1,2,\ldots,S\}$ indicates $c_i$is the ith word in X, and f_c is a hyper-parameter indicating the size of word embedding. Withal look up entity type embedding $f'_i \in R^{f_f}$from entity type embedding matrix A_f for each type of the entity which $c_i$appertains to, where f_f is a hyper-parameter expressing the dimensionality of entity type embedding. Linking the initial embedding vector with the $c'_i$and f'_i as$d_i = c'_i \oplus f'_i$where $\oplus$is the concatenation operator.

2.4.1.2 Recurrent layer. The LSTM is an RNN variant that uses a gated memory cell to occupy extended-range dependencies in the data and has the capability to prevent the rapidly increasing issues that are present in conventional RNNs.

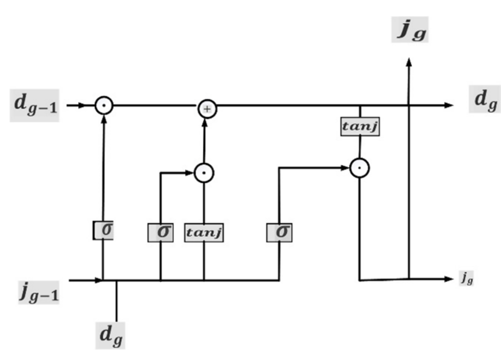

Fig 2. LSTM Cell Illustration

Figure 2 depicts the LSTM cell. For every place g, LSTM determine j_g with input d_g and earlier state $d_g - 1$, as:

$$i_g = \sigma\left(A_i d_g + U_i d_{g-1} + h_i\right) \tag{6}$$

$$f_g = \sigma\left(A_k d_g + U_k d_{g-1} + h_k\right) \tag{7}$$

$$\widetilde{l}_g = tanj\left(A_l d_g + U_l d_{g-1} + h_l\right) \tag{8}$$

$$l_g = k_g \odot l_{g-1} + i_g \odot \tilde{l}_g \tag{9}$$

$$m_g = \sigma\left(A_m d_g + U_m j_{g-1} + h_m\right) \tag{10}$$

$$d_g = m_g \odot \mathrm{tanj}\left(l_g\right) \tag{11}$$

where $j, i, k, m \in R^{f_j}$ are f_j-dimensional hidden state, input gate, forget gate and output gate, respectively; $A_i, A_k, A_l, A_m \in R^{f_j * f_d}, U_i, U_k, U_l, U_m \in R^{4 f_j * f_j}$ and $h_i,\ h_{k,} h_l,\ h_m \in R^{4 f_j}$ consist of the LSTM's settings; σ is the sigmoid function, and $\odot$designate element wise presentation.

Because of the present condition $j_g$also consider the prior state l_{g-1} and j_{g-1} , the final state j_{Si} can be considered to be an illustration of the entire segment B_i. The hidden state j_g of LSTM only takes information from past. Bidirectional LSTM, which take into account data from the past and future, is one remedy. Formally, network performs calculations both a left and a right and $\overrightarrow{j_{Si}}$illustrations of the last time step's sequence setting. The depiction of the entire segment B_i is created by concatenating them as $b = \overrightarrow{j_{Si}} \oplus \overleftarrow{j}_m$.

Five LSTMs were used to handle the five segments of the sentence because the two entities in the sentence can be divided into that many parts. Particularly, the context is less impact on relation classification and that there is a greater distance. However, Bi-LSTMs are used for the three middle categories left-to-right $\overrightarrow{LSTM}$ are used for the left-most segment, and right-to-left $\overleftarrow{LSTM}$ are used for the right-most segment. Therefore, the likelihood that the LSTM remember a context increases with how close it is to an entity.

2.4.1.3 Capsule layer.

The first application of the capsule layer is digit recognition. A capsule is a group of neurons whose activity vector illustrates the instantiation parameters of a particular kind of connection which uses only a neuron to represent the classification probability via a sigmoid function or a softmax function: The probability that the corresponding relation exists is represented by the length of the activity vector, and different vector orientations can designate various cases under the relation, giving the capsule a greater capacity for expression.In order to ensure that short vectors are lowered to almost zero length and long vectors are decreased to a length just underneath 1, which takes into account the fact that the length of a capsule is exploited as the chance of a relationship occurring.

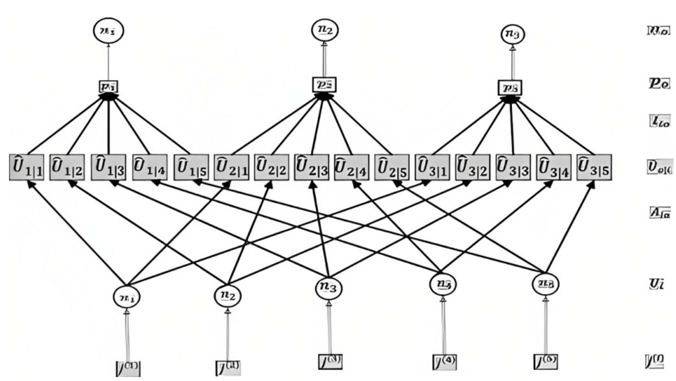

Fig 3. The layer of the capsule where squashing operations are indicated by double-line arrows

$$n_o = \frac{\|p_o\|^2}{1+\|p_o\|^2}\frac{p_o}{\|p_o\|} \tag{12}$$

Where n_o is the resultant vector produced by an input capsulep_o.

As shown in Figure 3, the combined input received by a capsulep_o is a summation of values "prediction vectors" $\widehat{U}_{o|i}$Generated by performing a multiplication operation on the output of capsules in the lower layerU_i of a capsule in the lower layer is multiplied by a matrix of weightsA_{io}.

$$\widehat{U}_{o|i} = A_{io}U_i \tag{13}$$

2.4.2 Fuzzy Interference System (FIS)

A fuzzy logic system is composed of a fuzzy inference system, which utilizes fuzzy set theory, IF-THEN rules, and fuzzy reasoning to determine the output that corresponds to crisp inputs.

2.4.2.1 FIS characteristics include.

- Obtain precise value from the action
- Uses the fuzzy membership function to transform the crisp value into a fuzzy value.
- Utilise the IF-THEN rules in the fuzzy rule base to produce fuzzy results.
- Apply some Methods used to convert fuzzy output into a precise/crisp value.

2.4.2.2 Blocks of Function in FIS.

- A detailed understanding of Fuzzy's construction is possible thanks to its five functional building blocks.
- The fuzzy IF-THEN rules make up the set of rules.
- The functions defining the membership of fuzzy sets employed in the fuzzy rules are defined in the database.
- The decision-making unit applies the fuzzy rules in its operations.
- Any crisp quantity can be transformed into a fuzzy quantity with the aid of the fuzzification interface unit.
- Any fuzzy quantity can be transformed into a crisp quantity using the defuzzification interface unit.

2.4.2.3 Operation of FIS.

- These steps can be used to breakdown the fuzzy inference system's overall operation.
- The various applications of the fuzzification techniques is supported by the fuzzification unit.
- When the crisp input is changed to a fuzzy input, a rule-based database's knowledge base is formed by assembling a collection of rules.
- The fuzzy input from the defuzzification unit is finally be transformed into a crisp output.

2.4.3 Optimized Bi-LSTM (O-Bidirectional Long short Memory)

A bidirectional RNN does the same thing as a basic RNN, but it also connects layers from the present time step to those from the past time step. Concatenated outputs from two hidden layers with opposing directions make up the layer's output in the current time step. This is an example of how knowledge of the past and future influences the present. Bidirectional LSTM (BiLSTM) is the network that results from replacing a basic RNN unit in a bidirectional RNN with an LSTM unit. The BiLSTM's hidden state, d_g, is then a concatenated matrix of the forward and backward hidden states,$\overrightarrow{d_g}$and $\overleftarrow{d_g}$respectively. The mathematical model is shown in Eq. (14).

$$d_g = \left[\overrightarrow{d_g}, \overleftarrow{d_g}\right] \tag{14}$$

A weight function is used in a Bi-LSTM to determine the relative weights of each input feature and hidden state at each time step.A group of learnable parameters that are optimised during training define the weight function. These parameters specify the contribution of each input feature and hidden state to the network's output.

Weight function of Bi-LSTM optimized via RSOA

Step 1: Initialization: Generate the initial Population of Search Agent randomly within search space.

Step 2: Fitness Computation:

Evaluate the fitness of each search agent using Eq. (15).

$$Fit = min(\varepsilon) \tag{15}$$

Where, ε is the error.

Step 3: Neighbourhood formation: Build a neighbourhood structure among the search agents using the small-world optimization algorithm. Each Search Agent is connected to its K- Nearest Neighbour in the search space.

Step 4: Exploration Space

During the exploration space, the RO is used to explore the search space by simulating the movement of rays of light. The search agent moves randomly within their neighbourhoods in a Zigzag pattern to cover the search space thoroughly. This movement can be given as:

$$X_i^{t+1} = X_i^t + \frac{s}{cos\left(\theta_i^t\right)}$$

$$Y_i^{t+1} = Y_i^t + \frac{s}{sin\left(\theta_i^t\right)}$$

X_i^t, Y_i^t is the coordinates of ith search agent, t iteration number, s is step size (fixed as 0.5), θ_i is direction angle of ith search agent.

Step 5: Exploitation phase (Small-World optimization):

During the exploitation phase, the Small-World Optimization Algorithm is used to exploit the best search agents found during the exploration phase. The search agents advance the guidance of the best search agent within their neighbourhoods to converge towards the optimal solution.The movement of each search agent is expressed using the following mathematical equation:

$$x_i\{t+1\} = x_i t \backslash frac\{\backslash beta\ (x_i t - x_i t)\}\left\{d_{ij}^2\right\} \tag{16}$$

$$y_i\{t+1\} = y_i t \backslash frac\{\backslash beta\ (y_i t - y_i t)\}\left\{d_{ij}^2\right\} \tag{17}$$

x_i and y_i are the systematize of the i-th search agent, j is the index of the best search agent within the neighborhood of the i-th search agent, t is the iteration number, $\left\{d_{ij}^2\right\}$ is the Euclidean distance between the i-th and j-th search agents, and β is a control parameter that determines the step size.

Update the fitness and neighborhood structure: Evaluate the fitness of each search agent using the objective function and update the neighbourhood structure using the Small-World Optimization Algorithm.

Termination: Terminate the algorithm if a condition for stopping is fulfilled (e.g., supreme number of iterations is accomplished, the fitness of the best search agent does not improve over a certain number of iterations, etc.). Otherwise, return to step 4.

3 Results and Discussion

The suggested model has been executed using MATLAB. The intended model has been analysed in terms of Accuracy, Precision, F-Measure and Recall.

3.1 Performance Metrics

Accuracy, F-measure, precision, Recall is used as comparison metrics for performance.

3.1.1 Accuracy

Accuracy is the degree of closeness between a measurement and its true value. A classification model's performance can be assessed using the metric of accuracy. According to Eq. (18), accuracy is the prediction model's percentage for both the number of values that were correctly predicted and the overall number of predicted values.

$$Accuracy = \frac{Detected\ Result}{Total\ no.\ of\ iteration} \tag{18}$$

3.1.2 Precision

Precision refers to the amount of information that is conveyed by a number in terms of its digits. The model's precision measures how accurate it is, or how many of the positive predictions have turned out true. The mathematical expression is shown in Eq. (19)

$$precision = \frac{true\ positive}{true\ positive + false\ negative} \tag{19}$$

3.1.3 F-Measure

The F-measure can be viewed as a compromise between recall and precision. F1 Score seeks to strike a balance in recall and precision. The mathematical expression of F-Measure is shown in Eq. (20)

$$F1\ score = \frac{2 * Precision * Recall}{recall + precision} \tag{20}$$

3.1.4 Recall

The ability of a model to find all the relevant cases within a data set. Recall measures how many Actual Positives the model identified properly. The mathematical expression is shown in Eq. (21)

$$recall = \frac{true\ positive}{true\ positive + false\ negative} \tag{21}$$

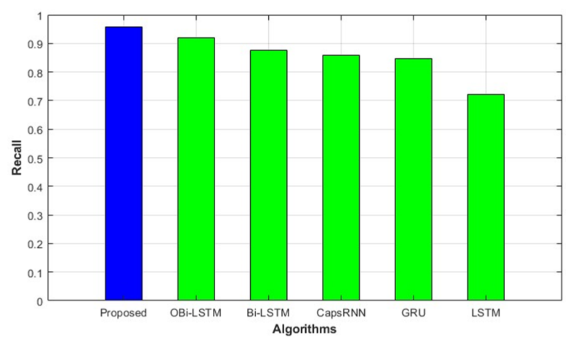

Fig 4. Recall of production and existing methods

Figure 4 illustrates the visual depiction of Recall. Proposed model has fulfilled the highest recall value of 0.957469 among the models tested. The Obi-LSTM (Optimized Bi-LSTM) model has the second-highest recall value of 0.918457, while the Bi-LSTM, CapsRNN, GRU, and LSTM models have progressively lower recall values of 0.875216, 0.857971, 0.848321, and 0.721057, respectively.

Figure 5 depicts a visual representation of precision. Proposed model has accomplished the utmost precision value of 0.883621 among the models tested. The Obi-LSTM (Optimized Bi-LSTM) model has the second-highest precision value of 0.874803, while the Bi-LSTM, CapsRNN, GRU, and LSTM models have progressively lower precision values of 0.823885, 0.806034, 0.762467, and 0.670341, respectively.

Figure 6 shows the accuracy performance. Proposed model has finished the highest accuracy value of 0.974961 among the models tested. Obi-LSTM (Optimized Bi-LSTM) model has the second-highest accuracy value of 0.964777, while the Bi-LSTM, CapsRNN[25] , GRU, and LSTM models have progressively lower accuracy values of 0.952493, 0.934068, 0.922414, and 0.87069, respectively.

Figure 7 presents the performance of F-measure. Proposed model has achieved the highest F-measure value of 0.986089 among the models tested. The Obi-LSTM (Optimized Bi-LSTM) model has the second-highest F-measure value of 0.980432, while the Bi-LSTM[24] , CapsRNN[25] , GRU[14] , and LSTM models have progressively lower F-measure values of 0.973607, 0.963371, 0.94181, and 0.903017, respectively.The work achieved 0.0102 to 0.1043% improvement in terms of accuracy, 0.0088% to 0.2133% of Precision, 0.039% to 0.2364% of Recall and 0.0056% to 0.083% of F-Measure.

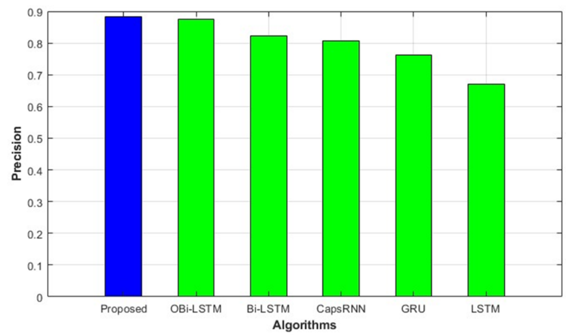

Fig 5. Precision of Performance and existing methods

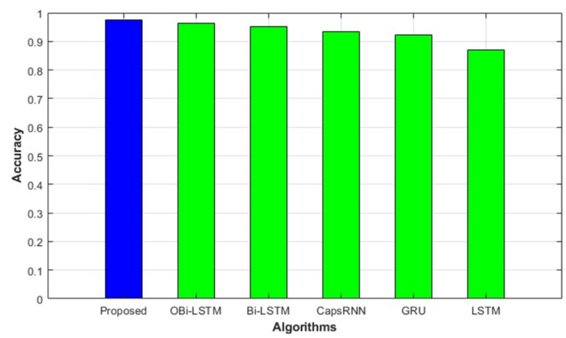

Fig 6. Accuracy of proposed and existing methods

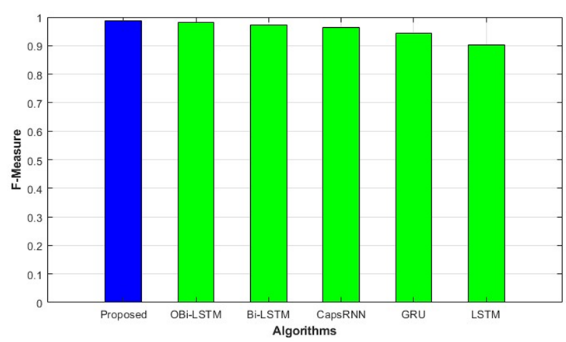

Fig 7. F-measure of proposed and existing methods

4 Conclusion

A new hybrid deep learning-based model is presented to accurately predict traffic congestion on roads. To reduce traffic congestion in connected cities, this research proposed a new traffic congestion control system using hybrid deep learning approach techniques that gather traffic data on available routes. Training and testing are the two main phases of the suggested model. The data are collected from kaggele about 8,000 roadside 12-hour manual counts. The Road Traffic Dataset is where the information is found. The gathered data are extracted using database features, higher order statistical features, correlation-based features, statistical features. From the extracted data, a new hybrid deep learning approach using recurrent capsule networks, fuzzy interface systems, and optimised bi-LSTM is used to predict traffic congestion. The newly developed hybrid deep learning approach (RSOA) is trained using the statistical features that were extracted. Additionally, the weight of the Bi-LSTM is adjusted using the new hybrid optimisation model to increase the accuracy of traffic detection. The proposed model TCC-HDL has been analysed in terms of Accuracy, Precision, F-Measure and Recall with the standard.

References

1) Subramaniam M, Rambabu C, Chandrasekaran G, Kumar NS. A Traffic Density-Based Congestion Control Method for VANETs. *Wireless Communications and Mobile Computing*. 2022;2022:1–14. Available from: https://doi.org/10.1155/2022/7551535.
2) Ngo TT, Huynh-The T, Kim DS. A novel VANETs-based traffic light scheduling scheme for greener planet and safer road intersections. *IEEE Access*. 2019;7:22175–22185. Available from: https://doi.org/10.1109/ACCESS.2019.2891250.
3) Jindal V, Bedi P. An improved hybrid ant particle optimization (IHAPO) algorithm for reducing travel time in VANETs. *Applied Soft Computing*. 2018;64:526–535. Available from: https://doi.org/10.1016/j.asoc.2017.12.038.
4) Ata A, Khan MA, Abbas S, Khan MS, Ahmad G. Adaptive IoT Empowered Smart Road Traffic Congestion Control System Using Supervised Machine Learning Algorithm. *The Computer Journal*. 2021;64(11):1672–1679. Available from: https://doi.org/10.1016/j.jksuci.2018.10.011.
5) Zheng H, Chang W, Wu J. Traffic flow monitoring systems in smart cities: Coverage and distinguishability among vehicles. *Journal of Parallel and Distributed Computing*. 2019;127:224–237. Available from: https://doi.org/10.1016/j.jpdc.2018.07.008.
6) Khatri S, Vachhani H, Shah S, Bhatia J, Chaturvedi M, Tanwar S, et al. Machine learning models and techniques for VANET based traffic management: Implementation issues and challenges. *Peer-to-Peer Networking and Applications*. 2021;14(3):1778–1805.
7) D'andrea E, Marcelloni F. Detection of traffic congestion and incidents from GPS trace analysis. *Expert Systems with Applications*. 2017;73:43–56. Available from: https://doi.org/10.1016/j.eswa.2016.12.018.
8) Jobaer S, Zhang Y, Hussain MAI, Ahmed F. UAV-Assisted Hybrid Scheme for Urban Road Safety Based on VANETs. *Electronics*. 2020;9(9):1499. Available from: https://doi.org/10.3390/electronics9091499.
9) Rashid MM, Datta P. Performance Analysis of Vehicular Ad Hoc Network (VANET) Considering Different Scenarios of a City. *International Journal of Computer Applications*. 2017;(10):162. Available from: https://doi.org/10.5120/ijca2017913329.
10) Lakshmanaprabu SK, Shankar K, Rani SS, Abdulhay E, Arunkumar N, Ramirez G, et al. An effect of big data technology with ant colony optimization based routing in vehicular ad hoc networks: Towards smart cities. *Journal of Cleaner Production*. 2019;217:584–593. Available from: https://doi.org/10.1016/j.jclepro.2019.01.115.
11) Rath M, Pati B, Pattanayak BK. Mobile agent-based improved traffic control system in VANET. 2019. Available from: https://link.springer.com/chapter/10.1007/978-981-10-8797-4_28.
12) Rizwan A, Karras DA, Dighriri M, Kumar J, Dixit E, Jalali A, et al. Simulation of IoT-based Vehicular Ad Hoc Networks (VANETs) for Smart Traffic Management Systems. *Wireless Communications and Mobile Computing*. 2022;2022:1–11. Available from: https://doi.org/10.1155/2022/3378558.
13) Elhoseny M, Shankar K. Energy efficient optimal routing for communication in VANETs via clustering model. 2020. Available from: https://doi.org/10.1007/978-3-030-22773-9.
14) Qureshi KN, Abdullah AH, Kaiwartya O, Iqbal S, Butt RA, Bashir F. A Dynamic Congestion Control Scheme for safety applications in vehicular ad hoc networks. *Computers & Electrical Engineering*. 2018;72:774–788. Available from: https://doi.org/10.1016/j.compeleceng.2017.12.015.
15) Jain R. A congestion control system based on VANET for small length roads. 2018. Available from: https://doi.org/10.48550/arXiv.1801.06448.
16) Liu X, Jaekel A. Congestion Control in V2V Safety Communication: Problem, Analysis, Approaches. *Electronics*. 2019;8(5):540. Available from: https://doi.org/10.3390/electronics8050540.
17) Mohanty A, Mahapatra S, Bhanja U. Traffic congestion detection in a city using clustering techniques in VANETs. *Indonesian Journal of Electrical Engineering and Computer Science*. 2019;13(3):884. Available from: https://ijeecs.iaescore.com/index.php/IJEECS/article/view/13156.
18) Sharma S, Pandey R. Accident detection, avoidance and prevention using intelligent transportation system. *International Journal of Computer Applications*. 2018;182(7):10–12. Available from: https://www.ijcaonline.org/archives/volume182/number7/sharma-2018-ijca-917639.pdf.
19) Ravikumar K, Vishvaroobi T. Congestion control in vehicular ad hoc networks (VANET) using meta-heuristic techniques. *International Journal of Computer Science Trends and Technology*. 2017;5(4):66–72. Available from: http://www.ijcstjournal.org/volume-5/issue-4/IJCST-V5I4P12.pdf.
20) Abdelatif S, Derdour M, Ghoualmi-Zine N, Marzak B. VANET: A novel service for predicting and disseminating vehicle traffic information. *International Journal of Communication Systems*. 2020;33(6):e4288. Available from: https://doi.org/10.1002/dac.4288.
21) Mallah RA, Quintero A, Farooq B. Distributed Classification of Urban Congestion Using VANET. *IEEE Transactions on Intelligent Transportation Systems*. 2017;18(9):2435–2442. Available from: https://doi.org/10.48550/arXiv.1904.12685.
22) Choe C, Ahn J, Choi J, Park D, Kim M, Ahn S. A Robust Channel Access Using Cooperative Reinforcement Learning for Congested Vehicular Networks. *IEEE Access*. 2020;8:135540–135557. Available from: https://doi.org/10.1109/ACCESS.2020.3011568.
23) Ullah A, Yaqoob S, Imran M, Ning H. Emergency Message Dissemination Schemes Based on Congestion Avoidance in VANET and Vehicular FoG Computing. *IEEE Access*. 2019;7:1570–1585. Available from: https://doi.org/10.1109/ACCESS.2018.2887075.
24) Kothai G, Poovammal E, Dhiman G, Ramana K, Sharma A, Alzain MA, et al. A New Hybrid Deep Learning Algorithm for Prediction of Wide Traffic Congestion in Smart Cities. *Wireless Communications and Mobile Computing*. 2021;2021:1–13. Available from: https://doi.org/10.1155/2021/5583874.
25) Tang J, Yang L, Liu S, Liu W, Wang M, Wang C, et al. Caps-LSTM: A Novel Hierarchical Encrypted VPN Network Traffic Identification Using CapsNet and LSTM. *Science of Cyber Security*. 2021;p. 139–153. Available from: https://link.springer.com/chapter/10.1007/978-3-030-89137-4_10.