Blockchain-Enabled Online Traffic Congestion Duration Prediction in Cognitive Internet of Vehicles

Huigang Chang, Yiming Liu, *Member, IEEE*, and Zhengguo Sheng, *Senior Member, IEEE*

***Abstract*—The real-time intelligent perception and prediction of traffic situation can assist connected automated vehicles (CAVs) in route planning and reduce traffic congestion in cognitive internet of vehicles (CIoVs). The traditional centralized offline training and deployment generally fail to adapt to the dynamic traffic environment and incur significant communication overheads. Blockchain technology has attracted great attention in the information storage of vehicular networks for its advantages in decentralization, transparency, traceability, and tamper-proof capability. However, due to the bottlenecks such as high computational cost and unable to prevent malicious attacks, current blockchains are incapable actuate on efficient online traffic situational cognition and prediction for CIoVs. Motivated by this, we propose a consortium blockchain-enabled cognitive segments sharing framework for online multi-step congestion duration prediction. We design a cognitive model of traffic situation based on anomaly detection and filtering mechanism to guarantee the accuracy of the cognitive segments before being packaged into the block. Furthermore, to improve the consensus efficiency and resist malicious attacks, we consider a credit evaluation mechanism and proposed a credit-based delegated byzantine fault tolerance (CDBFT) consensus algorithm. Last, we propose an online multi-step prediction algorithm based on long short-term memory (LSTM) to predict future traffic congestion duration. Experiment results based on a real dataset demonstrate that the proposed algorithms achieve shorter consensus delay and higher predictive accuracy than existing algorithms while effectively resisting malicious attacks.**

***Index Terms*—Connected automated vehicles, cognitive internet of vehicles, consortium blockchain, consensus algorithm, congestion duration prediction.**

I. Introduction

THE cognitive internet of vehicles (CIoVs) scenario produces amounts of data that build the basis for data-driven applications assisted by the rapid development of 5G V2X and artificial intelligence (AI) [1], [2]. According to the report [3], it is known that each connected automated vehicle (CAV) generates and consumes about 40 TB of sensory data every eight hours while driving. The precise cognition and prediction of traffic congestion duration by analyzing the sensory data can guide dynamic route planning and traffic management, effectively alleviating road congestion, and reducing travel time [4]. Therefore, secure and efficient traffic situational cognitive and sharing is fundamental to improving the capabilities of the cooperative intelligent transportation system (CITS) [5].

CIoVs provide a new paradigm to perceive the traffic situation through communication technologies and intelligent algorithms deployed on the cognitive engines (CEs) [6], [7]. However, the centralized learning architecture requires uploading large amounts of raw sensory data to a remote central server for processing and analysis, which tends to induce huge communication overload as well as data security and privacy issues [8]. Moreover, traditional centralized data storage and learning manner cannot withstand single point of failure (SPoF) and malicious attacks, failing to establish secure and efficient decentralized vehicular networks for cognitive data management and sharing in CIoVs [9].

Recently, blockchain has attracted great attention and research works in vehicular networks to facilitate establishing a decentralized and secure CITS ecosystem [10]–[12]. Leveraging blockchain technology can build a distributed vehicular self-organizing network to serve critical information sharing [13]. Furthermore, blockchain provides the new technical foundations for decentralized learning to support automated driving systems (ADS) [14]. In [15], the authors propose a blockchain-based crowdsourcing model and analyze historical data with neural networks for traffic congestion estimation. Generally, the highly dynamic and latency-sensitive vehicular networks require a faster and more secure consensus process [16]. However, due to the high cost and unauthorized nature of the public blockchain, the conventional consensus mechanisms cannot be suitable for the large-scale CIoVs as well as unauthenticated nodes will launch attacks to prolong consensus latency and even consensus bias [17], [18].

On the other hand, the accuracy and reliability of cognitive data before packaging into the block is critical to the performance of online traffic perception and prediction. Current popular approaches estimate and forecast road congestion based on long-cycle regularities of historical data [19], [20]. The authors in [21] propose a convolutional neural network (CNN) based supervised congestion prediction method on a statistical analysis framework. To better evaluate traffic congestion criteria for prediction, the authors in [22] propose a traffic congestion prediction model based on a roadway grouping algorithm by combining traffic data mining and CNN. In [23], the authors construct the congestion matrix of regional traffic networks to predict future congestion at all locations of the road network. In [24], the authors propose a traffic congestion prediction strategy based on edge data collection and analysis.

H. Chang and Y. Liu are with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: changhuigang@bupt.edu.cn; liuyiming@bupt.edu.cn). *(Corresponding author: Yiming Liu.)*

Z. Sheng is with the School of Engineering and Informatics, University of Sussex, Brighton BN1 9RH, U.K. (e-mail: z.sheng@sussex.ac.uk).

However, the current works with high computational costs and lack online real-time traffic congestion duration perception and prediction, making it impractical to adapt to real-life traffic environments.

Motivated by these developments and challenges, we propose a consortium blockchain-enabled cognitive segments sharing framework for decentralized online traffic congestion duration prediction in CIoVs (BCIoVs). The sharing of cognitive segments and decentralized predictive models can effectively reduce communication overload and enhance the intelligence of CITS. In the proposed BCIoVs framework, We design the traffic situation cognitive model based on anomaly detection and filtering mechanism to overcome the interference of malicious data, which ensures the correctness of the cognitive segment to improve the predictive performance. To facilitate the secure and efficient sharing of cognitive segments, we propose a credit-based delegated byzantine fault tolerance (CDBFT) consensus algorithm to prevent malicious attacks and reduce consensus latency. Based on the cognitive segments of traffic efficiency derived from the cognitive model, we propose an online multi-step traffic congestion duration prediction algorithm, which can support travelers achieve reasonable trip plans and routes according to realistic environmental changes. The main contributions of this paper are summarized as follows:

- We propose an online cognitive segments sharing framework based on consortium blockchain for secure and efficient data management. We formulate the cognitive structure and interactions of different participants.
- We design an anomaly detection and filtering mechanism based cognitive model of traffic situation to ensure the accuracy of cognitive segments and evaluate the confidence level of CAVs for online perception and credit assessment.
- To resist malicious attacks and improve the consensus efficiency, we establish the credit evaluation mechanism for participants and propose a CDBFT consensus algorithm.
- We propose the online multi-step congestion duration prediction algorithm based on long short-term memory (LSTM) for multi-step forecasting. Extensive experiments are conducted to evaluate the effectiveness of the proposed algorithms compared with existing methods.

The remainder of this paper is organized as follows. In Section II, we discuss the related works. Section III introduces the proposed BCIoVs framework in detail. Section IV formulates the anomaly detection and filtering mechanism based traffic situation cognitive model as well as the credit evaluation for BCIoVs. Section V presents the CDBFT consensus algorithm, the online multi-step traffic congestion duration prediction algorithm, and theoretical analysis. Section VI shows and discusses the simulation results. Finally, we summarize this paper in Section VII.

II. Related Works

In this section, we discuss the motivation by investigating the related works on the blockchain-enabled IoVs, machine learning (ML) for predicting traffic flow, and the integration of blockchain and ML for CITS, respectively.

A. Blockchain-enabled IoVs

In recent years, blockchain has attracted widespread attention in IoVs from academia and industries for its characteristic of decentralization, security, and anonymity [11], [25]. In [26] and [27], the authors proposed a secure information sharing approach for vehicle edge computing and networks (VECONs) based on consortium blockchain. Sharma *et al*. [28] proposed a blockchain-based distributed cluster optimization model to reduce the consumption of energy for IoVs. Wang *et al*. [29] proposed a data sharing mechanism to disseminate vehicular data in connected autonomous driving scenarios through blockchain technology and exploit reputation appreciation and task rewards to incentivize CAVs to deliver reliable content. Jiang *et al*. [30] proposed a blockchain-based secure and distributed big data storage architecture and analyzed the broad prospects of blockchain applications in IoVs. Cheng *et al*. [31] performed traffic flow analysis and control by collecting sensory data from vehicles with an attribute-based blockchain framework to obtain a trade-off between the effectiveness of data dissemination and privacy protection. F. Ayaz *et al*. [32] proposed a blockchain-based message dissemination method in vehicular networks to ensure message authenticity and protect user privacy.

Nonetheless, blockchain protects user privacy in the dissemination and sharing of data, still issues of perceived data quality control and malicious node attacks that cannot guarantee the reliability of information before being packaged into the block. In addition, the current consensus mechanism fails to satisfy the security and efficiency requirements of large-scale CIoVs with many types of participants.

B. ML for Predicting Traffic Flow

Guo *et al*. [33] proposed an attention-based spatial-temporal graph convolutional network (ASTGCN) for traffic flow prediction. The authors in [34] proposed an LSTM-based traffic flow time series prediction method to predict future traffic flow based on historical data and optimize route planning to reduce urban road congestion. Zhao *et al*. [35] proposed a traffic flow forecast scheme based on the temporal graph convolutional network (T-GCN) model which combines GCN and gated recurrent unit (GRU) to obtain traffic flow spatio-temporal features from traffic data. Dai *et al*. [36] presented a short-term traffic flow forecasting model that combines spatio-temporal analysis with GRU. Gu *et al*. [37] proposed an ML-based bayesian combinatorial framework that assembles GRU neural network (GRUNN), radial basis function neural network (RBFNN) and autoregressive integrated moving average model (ARIMA) for traffic flow prediction. Zhou *et al*. [38] proposed reinforced spatial-temporal attention graph (RSTAG) neural networks to perform traffic flow prediction. Jin *et al*. [39] constructed a hybrid traffic flow prediction model that includes stacked autoencoders as well as an LSTM. Chowdhury *et al*. [40] proposed a traffic congestion prediction model and the congestion level is modeled based on the historical traffic flow congestion table of the intersection.

However, these methods lead to a high computational cost and unable to identify and predict the traffic congestion dura-

tion from congestion to smoothness, which is a more direct and informative indicator for intelligent long-term driving plans and arrangements.

C. Integration of Blockchain and ML for CITS

Song *et al*. [41] proposed a blockchain-based cooperative vehicle location method based on a deep neural network (DNN) and established a multi-intelligent vehicle positioning error sharing model using blockchain subsystem to improve the positioning accuracy of ordinary vehicles. Fu *et al*. [42] proposed blockchain-based collective learning (BCL) framework to support large-scale CAVs scenarios. The framework reduces communication costs and improves the accuracy of ML through the local training models and blockchain. Fu *et al*. [43] proposed an autonomous lane-changing system leveraging deep reinforcement learning (DRL) and vehicular blockchain. Based on BCL framework CAVs upload privileged information extracted from the local ML model to update global learning models through the blockchain system for lane changing. Jiang *et al*. [14] proposed a blockchain-enabled distributed deep learning (DDL) framework to perform object detection to improve the performance of ADS. Pokhrel *et al*. [44] proposed a federated learning framework based on blockchain (BFL) to enhance the performance and protect the privacy of autonomous vehicles.

The integration of blockchain and ML enables the sharing of data or ML models for intelligent decentralized decision-making of CAVs. However, the conventional offline ML methods based on fixed historical datasets are incapable of the dynamic and complex real-life traffic environment due to the inability to conduct online cognition of traffic situation.

III. Framework Design

In this section, we introduce the proposed blockchain-enabled online cognitive segments sharing framework. The main notations in this paper are illustrated in Table I.

A. BCIoVs System Architecture

Fig. 1 shows the architecture of the blockchain-enabled cognitive segments sharing of BCIoVs. The cognitive segment is the representation of the cognitive result of the traffic environment, which includes road information, traffic situations, time, etc. The roadside units (RSUs) equipped with CEs are deployed at the roadside, denote as $\mathcal{R} = \{r_1, r_2, \cdots, r_m\}$, m is the number of RSUs. The gateway CAVs v^g in a cluster perform data collection and derive the cognitive segments of the road section when CAVs are not within the communication range of RSUs, $v^g = \{v_1^g, v_2^g, \cdots, v_G^g\}, v^g \subseteq \mathcal{V}$. The RSU and gateway upload the cognitive segment to the nearby mobile edge computing nodes (MECNs), the set of MECNs is $\mathcal{M} = \{m_1, m_2, \cdots, m_s\}$. The CAVs set as the source of sensory data are equipped with various sensing devices, such as cameras, LIDAR, radar, GPS, onboard sensors, etc., which generate a large amount of raw road environment data, high definition map data, and sensory data. The set of CAVs is $\mathcal{V} = \{v_1, v_2, \cdots, v_n\}$, where n is the number of CAVs.

TABLE I
Main Notations

Notation	Definition
$\mathcal{R}, v^g, \mathcal{V}$	Set of RSUs, gateway, and CAVs
$s_v(n)$	Confidence level of CAV v_n
$C_{d,j}(t)$	Vehicle density of the road section j in timeslot t
L_j,u_j,w_j	Length, number of lanes, and line width of road j
l_i,w_i	Length and width of the vehicle i
$P_j(t)$	Traffic efficiency of the road section j in timeslot t
v_m,$\overline{v}(t)$	Maximum speed and average velocity in timeslot t
$A(t)$	Severity level of road traffic incidents in timeslot t
q	Flatness of the road surface
$R(t)$	Road flow index in timeslot t
$v\_in(t)$,$v\_out(t)$	Volume of traffic flow in timeslot t
K_{xi}^{pu}, K_{xi}^{pr}	Public-private key pair for participant x
G	Set of cognitive segment Cog_l
SK_m	Private key of the TA m
P_b	Select priority of BNs
N_v	Number of participated CAVs
$R_{vi}(t)$,R_j,R_{Cx}	Credibility of CAVs, RSUs, and CNs
$p_{vi}(t_k)$, $s_{vi}(t_k)$	Positioning error and confidence level of the velocity in the k-th consensus cycle
N_{cog}, S	Number of cognitive segments and computational load
$P_{ij}(t)$	Sequence of traffic efficiency of road i at timeslot t
$P_r(t)$	Road network traffic efficiency in region r
$D(t)$	Total amount of data in timeslot t
$K(t)$	Total volume of cognition in timeslot t
$V_d(t)$	Data volume of proposed scheme in timeslot t
N_c	Number of selected CNs set
N_n	Number of all CNs in the consortium
T_b	Consensus node selection cycle
N_b	Number of new blocks generated in cycle T_b
N_p	Number of full nodes FNs

Cognitive segments about the current traffic situation can be deduced by analyzing sensory data for online learning and prediction. The prediction results can also be stored and shared through the proposed BCIoVs system to form a cognitive loop. As shown in Fig. 1, the process of cognitive segments sharing in BCIoVs involves the following steps:

1) Gateway v_j^g and RSU R collect sensory data of CAVs and perform local cognition to get cognitive results P_s according to the proposed traffic cognitive model;
2) Cognitive node uploads P_s to the nearest mobile edge server for the local fusion of traffic situation information;
3) The local road traffic condition cognitive segment Cog_l is verified to form the global cognitive view $\boldsymbol{G} = \{Cog_1, Cog_2, \cdots, Cog_l, \cdots, Cog_L\}$ of the road network which can be utilized as a basis for route planning;
4) Package the global cognitive segments into the new block and perform a consensus process to concatenate the new block to the blockchain;
5) The CEs download the blocks to perform online learning and forecasting. The prediction results are then shared through the proposed blockchain system.

B. Component and Structure for BCIoVs

Consortium blockchain is a semi-open system with access rights and advantages of moderate cost, high throughput, and high scalability for similar organizations and industry applications, making it well suited for building a decentralized sharing system [45], [46]. The virtualization for distributed ledger technology is exploited to build the consortium blockchain platform for cognitive segments sharing [47]. As shown in

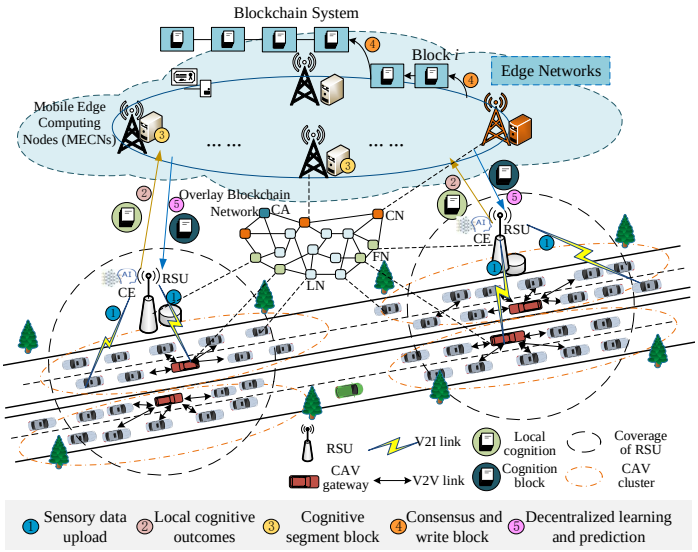


Fig. 1. Architecture of BCIoVs.

Fig. 1, each participant in the BCIoVs system is mapped as a virtual node in the overlay blockchain network [48].

1) Components and functions: The certificate authority (CA) participates in the consortium blockchain with three types of nodes, namely consensus nodes (CNs), full nodes (FNs), and the light node (LNs) [49]. CNs are involved in the consensus and bookkeeping nodes (BNs) selection for the new block, while the other types of nodes are only responsible for requesting, broadcasting, and sharing ledgers [50]. Specifically, the components and functions are as follows:

- CA: The CA is responsible for the management of identity and legitimacy, issuing certificates, and distributing public and private keys through secure channels based on public key infrastructure (PKI). It records the pseudo-identity and manages the dynamic joining and exiting of vehicles by employing a blacklist to improve the flexibility of the network [18].
- LNs: LNs correspond to CAVs, which provide native sensory data to FNs for the cognitive procedure and do not have to keep the ledger information of the whole network which are located at the perception layer for uploading sensory data.
- FNs: FNs correspond to RSUs or CAV gateways, which play the role of local cognitive units, and are mainly responsible for local cognition, uploading cognitive segments, and performing prediction. It maintains the blockchain system by storing, broadcasting, and synchronizing blocks.
- CNs: CNs refer to MECNs, which are mainly responsible for the fusion of cognitive segments and completing the consensus process.
- BNs: BNs are selected from the CNs according to the priority P_b. It is responsible for collecting cognitive segments and generating new blocks.

2) Cognitive Blockchain Structure: The cognitive segment refers to the cognitive results with a specific data format packaged in the block. As shown in Fig. 2, each cognitive

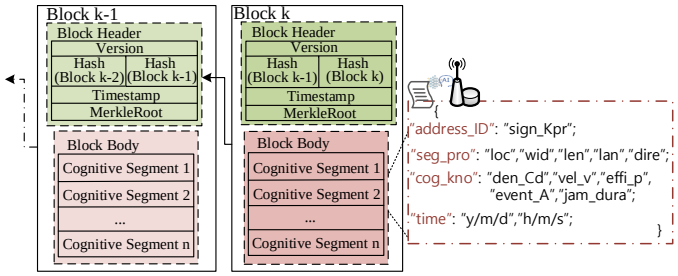


Fig. 2. The structure of cognitive blockchain.

block consists of two parts: the block header (BH) and the block body (BB) [43]. The BH contains the hash of the previous block and current block, and the timestamp, etc. The BB records all valid cognitive segments in each block generation cycle. FN records the complete ledger (i.e., all blocks), while LN, due to computation and their requirements, stores only the metadata of the blocks (i.e., block headers) [51]. Fig. 2 shows the structure of cognitive segments recorded in the blockchain. The cognitive segment Cog_s includes the digital signature ID of the FNs, the road information (location, length, width, number of lanes, direction), the cognitive results of traffic situations (road vehicle density, average speed, traffic efficiency, event, and congestion duration) and the output time, which is represented as $Cog_s\{"address\_ID","seg\_pro","cog\_kno","time"\}_{FN}$. Cognitive blockchain is a distributed ledger, leveraging hash cryptography and consensus mechanisms to string blocks together for distributed storage that does not rely on third party [52].

C. Authorization and Verification

Elliptic curve digital signature technology (ECDSA) and hash encryption algorithms (SHA-256) are exploited in the initialization phase of the BCIoVs system. The legitimate user authorized by CA receive the asymmetric public-private key pair $\{K_{xi}^{pu}, K_{xi}^{pr}\}$ and delegate anonymous identity to encrypt the shared data for identity verification and authorization management. The public key is visible to all participants in the system, but the private key is kept only by its owner. The authorization records T_X endorsed by CA can be written as

$$T_x = \{[K_{x1}^{pu}, K_{x1}^{pr}], [K_{x2}^{pu}, K_{x2}^{pr}], ..., [K_{xi}^{pu}, K_{xi}^{pr}]\}_{Sk_m}, \quad (1)$$

where Sk_m is the private key of the CA, and K_{xi}^{pu}, K_{xi}^{pr} is the public key and private key of authorized users, respectively. The private key K_{xi}^{pr} is utilized to generate a digital signature allowing for mutual recognition of identity and knowing whether the participants are legitimate. The sender signs the message with the digital signature and the receiver verifies the identity of the sender with the public key K_{xi}^{pu} to ensure the integrity of the received message cannot be tampered with, destroyed, and forged. The signature can be easily verified by any other node that uses the public key of signers [16].

D. The Interaction and Workflow of BCIoVs

Cognitive segment sharing framework mainly includes the perception layer, cognitive layer, and consensus layer with

different properties and functions. The interactions and workflow among CA, LNs, FNs, and CNs are shown in Fig. 3. Firstly, the LNs, FNs, and CNs register with the CA to obtain legal identity and public-private key pair $\{K_{xi}^{pu}, K_{xi}^{pr}\}$ in the initialization phase. Then FNs send crowd sensing tasks request to LNs and LNs upload perception data to FNs for environmental cognition in the cognitive phase. Finally, FNs upload cognitive segments to CNs for consensus sharing in the consensus phase.

Specifically, the CAVs upload sensory data to the CAV gateway or RSUs through the 5G V2X communication in the perception layer [53]. In the cognitive layer, RSUs as cognitive producers create local cognitive segments Cog_j of the traffic situation and signed by the private key K_j^{pr} to ensure the authenticity of the submitted cognitive information. The cognitive segments Cog_j produced by RSU j are uploaded to $MECN_x$ for consensus and the global traffic view can be obtained in the consensus layer. The packaged message uploaded by RSU j is $UP_{RSU_j \to MECN_x}\{K_j^{pu}, Cog_j, C_r, timestamp\}_{sign\_K_j^{pr}}$, where K_j^{pu} is the public key and be used to decrypt signature and verify the authenticity, Cog_j is the uploaded cognitive segment set, C_r represents the credit value of uploader. The $MECN_x$ accepts the cognitive segments uploaded by multiple RSUs and verifies the file content hash and signatures are correct. Once the validation is passed, The $MECN_x$ will pack all the cognitive segments G_x into new record message and add other information for broadcast, $BRO_{MECN_x}\{G_x, K_x^{pu}, hash, timestamp\}_{sign\_K_x^{pr}}$, where $sign\_K_x^{pr}$ is the signature of $MECN_x$ to ensure legality and verify the message has not been tampered with [43], [54].

The $MECN_x$ collects uploaded cognitive segments regularly to form a cognitive pool to preserve the congnitive segments to be packaged [16]. For each period, the selected BN packs the collected cognitive segments into a new block proposal and broadcasts it to other $MECN_x$ for consensus. If the block is verified and voted by most of the CNs, it will be added to the end of the blockchain. Then, the RSUs perform distributed online congestion duration prediction by observing and learning from the environment and store the prediction results in the blockchain as well.

IV. Constructions of Cognitive Model and Credit Evaluation Mechanism

This section formulates the cognitive model of traffic situation based on anomaly detection and filtering mechanism and establishes a credit evaluation mechanism to resist malicious attacks in BCIoVs.

A. Anomaly Detection and Filtering Mechanism

Participants may upload incorrect or false data leading to inaccurate cognitive results due to perception failures, malicious attacks, or selfish reasons [55]. In the real-life environment, road congestion conditions are generally different from driving directions. However, the locations perceived by CAVs may deviate and cannot accurately reflect the status of the road in the same direction. To delineate data analysis boundaries and

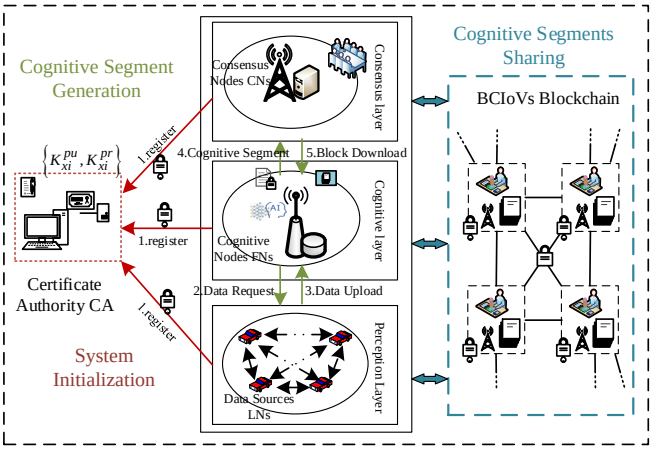


Fig. 3. The interaction among CA, LNs, FNs, and CNs.

get fine-grained two-way road traffic conditions, we consider the driving directions from the directional sensors as well as the changes of position. The number of CAVs N_v in the same direction can be defined as

$$N_v = \sum_i V_i \ , \quad V_i = \begin{cases} 1 & Samedirection \\ 0 & Otherwise \end{cases} \quad i = 1, 2, ..., n. \tag{2}$$

In addition to location information, velocity and vehicle density are also important indicators for judging traffic conditions [56]. If the vehicle density is high, the velocity distribution will be relatively concentrated, the reason is that the degree of driving freedom is reduced. The average velocity reflects the traffic efficiency and the velocity distribution in the same road section is relatively stable which does not change drastically.

Based on the above analysis, we design the anomaly detection and filtering mechanism in the cognitive layer. The average velocities v_i involved in crowdsensing is ranked to obtain the average velocity sequence $N_v\_V = \{v_1, v_2, v_i, \cdots, v_{N_v}\}$. The filtering mechanism is formulated as

$$\begin{matrix} U = v_{Q_3} + \lambda(v_{Q_3} - v_{Q_1}) \\ B = v_{Q_1} - \lambda(v_{Q_3} - v_{Q_1}) \end{matrix} \ , \tag{3}$$

where, $v_{Q_3} = N_v\_V(round(N_v + 1) * 0.75)$ is the upper quartile, $v_{Q_1} = N_v\_V(round(N_v + 1) * 0.25)$ is the lower quartile of N_v_V, λ is the filter step size. The filtered velocity distribution can be expressed as

$$v_j = \begin{cases} v_i & B \le v_i \le U \\ 0 & Otherwise \end{cases} \quad v_i \in N_v\_V, \tag{4}$$

where U and B are the upper and lower bounds of data filtering, respectively. The average velocity obtained after the filtering mechanism can be derived as

$$\overline{v} = \frac{\sum_j^{n_x} v_j}{n_x} \ j = 1, 2..., n_x, \tag{5}$$

where v_j and n_x denote the filtered velocity and number of vehicles, respectively.

Malicious and unreliable nodes are identified effectively in this process based on the authenticity of the provided data.

Then, we get the confidence level $s_v(n)$ of CAV v_n, which is a crucial indicator for credit evaluation.

$$s_v(n) = e^{-(|v-\bar{v}|/\bar{v}+k_n)}, \tag{6}$$

where k_n denotes the number of times participant n was filtered out. In addition, CA can add unreliable and malicious data providers to the blacklist based on their confidence level.

B. Cognitive Model of Traffic Situation

The traffic efficiency is defined as the average vehicle throughput carried by the road section per travel time unit, which is a crucial metric to scale the road congestion level and driving status of the traffic situation The timestamp feature of blockchain fits well with the intense time correlation of traffic scenarios. The density $C_{d,j}$ of the road section j is derived as

$$C_{d,j}(t) = \frac{\sum_{i=1}^{N_v(t)} l_i * w_i}{L_j u_j w_j}, \tag{7}$$

where L_j denotes the length of the road section j, u_j and w_j are the number of lanes and the width of the lane in the road section j; l_i and w_i are the length and width of the vehicle i respectively. However, road density alone cannot accurately reflect traffic efficiency. For instance, if vehicles travel with a high velocity, the traffic efficiency can still be at a high-level [23]. Thus, it is not scientifically sound to determine traffic efficiency only based on road density [57].

Traffic efficiency is affiliated to a variety of factors, such as road type, road event, vehicle density, the average speed [15], [21], [22]. Lower traffic efficiency means that the roads with high congested levels. To formulate the variation of vehicles on the road, the road accessibility index $R(t)$ is defined according to the number of vehicles entering and leaving the road section per time interval, $R(t) = v\_out(t)/v\_in(t)$. We formulate traffic efficiency P_j of road section j based on anomaly detection and filtering mechanism as

$$P_j(t) = \frac{q\overline{v}(t)R(t)}{v_m A(t) C_{d,j}(t)}, \tag{8}$$

where v_m is the maximum free speed, $\overline{v}(t)$ represents the average velocity over the timeslot t. The quality of the road is q, which represents the flatness of the road, $q \in (0, 1]$, where 1 represents the road with the best road quality and flatness. We use $A(t)$, where $A(t) \in [1, 5]$, to scale the severity level of road traffic incidents. 1 denotes normal road conditions with no accidents or safety incidents and 5 represents a serious accident occupies more lanes with a long processing time.

C. Credit Evaluation Mechanism

The credibility evaluation for each type of node stimulates highly credible nodes to contribute reliable data and prevent attacks from malicious nodes [58]. As the proposed CDBFT consensus algorithm in section V, the selection of BN is no longer simply random ranking rotation of CNs. It is not only related to the credit value but also the resource allocation status for the efficiency and security of the CDBFT. We first define the BN priority P_b as

$$P_b = \alpha N_{cog} + \kappa/S + \omega R_{Cx}, \tag{9}$$

where R_{Cx} is the credibility of CN x. N_{cog}, S are the number of cognitive segments and the computational load, respectively. Moreover, α, κ, and ω are the weight coefficients. The credibility measurement R_{vi} of CAV i is mainly based on the reliability, confidence level, and quantity of contributed data as follows,

$$R_{vi}(t) = \frac{1}{\varphi}\sum_{k=1}^{\varphi}\left[ap_{vi}(t_k) + \beta s_{vi}(t_k) + \gamma c_{vi}(t_k)\right] \times e^{-\eta(t-t_k)}, vi \in V. \tag{10}$$

The credit of RSU j R_j is mainly based on the accuracy and timeliness of cognitive task completion as follows,

$$R_j = (1 + e^{-\omega q_j}), \forall j \in R, \tag{11}$$

where φ is the number of timeslot t, $c_{vi}(t_k)$ represents the contribution degree, i.e. the amount of data provided, q_j is the volume of task completion of RSU j. The $p_{vi}(t_k)$, $s_{vi}(t_k)$ are the positioning error and confidence level respectively, which is determined based on the authenticity and quality of data by anomaly detection and filtering mechanisms [29].

The credit R_{Cx} of the CN x changes in real-time depending on the normal and abnormal behavior of the consensus process. Normal behavior, i.e. sending transactions in compliance with the system rules, gradually increases the credit value over time, and the abnormal behaviors decrease the credit value. The credibility function of CN can be written as

$$R_{Cx} = \eta_1 R^P_{Cx}(t) + \eta_2 R^N_{Cx}(t), \tag{12}$$

where η_1, η_2 are the weight coefficients, and R^P_{Cx}, R^N_{Cx} represent the normal and abnormal behaviors that affect the credibility of CN x, respectively. By adjusting coefficient η_2, we can obtain a stricter penalty strategy.

R^P_{Cx} related to the activity of CNs as positive to the number of normal transactions per unit of time is derived as

$$R^P_{Cx} = \frac{1}{T}\sum_{b=1}^{nx}(1 + \eta_3 w_b), \tag{13}$$

where T is the time interval, nx denotes the number of new blocks verified by node x in T, w_b is the number of valid transactions contained in the block b, η_3 denotes the weight, i.e., the times of the block is verified.

R^N_{Cx} is related to the abnormal attack behavior of CN, we consider two typical attacks models, *Blackhole miners* and *Colluding attacks*. *Blackhole miners* is referred to that the selected CN refuses to submit transactions and perform validation causing delays in the consensus process, thus affecting the real-time performance of the system. *Colluding attacks* is malicious CNs conspire together to perform biased consensus and add malicious blocks to the chain. Multiple malicious nodes in a selected CNs set may add false data to the blockchain, thus affecting the correctness of the online prediction system [59]. Thus, we derive the negative function R^N_{Cx} of consensus as follows:

$$R^N_{Cx} = -\frac{1}{T}\sum_{k=1}^{mx} a(\varphi)_k, \tag{14}$$

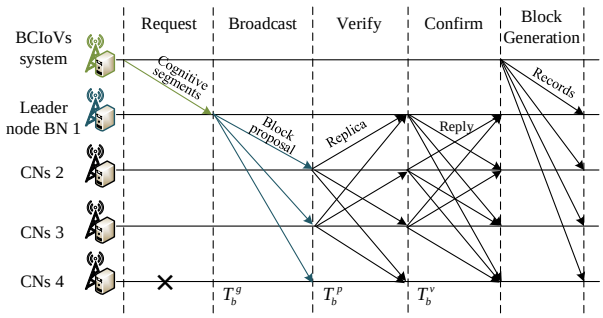


Fig. 4. The consensus process of CDBFT.

where $a(\varphi)_k$ denotes the penalty factor of the *k-th* malicious act φ. Then, we defined $a(\varphi)_k$ as

$$a(\varphi)_k = \begin{cases} a_b & \varphi \in blackhole\ miners\ behaviors \\ a_c & \varphi \in colluding\ attacks\ behaviors \end{cases}, \quad (15)$$

which can be adjusted according to the severity of the malicious behavior and the requirement of security sensitivity of the system. Thus, the credibility of CN x can be derived as

$$R_{Cx} = \frac{\eta_1}{T}\sum_{b=1}^{nx}(1+\eta_3 w_b) - \frac{\eta_2}{T}\sum_{k=1}^{mx} a(\varphi)_k, \quad (16)$$

where mx represents the total number of malicious acts of node x.

V. Algorithm Design

In this section, we propose a CDBFT consensus algorithm as well as an online multi-step congestion duration prediction algorithm based on the proposed cognitive sharing framework. Then, we conduct the theoretical analysis.

A. CDBFT Consensus Algorithm

The proposed CDBFT consensus algorithm is presented as Alg.1 to shorten the block confirmation delay and against malicious attacks. The proposed consensus algorithm facilitates the decentralized cognitive system by selecting a CNs set with a higher credit value ensuring the security and efficiency of the system. In a consensus cycle, the selected CNs set is chosen from all CNs candidates based on the credit evaluation mechanism to perform consensus. The optimal selected CNs set C_x^* can be obtained by solving the following problem,

$$C_x^*(t) = \arg\max_{x \in MECNs}(R_{Cx}). \quad (17)$$

We choose the top-ranked CNs candidates as the selected CNs set and the BN is chosen from the selected CNs set with the highest priority P_b to participate in the consensus process. Assume that the number of selected CNs set is N_c. Then, the maximum number of fault-tolerant nodes is $f = (N_c - 1)/3$, and the proposal is passed when the number of votes is no less than $N_c - f$ [60]. As shown in Fig.4, the main steps of the CDBFT consensus process are as follows:

a) Broadcast: The selected BN of this round packaged cognitive segments for initiating the proposal and broadcasting the unconfirmed block to other selected CNs to perform voting and propagating new blocks outwards in the peer-to-peer (P2P) network. Then, moving to the next round of consensus.

Algorithm 1 CDBFT Consensus Algorithm

Initialize the public keys and signatures of all CNs.
for $t = Consensus\ cycle$ **do**
  procedure CNs and BN selection.
  Calculate and sort the credit value of all CNs.
  $selected_{set}(CNs) \leftarrow get\ N_c\ CNs\ by\ voting.$
  $BN \leftarrow highest\ P_b(selected_{set}(CNs))$.
  end procedure.
  procedure Block proposal and consensus.
  $\boldsymbol{G} \leftarrow BN\ \{Cog_1, Cog_2, \cdots, Cog_l, \cdots, Cog_L\}$.
  if $BN_{Signature} = True$ **then**
    $Block_k = generate\ unverified\ block$;
  else
    Re-select BN;
  end if
  $CNs_{set} \overset{BRO}{\leftarrow} Block_k\{BH||BB||timestamp\}_{Sig_{BN_i}}$.
  $BN_i \overset{REP}{\leftarrow} Block_k\{CN_j^{ID}||vote||timestamp\}_{Sig_{CN_j}}$.
  if $vote = True$ **then**
    $conf++$;
    if $conf \geq 2f+1$ **then**
      $Block_k\ Confirmation$;
    else
      $Resend\ broadcast\ BRO$;
    end if
  end if
  end procedure
end for

b) Validation: The selected CNs validate the block based on the identifier of the public key and signature. If the verification passes, the block will be multicast using their signatures for mutual confirmation. The selected CNs broadcast the verified transactions to initiate voting and confirmation.

c) Confirm: The selected CNs sign the verified block for signature confirmation. All selected CNs send confirmation messages to others. Once receives no less than $N_c - f$ $\langle block \rangle_{\alpha i}$ signatures indicates the commit state is entered.

d) Block Generation: Consensus is achieved when the proposal has been endorsed by more than $N_c - f$ of the selected CNs. Then, the new block will be concatenated to the blockchain indicating successful cognitive sharing. The process of generating new blocks takes place simultaneously with the CDBFT consensus process.

B. Implementation of Online Multi-step Prediction

Fig. 5 shows the implementation of the proposed online learning model, which performs decentralized training and prediction based on real-time cognition and observation of traffic efficiency shared by BCIoVs. Once a roadway is deemed congested which can be obtained from the statistics of traffic efficiency, the observation and learning will be triggered to record the congestion duration changes with different traffic efficiency and a matching list $\{\boldsymbol{P}_r, \boldsymbol{C}_r\}$ is updated by the cognition and observation system for online multi-step prediction.

Based on the above analysis we formulate the online traffic congestion duration cognition and prediction model. The traffic efficiency of road section j on road i at the time t is

$P_i^j(t)$. The road congestion duration is identified based on the timestamp of traffic efficiency, which is a sequence variation about time. The sequence of traffic efficiency of road i at time t can be expressed as

$$\mathbf{P}_{ij}(t) = \left\{P_i^1(t), P_i^2(t), ..., P_i^s(t)\right\} \; j = 1, 2, ...s, \tag{18}$$

where s is the road sections. Then the sequence of the traffic efficiency of road i in the time dimension can be expressed as

$$\boldsymbol{P}_i^{seq}(t) = \{\mathbf{P}_{ij}(t - l * \Delta T), ..., \mathbf{P}_{ij}(t - \Delta T), \mathbf{P}_{ij}(t)\}. \tag{19}$$

Considering the spatial and temporal correlation, the road network traffic efficiency in region r can be expressed as

$$\boldsymbol{P}_r(t) = \begin{bmatrix} \boldsymbol{P}_1^{seq}(t - l * \Delta T) & \dots & \boldsymbol{P}_1^{seq}(t) \\ \vdots & \ddots & \vdots \\ \boldsymbol{P}_m^{seq}(t - l * \Delta T) & \cdots & \boldsymbol{P}_m^{seq}(t) \end{bmatrix}, \tag{20}$$

where m is the number of roads and l stands for the number of time steps. Based on the timestamp of the cognitive segments $\boldsymbol{P_r}$, we can obtain the change in traffic congestion duration $[\boldsymbol{C}(t - xstep), ..., \boldsymbol{C}(t)]$ of road m using $timestamp_{p_r}^{jam} - timestamp_{p_r}^{clear}$. Tensors on traffic efficiency and congestion duration $\{\boldsymbol{P}_r, \boldsymbol{C}_r\}$ are recorded and updated through online learning. Then the distributed online multi-step prediction model is trained and predicted based on the tensors obtained from the real-time observation and statistics of traffic efficiency. The online multi-step prediction model is defined as

$$\{\boldsymbol{P_r}, [\boldsymbol{C}(t - xstep), ..., \boldsymbol{C}(t)]\}_{step} \overset{H(\bullet)}{\rightarrow} \boldsymbol{X}_{multi\_step}, \tag{21}$$

where $H(\bullet)$ denote the proposed prediction algorithm, $\boldsymbol{X}_{multi\_step}$ is the predicted future multi-step congestion duration, $\boldsymbol{C}_i^{sim} = \{\boldsymbol{C}(t + step), \boldsymbol{C}(t + xstep), ...\}_{sim}$. Online learning needs to continuously adjust the prediction model through real-time feedback from the environment to obtain higher accuracy and adaptability. The online multi-step prediction utilizes actual observations rather than predicted values to update the network and minimize the objective error function by solving the following problem,

$$\begin{aligned} &Min\ error\left\{MSE\left[\boldsymbol{C}(t + step), ..., \boldsymbol{C}(t + xstep)\right]\right\} \\ &S.t.\ MSE = \frac{1}{N}\sum_{i=1}^{N}\left(\boldsymbol{C}_i^{obs} - \boldsymbol{C}_i^{sim}\right)^2, \end{aligned} \tag{22}$$

where $\boldsymbol{C}_i^{obs}$ denote the observed values, $\boldsymbol{C}_i^{sim}$ is the predicted multi-step values, N is the size of steps. The loss function is mean square error (MSE) to update the training model.

C. Online Multi-step Prediction Algorithm

To address the above problem, we propose an online multi-step traffic congestion duration prediction algorithm based on LSTM as shown in Alg.2. The learning rate Lr is decreased with training iterations $\zeta \leq 125$ to accelerate convergence for fine-grained optimization search.

As shown in Fig. 5, the LSTM is a chain structure with many LSTM units which is responsible for state transfer of input sequence at different time steps [61]. Each unit includes three gating cells z^f,z^i and z^o are forgotten, input,

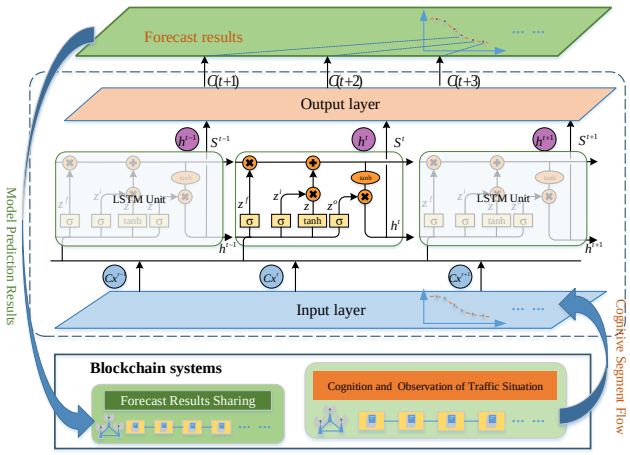


Fig. 5. The structure of the online prediction model.

Algorithm 2 Online Multi-step Congestion Duration Forecast

Input: Training samples variables, Initialization parameters.
Output: Predictions $\boldsymbol{C}(t + xstep)$, Performance indicators.

```
for ℜ = 1, ..., ϖ do
    Performing data collection and pre-processing.
    Initialize the maximum iterations Γ and set the starting iteration τ = 0, regulators ζ, ϑ.
    Anomalous data filtering using Eq. (2), (3), (4).
    Calculate traffic efficiency P_r using Eq. (7), (8), (5).
    if Congestion Triggers then
        Online cognitive systems observe changes in traffic efficiency and congestion duration {P_r(t), C(t)}.
        while τ ≤ Γ do
            Data standardization;
            if τ ≤ ζ then
                Learning rate = Lr;
            else
                Learning rate = Lr * ϑ;
            end if
            Calculate h^t using formulations Eq.(23-25);
            Calculate MSE following Eq. (22);
            Minimize the objective function MSE through BPTT and update the network parameters;
            τ = τ + 1;
        end while
    else
        Break
    end if
end for
```

and output gates, respectively. Where z is the cell renewal state [62]. The current cell gate S^t update based on the previous cell sate S^{t-1}, z, z^f, z^i. Then, the forgetting gate z^f discards unimportant information by controlling the weights, the current output of hidden state h^t is obtained according to the output gate z^o and the cell state S^t. The control update function can be derived as matrix operation,

$$\begin{Bmatrix} z \\ z^i \\ z^f \\ z^o \end{Bmatrix} = \begin{bmatrix} \tanh \\ \sigma \\ \sigma \\ \sigma \end{bmatrix} \left(W_i \bullet \left[h^{t-1}, Cx^t\right] + b_i\right), \tag{23}$$

where Cx^t is current input, h^{t-1} is previous hidden state, W_i, b_i are weights and bias of each network layer and σ is the sigmoid activation function. In the current LSTM unit, the dimension of the input vector x^t is p, the dimension of the hidden state vector $h^{t-1}$is u, thus W_i is a $p+u$ dimensional weight vector. The above four states are used as gating units to control the selective forgetting, selective memory and output stages. We can obtain the updated status,

$$S^t = z^f \odot S^{t-1} + z^i \odot z, \tag{24}$$

$$h^t = z^o \odot \tanh(S^t), \tag{25}$$

where $\odot$ is the Hadamard product, S^t is the current cell state and h^t is the hidden state. The network update process of the proposed algorithm consists of the selective forgetting phase, selective memory phase, and output phase. Selective forgetting of the unimportant input from the previous node and remembering the meaningful information. The z^f acts as a forgetting gate to control what needs to forget in the previous cell state of S^{t-1}. The selective memory phase is mainly for selective memory of the input Cx^t, the current input content is represented by z. The output phase determines the output of the current state, which is related to z^o and S^t. The corresponding predicted values are obtained after the output layer. Therefore the final output of the network is not only related to the current input but also to the previous input.

D. Theoretical Analysis

1) Communication Overhead: We discuss the communication load of the proposed BCIoVs compared to the centralized learning approach, which collects native sensory data to the central server leading to a high communication overhead [42]. Suppose that the amount of data generated by CAV v_j at time t is d_j , then the total amount of data generated in sampling period time T can be expressed as

$$D(t) = \sum_{t=1}^{T} \sum_{j=1}^{n} d_{j,t} \;\; v_j \in V, t \in T. \tag{26}$$

The proposed approach convert large amount of raw data into cognitive segments at the edge network. The set of cognitive segments generated at time interval t by each RSU R_c or gateway G_c is $\{k_1, k_2, ..., k_l, k_g\}$, k_g is the derived cognitive segment from $D(t)$, $k_g = Cog(D(t))$. Then, the total volume of cognition can be derived as

$$K(t) = \sum_{t=1}^{T} (\sum_{r=1}^{m} k_r + \sum_{g=1}^{G} k_g) \;\; R_c \in R, G_c \in v^g, t \in T, \tag{27}$$

where k_r, k_g denote the cognitive segments generated by the RSU and gateway, respectively. Since the data processing and learning are performed on the edge network, we can get the communication overload $V_d(t)$ of the proposed approach in a sampling period is

$$V_d(t) = D(t) + K(t), \; t \in T. \tag{28}$$

For the centralized method, the amount of data transferred is $V_{cd}(t) = \sum_{rely}^{r} D(t), \; t \in T$ since frequent backhaul relays r. Conventional methods that transmit such large amounts of raw data would incur significant communication overhead in a large-scale CIoVs environment. However, our proposed approach can effectively reduce the communication overhead by converting large amounts of raw data into valuable cognition with a small size.

2) Consensus Delay: The consensus time of CDBFT for cognitive segments sharing mainly includes block generation latency, propagation latency, and verification latency. The cognitive block generation time is

$$T_b^g = \frac{\kappa D_B C_m^r}{C_m^a}, \tag{29}$$

where C_m^r , C_m^a represent the amount of computation required for block generation and the computational resources of the BNs, respectively, κ is the number of cognitive segments, and D_B is the size of the generated block. In the P2P network, the block propagation time T_b^p needs to be satisfied that each CN successfully receives the block, which can be defined as

$$T_b^p = \underset{m \in \mathcal{M}}{max} \left\{ \frac{D_B}{C_m} \right\}, \tag{30}$$

where C_m is the communication rate. Similarly, the verification time of selected CNs is

$$T_b^v = \underset{m \in \mathcal{M}}{max} \left\{ \frac{C_v^r}{C_m^p} \right\}, \tag{31}$$

where C_v^r is the amount of computation required for verification and C_m^p is the computational resource provided by selected CNs m. Thus, the total time delay of the consensus process is

$$T_b^{all} = \frac{\kappa D_B C_m^r}{C_m^a} + \underset{m \in \mathcal{M}}{max} \left\{ \frac{D_B}{C_m} \right\} + \underset{m \in \mathcal{M}}{max} \left\{ \frac{C_v^r}{C_m^p} \right\}. \tag{32}$$

Based on the above analysis, the normal consensus delay of a predefined cognitive block is mainly associated with the communication and computing capabilities of the network. In the proposed online system, the communication capability will be greatly improved assisted by 5G. Meanwhile, combining consortium blockchain with mobile edge computing (MEC) can well meet the low-latency requirements of vehicular networking [18]. However, the conventional algorithm cannot perform efficient data cognition as well as the attacks from unscreened CNs will seriously slow down the consensus time.

3) Computation Complexity: Regarding the complexity of the proposed CDBFT consensus algorithm, the selection of CNs based on credit evaluation reduces the number of CNs involved in consensus at each step. Each CNs verify the credit of candidates and vote in each epoch, yielding a total $O(N_c N_n)$ complexity. In a CNs selection cycle T_b, each valid block shares $O(N_c N_n / N_b)$. The FNs N_p package the cognitive segments to all selected CNs to vote for generating new blocks, yielding $O(N_p N_c)$. Thus, the total complexity yielded in one block is $O(N_p N_c + N_c N_n / N_b)$, which is smaller than $O(N_p N_n^2)$ of practical byzantine fault tolerance (PBFT) [16]. In addition, the computational complexity of the prediction phase is $O(N)$ once the online learning is completed, N is the size of the input variables. Thus, the proposed online cognitive and predictive algorithm can be well implemented in real-life traffic scenarios.

TABLE II
COMPARISON WITH EXISTING WORKS

Properties	[5]	[9]	[15]	[17]	Proposed
Data security	√	√	√	√	√
Decentralization	×	√	√	√	√
Online traffic forecast	√	×	×	×	√
Attack resistance	×	×	×	√	√
Cognitive segment sharing	×	×	×	×	√
Data quality improvement	×	√	×	×	√

4) Predictive Performance Analysis: The predictive evaluation indicators can be measured based on the deviation between the predicted and actual observed values. The performance indicators are defined to verify the effectiveness of the online prediction algorithm as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|\boldsymbol{C}_i^{obs} - \boldsymbol{C}_i^{sim}\right|, \tag{33}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\boldsymbol{C}_i^{obs} - \mathbf{C}_i^{sim}\right)^2}, \tag{34}$$

$$MAPE = \frac{100}{N}\sum_{i=1}^{N}\left|\frac{\boldsymbol{C}_i^{obs} - \boldsymbol{C}_i^{sim}}{\boldsymbol{C}_i^{obs}}\right|\%, \tag{35}$$

where MAE denotes mean absolute error and $RMSE$ denotes root mean square error. $MAPE$ refers to the mean absolute percentage error, which represents the quality of the trained model [38], [61]. The smaller the value of the above predictive evaluation indicators, the higher the accuracy of the prediction.

VI. NUMERICAL RESULTS AND DISCUSSIONS

In this section, we conduct extensive experiments to verify the effectiveness of the proposed algorithms compared with the existing methods. Table II illustrates the contribution of the proposed approach compared with the existing related works [5], [9], [15], and [17], our approach can gain more merits.

A. Simulation Settings

The experiments are conducted based on the public real-world dataset PEMS08[1] released by (Li et al. 2021), which is collected by 170 perceptrons on California road with three characteristics of flow, occupancy, and velocity respectively [33], [63]. The virtualization for distributed ledger technology (vDLT) [47] is leveraged to emulate the overlay blockchain network. The total number of CNs N_n in the consortium blockchain is set to $\{40, 50\}$ and the selected CNs set N_c to 21 [16], [18]. The other main parameters are presented in Table III. We implement the proposed algorithms, employing the hash function library and the deep learning toolbox on a computer with an Intel(R) Core(TM) i5-8250U CPU @ 1.60 GHz, 8 GB 1600 MHz DDR4 RAM.

[1]https://github.com/MengzhangLI/STFGNN/tree/master/data

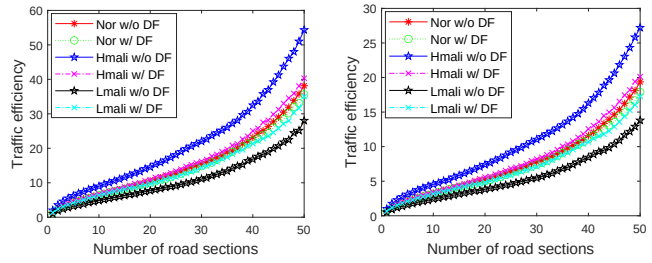
TABLE III
THE MAIN SIMULATION PARAMETERS

Simulation parameters	Value
Speed limits	[0-70] Km/h
Hmali data distribution	$N(70, 20^2)Km/h$
Lmali data distribution	$N(3, 10^2)Km/h$
Filter step size λ	0.1
Road length	$3Km$
Number of lanes u_j	3
Lane width w_j	$3.5m$
RSU interval	$300m$
Ratio of malicious CAVs	0.3
Road quality	0.9
Level of incident A	[1,5]
Number of CAVs	1000
Block size [48]	4MB
Hash algorithm	SHA-256
Maximum block interval	5s
LSTM hidden units	200
Layers	4
Initial learning rate Lr	0.005
Maximum number of iterations	250
Learning rate drop factor ϑ	0.2

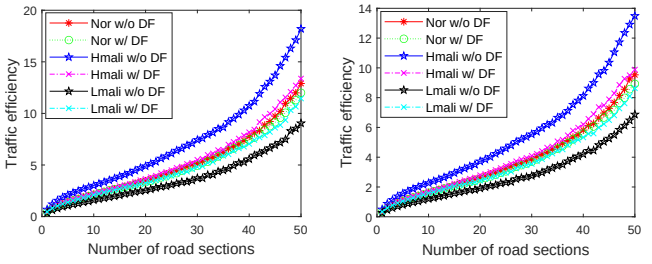
B. Effect of Anomaly Detection and Filtering Mechanism

In this subsection, we consider two anomalous data sets in the experiments as shown in Table III. One is that the malicious velocity value is higher than the true velocity value. The other is that the uploaded malicious velocity value is lower than the true velocity value. We select six schemes for comparison as follows: 1) experiment on the real-world normal datasets without anomaly detection and filtering mechanism, which as the baseline for the comparative analysis, as labelled as "Nor w/o DF"; 2) the real-world normal datasets with anomaly detection and filtering mechanism, as labelled as "Nor w/ DF"; 3) malicious attacks with high values exist in the real-world normal datasets without anomaly detection and filtering mechanism, as labelled as "Hmali w/o DF" [17]; 4) malicious attacks with high values exist in the real-world normal datasets with anomaly detection and filtering mechanism, as labelled as "Hmali w/ DF"; 5) malicious attacks with low values exist in the real-world normal datasets without anomaly detection and filtering mechanism, as labelled as "Lmali w/o DF"; 6) malicious attacks with low values exist in the real-world normal datasets with anomaly detection and filtering mechanism, as labelled as "Lmali w/ DF". For the fairness of comparison, each scheme adopts the same model parameters as Table I.

Fig. 6 shows the cognitive results of the above schemes based on the real-world datasets under different incident severity levels $A(t)$. We can find that the cognitive results of the schemes with anomaly detection and filtering mechanism are close to the baseline value, while that of Hmali w/o DF and Lmali w/o DF are deviates significantly. The reason is that the anomalous data uploaded by malicious nodes are inconsistent with the real traffic situation leading to biased cognitive results. The traffic efficiency of Hmali w/o DF is higher than other schemes because the upload velocity value of malicious nodes is higher than the true velocity value. These two malicious attack models will have the consequences of failing to detect congested roadways and inaccurate traffic congestion perception. The experiment results demonstrate



(a) Cognitive result under A = 1 (b) Cognitive result under A = 2



(c) Cognitive result under A = 3 (d) Cognitive result under A = 4

Fig. 6. Cognitive results of traffic efficiency (Malicious node ratio r = 0.3).

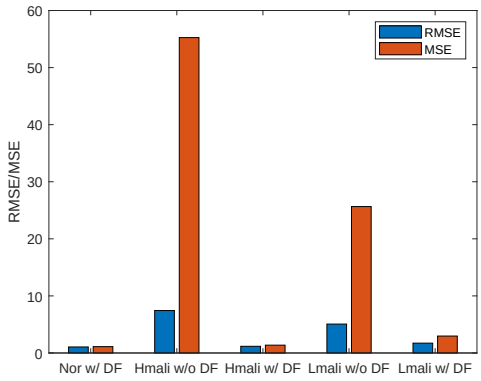


Fig. 7. Performance metrics of cognitive error.

that the interference and attacks from malicious nodes can be significantly suppressed by the proposed anomaly detection and filtering mechanism. Moreover, the traffic efficiency increases as the road section is further away from the congested area and decreases as the severity of incidents increases. The results are in line with the real-life traffic environment proving the validity of the proposed cognitive model.

Fig. 7 shows the cognitive errors of different schemes. We can find that the Nor w/ DF, Hmali w/ DF, and Lmali w/ DF schemes have lower MSE and RMES than Hmali w/o DF and Lmali w/o DF schemes. Table IV lists the MSE and RMSE under different schemes. The results demonstrate that the proposed anomaly detection and filtering mechanism based cognitive model can effectively eliminate the influence of malicious nodes on the cognitive results to ensure the reliability of online prediction.

C. Communication Overhead and Consensus Efficiency

In this subsection, we verify the communication overhead and consensus efficiency under different malicious node ratios

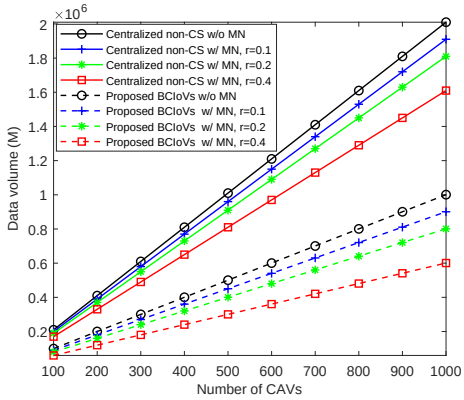


Fig. 8. Comparison of data volume (Malicious node ratio r = 0, 0.1, 0.2, 0.4).

compared with the following methods:

- Traditional centralized scheme without cognitive segment sharing existing no malicious nodes, namely "Centralized non-CS w/o MN" [5].
- Traditional centralized scheme without cognitive segment sharing existing malicious nodes, namely "Centralized non-CS w/ MN" [9].
- Proposed BCIoVs scheme existing no malicious nodes, namely "Proposed BCIoVs w/o MN".
- Proposed BCIoVs scheme existing malicious nodes, namely "Proposed BCIoVs w/ MN".
- DBFT consensus mechanism without credit evaluation, namely "DBFT w/o credit evaluation" [60].
- Proposed CDBFT consensus algorithm, namely "Proposed CDBFT algorithm".
- The "PBFT consensus mechanism" in which all unscreened CNs participate in the consensus process [16].

Fig. 8 shows the communication overhead of different schemes versus malicious node ratios. We can find that the proposed scheme reduced the data volume effectively compared with other schemes. The reason is that a large amount of raw data can be compressed into smaller size cognitive segments and the decentralized edge architecture can reduce the communication relays to the cloud server, which can effectively save bandwidth and energy consumption. In addition, as the proportion of malicious nodes increases, the data volume is further reduced due to the large amount of anomalous data being filtered out to prevent attacks by malicious nodes.

Fig. 9 shows that the proposed algorithm has the shortest consensus delay versus credit node ratio. With the increasing proportion of credit nodes, the consensus delay decreases gradually. The reason behind is that the attack behavior of malicious nodes will seriously slow down the normal consensus process. Another observation is that the consensus delay becomes higher with the increase of CNs. This is because numerous validators (CNs) will take longer to complete the block validation. The results show that without solving a meaningless nonce puzzle, consensus time can reach ms level unlike other consensus methods, such as 600 seconds for Bitcoin and 15 seconds for Ethereum [18].

Fig. 10 shows the consensus efficiency compared with existing consensus algorithms which is defined as the ratio of

TABLE IV
COGNITIVE ERRORS

Performance Metrics	Nor w/ DF	Hmali w/o DF	Hmali w/ DF	Lmali w/o DF	Lmali w/ DF
RMSE	1.0113	7.4338	1.2072	5.0855	1.7223
MSE	1.0226	55.2612	1.4573	25.8619	2.9662

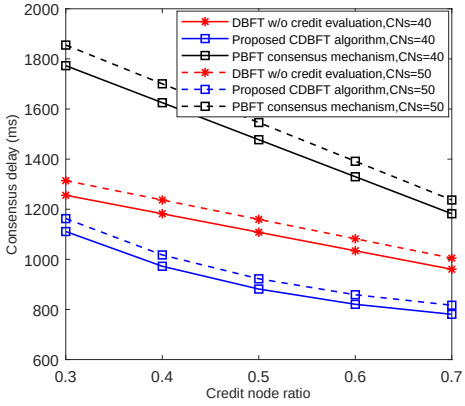


Fig. 9. Consensus delay.

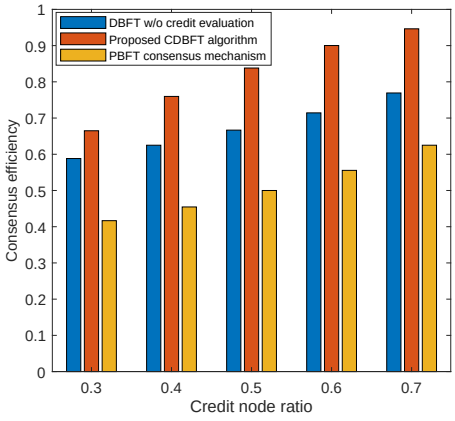


Fig. 10. Consensus efficiency.

blocks that successfully reach consensus to the total number of proposals at each maximum consensus interval. The proposed CDBFT algorithm achieves the highest consensus efficiency. However, PBFT is the lowest which is because all unscreened CNs are involved in the consensus process causes a huge number of validations and attacks to slow down the block consensus process. The increase in credit nodes reduces the probability of malicious attacks, which is why consensus efficiency improves as the ratio of credit nodes increases.

D. Online Multi-step Prediction Algorithm Performance

This subsection demonstrates the convergence and prediction accuracy of the proposed online multi-step prediction algorithm compared with offline DNN [15], LSTM [34] algorithms. The LSTM model is a chain structure with 200 LSTM units, each of which is composed of neural network layers with different activation functions [64]. DNN is a fully connected neural network structure including one input layer, two hidden layer, and one output layer [65]. The number of neural units in each hidden layer is 42, and the number of neural units in the input layer and output layer is determined by the input vector and the output vector. The tansig and purelin activation functions are chosen for the hidden and output layers, respectively. Fig. 11 shows the change of congestion duration versus traffic efficiency according to the proposed cognitive model, which is utilized to verify the effectiveness of the online multi-step prediction algorithm. The results show that the congestion duration decreases gradually as the traffic efficiency increases consistent with the realistic traffic scenarios.

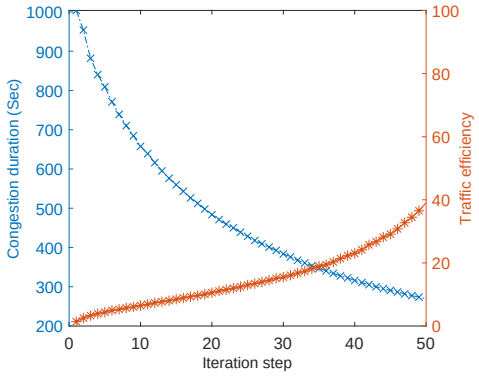
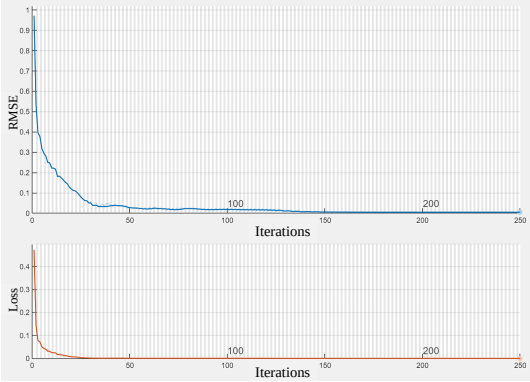


Fig. 11. Changes in traffic congestion duration.

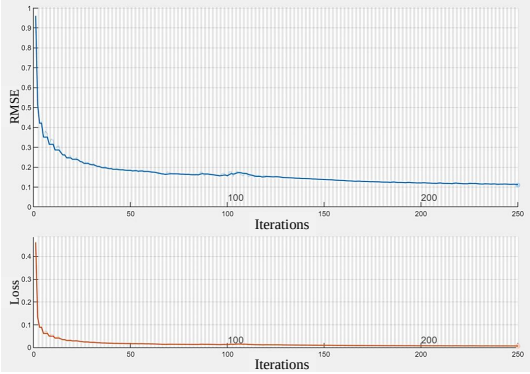
Fig. 12 shows the convergence of the proposed algorithm is faster than traditional methods as well as a lower RMSE is achieved quickly at iteration 150. The reason is that the proposed approach effectively eliminates the interference of anomalous data and acquires higher quality cognitive results for training. To verify the accuracy of the predictions, we compared the multi-step predicted values with the actual values and errors in the following.

Fig. 13 shows the multi-step prediction results and errors of different methods. Fig. 13 (a) shows that the offline LSTM method in which the error between the observed and predicted values increases with the iteration steps. The results are prone to error accumulation, which affects the prediction accuracy and is not effective for long multi-step predictions. The reason is that the offline method is unable to obtain observations and cognition of the current traffic state for future multi-step predictions. Fig. 13 (b) demonstrates that the proposed online prediction algorithm has the lowest error and obtains the highest prediction accuracy of multi-step forecasts. The errors between the actual and predicted values are significantly reduced compared with other methods. The reason behind is that the proposed algorithm can track changes of realistic traffic status from the online cognitive systems to eliminate accumulated error.

Fig. 13 (c) illustrates the offline DNN method has higher errors compared to the other two methods. We can find that the prediction accuracy is the lowest and has great volatility. The

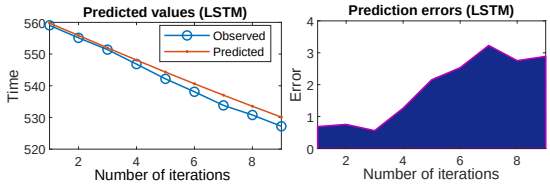


(a) Proposed online forecasts

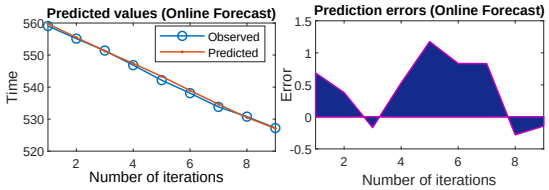


(b) Traditional offline forecasts

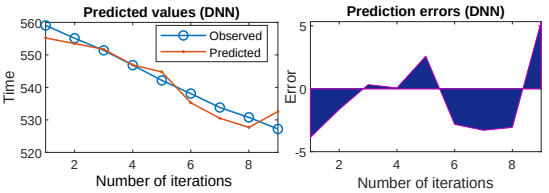
Fig. 12. Comparison of convergence.



(a) Multi-step offline prediction by LSTM



(b) The proposed online multi-step prediction



(c) Multi-step offline prediction by DNN

Fig. 13. Prediction accuracy under different methods.

reason is that offline DNN cannot eliminate cumulative errors in multi-step prediction and cannot obtain the correlations for long time series. Fig. 14 demonstrate the comparison

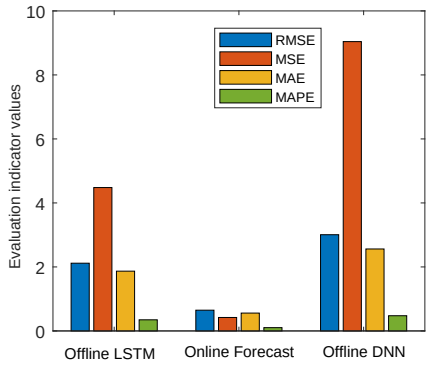


Fig. 14. Performance evaluation.

TABLE V
PREDICTIVE PERFORMANCE

Performance metrics	Methods		
	Offline DNN	Offline LSTM	Proposed algorithm
RMSE	3.00683	2.11675	0.648485
MSE	9.041	4.48062	0.420532
MAE	2.56112	1.8679	0.55809
MAPE	0.475033	0.347646	0.102783

of predicted performance indicators and Table V lists the performance indicator values of different methods. The results show that the proposed online multi-step prediction algorithm has the lowest errors in all performance metrics and achieves a noticeable improvement in prediction accuracy.

VII. Conclusions and Future Works

In this paper, we proposed a blockchain-enabled cognitive segments sharing framework to accomplish credible cognition and support the online multi-step prediction of traffic congestion duration. In the proposed framework, we designed a cognitive model based on anomaly detection and filtering mechanism to carry out local traffic situation perception, which guarantees the accuracy of cognitive segments before concatenating to the blockchain. To resist attacks from malicious nodes and improve the consensus efficiency, we proposed the CDBFT consensus algorithm assisted by the credit evaluation mechanism and P2P network. Last, a decentralized online multi-step traffic congestion duration prediction algorithm was proposed based on the established cognitive sharing system. Simulation results on real-world datasets show that the proposed algorithms effectively improve the consensus delay and the accuracy of multi-step predictions compared to the existing works, the consensus delay and average prediction error are reduced by approximately 35% and 37.3%, respectively.

In future work, we will further investigate the multi-agent collaborative path planning issues for the blockchain-enabled CIoVs. Moreover, we will design incentive mechanisms to build a knowledge sharing and collaborative decision-making platform for blockchain-enabled CIoVs to facilitate secure and efficient collaboration among vehicles.

References

[1] 3GPP, "Digital cellular telecommunications system; Universal Mobile Telecommunications System (UMTS); LTE; 5G," 3rd Generation

Partnership Project (3GPP), Technical Specification (TS) 21.916, 01 2022, version 16.1.0. [Online]. Available: https://www.3gpp.org/release-16

[2] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015.

[3] W. Zhang, "5G vehicular communication technology," TIAA & FuTURE Forum Joint Working Group, Tech. Rep., Oct. 2017.

[4] K. Lin, C. Li, G. Fortino, and J. J. Rodrigues, "Vehicle route selection based on game evolution in social internet of vehicles," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2423–2430, 2018.

[5] Z. Lv, Y. Li, H. Feng, and H. Lv, "Deep learning for security in digital twins of cooperative intelligent transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2021.

[6] Y. Qian, Y. Jiang, L. Hu, M. S. Hossain, M. Alrashoud, and M. Al-Hammadi, "Blockchain-based privacy-aware content caching in cognitive internet of vehicles," *IEEE Network*, vol. 34, no. 2, pp. 46–51, 2020.

[7] H. Lu, Q. Liu, D. Tian, Y. Li, H. Kim, and S. Serikawa, "The cognitive internet of vehicles for autonomous driving," *IEEE Network*, vol. 33, no. 3, pp. 65–73, 2019.

[8] K. F. Hasan, T. Kaur, M. M. Hasan, and Y. Feng, "Cognitive internet of vehicles: motivation, layered architecture and security issues," in *2019 International Conference on Sustainable Technologies for Industry 4.0 (STI)*. IEEE, 2019, pp. 1–6.

[9] J. Kang, R. Yu, X. Huang, M. Wu, S. Maharjan, S. Xie, and Y. Zhang, "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4660–4670, 2019.

[10] Z. Zhou, M. Wang, J. Huang, S. Lin, and Z. Lv, "Blockchain in big data security for intelligent transportation with 6g," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2021.

[11] M. B. Mollah, J. Zhao, D. Niyato, Y. L. Guan, C. Yuen, S. Sun, K.-Y. Lam, and L. H. Koh, "Blockchain for the internet of vehicles towards intelligent transportation systems: A survey," *IEEE Internet of Things Journal*, pp. 4157–4185, 2020.

[12] C. Wang, X. Cheng, J. Li, Y. He, and K. Xiao, "A survey: applications of blockchain in the internet of vehicles," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, pp. 1–16, 2021.

[13] S. K. Dwivedi, R. Amin, S. Vollala, and R. Chaudhry, "Blockchain-based secured event-information sharing protocol in internet of vehicles for smart cities," *Computers & Electrical Engineering*, vol. 86, p. 106719, 2020.

[14] X. Jiang, F. R. Yu, T. Song, Z. Ma, Y. Song, and D. Zhu, "Blockchain-enabled cross-domain object detection for autonomous driving: A model sharing approach," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 3681–3692, 2020.

[15] V. Hassija, V. Gupta, S. Garg, and V. Chamola, "Traffic jam probability estimation based on blockchain and deep neural networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 3919–3928, 2021.

[16] G. Sun, M. Dai, J. Sun, and H. Yu, "Voting-based decentralized consensus design for improving the efficiency and security of consortium blockchain," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6257–6272, 2021.

[17] Z. Yang, R. Wang, D. Wu, B. Yang, and P. Zhang, "Blockchain-enabled trust management model for the internet of vehicles," *IEEE Internet of Things Journal*, pp. 1–11, 2021.

[18] J. Cui, F. Ouyang, Z. Ying, L. Wei, and H. Zhong, "Secure and efficient data sharing among vehicles based on consortium blockchain," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2021.

[19] N. Chiabaut and R. Faitout, "Traffic congestion and travel time prediction based on historical congestion maps and identification of consensual days," *Transportation Research Part C: Emerging Technologies*, vol. 124, p. 102920, 2021.

[20] B.-E. Soussi Niaimi, M. Bouhorma, and H. Zili, "The evolution of the traffic congestion prediction and ai application," *Networking, Intelligent Systems and Security*, pp. 19–31, 2022.

[21] M. Z. Mehdi, H. M. Kammoun, N. G. Benayed, D. Sellami, and A. D. Masmoudi, "Entropy-based traffic flow labeling for cnn-based traffic congestion prediction from meta-parameters," *IEEE Access*, vol. 10, pp. 16 123–16 133, 2022.

[22] Y. Tu, S. Lin, J. Qiao, and B. Liu, "Deep traffic congestion prediction model based on road segment grouping," *Applied Intelligence*, vol. 51, no. 11, pp. 8519–8541, 2021.

[23] M. Bai, Y. Lin, M. Ma, P. Wang, and L. Duan, "Prepct: Traffic congestion prediction in smart cities with relative position congestion tensor," *Neurocomputing*, vol. 444, pp. 147–157, 2021.

[24] A. M. Kishk, M. Badawy, H. A. Ali, and A. I. Saleh, "A new traffic congestion prediction strategy (tcps) based on edge computing," *Cluster Computing*, vol. 25, no. 1, pp. 49–75, 2022.

[25] F. Ayaz, Z. Sheng, D. Tian, and Y. L. Guan, "A proof-of-quality-factor (poqf)-based blockchain and edge computing for vehicular message dissemination," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2468–2482, 2021.

[26] M. Gawas, H. Patil, and S. S. Govekar, "An integrative approach for secure data sharing in vehicular edge computing using blockchain," *Peer-to-Peer Networking and Applications*, vol. 14, no. 5, pp. 2840–2857, 2021.

[27] J. Kang, R. Yu, X. Huang, M. Wu, S. Maharjan, S. Xie, and Y. Zhang, "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4660–4670, 2019.

[28] V. Sharma, "An energy-efficient transaction model for the blockchain-enabled internet of vehicles (iov)," *IEEE Communications Letters*, vol. 23, no. 2, pp. 246–249, 2018.

[29] Y. Wang, Z. Su, K. Zhang, and A. Benslimane, "Challenges and solutions in autonomous driving: A blockchain approach," *IEEE Network*, vol. 34, no. 4, pp. 218–226, 2020.

[30] T. Jiang, H. Fang, and H. Wang, "Blockchain-based internet of vehicles: Distributed network architecture and performance analysis," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4640–4649, 2018.

[31] L. Cheng, J. Liu, G. Xu, Z. Zhang, H. Wang, H.-N. Dai, Y. Wu, and W. Wang, "Sctsc: A semicentralized traffic signal control mode with attribute-based blockchain in iovs," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 6, pp. 1373–1385, 2019.

[32] F. Ayaz, Z. Sheng, D. Tian, G. Y. Liang, and V. Leung, "A voting blockchain based message dissemination in vehicular ad-hoc networks (vanets)," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–6.

[33] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 922–929.

[34] J. Zheng and M. Huang, "Traffic flow forecast through time series analysis based on deep learning," *IEEE Access*, vol. 8, pp. 82 562–82 570, 2020.

[35] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, "T-gcn: A temporal graph convolutional network for traffic prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3848–3858, 2019.

[36] G. Dai, C. Ma, and X. Xu, "Short-term traffic flow prediction method for urban road sections based on space–time analysis and gru," *IEEE Access*, vol. 7, pp. 143 025–143 035, 2019.

[37] Y. Gu, W. Lu, X. Xu, L. Qin, Z. Shao, and H. Zhang, "An improved bayesian combination model for short-term traffic prediction with deep learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 1332–1342, 2019.

[38] F. Zhou, Q. Yang, K. Zhang, G. Trajcevski, T. Zhong, and A. Khokhar, "Reinforced spatiotemporal attentive graph neural networks for traffic forecasting," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6414–6428, 2020.

[39] Y. Jin, E. Tan, L. Li, G. Wang, J. Wang, and P. Wang, "Hybrid traffic forecasting model with fusion of multiple spatial toll collection data and remote microwave sensor data," *IEEE Access*, vol. 6, pp. 79 211–79 221, 2018.

[40] M. M. Chowdhury, M. Hasan, S. Safait, D. Chaki, and J. Uddin, "A traffic congestion forecasting model using cmtf and machine learning," in *2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*. IEEE, 2018, pp. 357–362.

[41] Y. Song, Y. Fu, F. R. Yu, and L. Zhou, "Blockchain-enabled internet of vehicles with cooperative positioning: A deep neural network approach," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3485–3498, 2020.

[42] Y. Fu, F. R. Yu, C. Li, T. H. Luan, and Y. Zhang, "Vehicular blockchain-based collective learning for connected and autonomous vehicles," *IEEE Wireless Communications*, vol. 27, no. 2, pp. 197–203, 2020.

[43] Y. Fu, C. Li, F. R. Yu, T. H. Luan, and Y. Zhang, "An autonomous lane-changing system with knowledge accumulation and transfer assisted by vehicular blockchain," *IEEE Internet of Things Journal*, vol. 7, no. 11, pp. 11 123–11 136, 2020.

[44] S. R. Pokhrel and J. Choi, "Federated learning with blockchain for autonomous vehicles: Analysis and design challenges," *IEEE Transactions on Communications*, vol. 68, no. 8, pp. 4734–4746, 2020.

[45] Y. Wang, A. Zhang, P. Zhang, Y. Qu, and S. Yu, "Security-aware and privacy-preserving personal health record sharing using consortium blockchain," *IEEE Internet of Things Journal*, pp. 1–15, 2021.

[46] Y. Yahiatene, A. Rachedi, M. A. Riahla, D. E. Menacer, and F. Nait-Abdesselam, "A blockchain-based framework to secure vehicular social networks," *Transactions on emerging telecommunications technologies*, vol. 30, no. 8, p. e3650, 2019.

[47] F. R. Yu, J. Liu, Y. He, P. Si, and Y. Zhang, "Virtualization for distributed ledger technology (vdlt)," *IEEE Access*, vol. 6, pp. 25 019–25 028, 2018.

[48] Y. Liu, F. R. Yu, X. Li, H. Ji, and V. C. Leung, "Decentralized resource allocation for video transcoding and delivery in blockchain-based system with mobile edge computing," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 11 169–11 185, 2019.

[49] J. Huang, L. Kong, G. Chen, L. Cheng, K. Wu, and X. Liu, "B-iot: Blockchain driven internet of things with credit-based consensus mechanism," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019, pp. 1348–1357.

[50] Y. Wang, Z. Su, and N. Zhang, "Bsis: Blockchain-based secure incentive scheme for energy delivery in vehicular energy network," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3620–3631, 2019.

[51] Y. Liu, K. Wang, Y. Lin, and W. Xu, "A lightweight blockchain system for industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3571–3581, 2019.

[52] J. Huang, L. Kong, G. Chen, M.-Y. Wu, X. Liu, and P. Zeng, "Towards secure industrial iot: Blockchain system with credit-based consensus mechanism," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3680–3689, 2019.

[53] C. Li, Q. Luo, G. Mao, M. Sheng, and J. Li, "Vehicle-mounted base station for connected and autonomous vehicles: Opportunities and challenges," *IEEE Wireless Communications*, vol. 26, no. 4, pp. 30–36, 2019.

[54] C. Li, Y. Fu, F. R. Yu, T. H. Luan, and Y. Zhang, "Vehicle position correction: A vehicular blockchain networks-based gps error sharing framework," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 2, pp. 898–912, 2020.

[55] T. Cai, H. Cai, H. Wang, X. Cheng, and L. Wang, "Analysis of blockchain system with token-based bookkeeping method," *IEEE Access*, vol. 7, pp. 50 823–50 832, 2019.

[56] M. A. Mondal and Z. Rehena, "Identifying traffic congestion pattern using k-means clustering technique," in *2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU)*. IEEE, 2019, pp. 1–5.

[57] T. Li, A. Ni, C. Zhang, G. Xiao, and L. Gao, "Short-term traffic congestion prediction with conv–bilstm considering spatio-temporal features," *IET Intelligent Transport Systems*, vol. 14, no. 14, pp. 1978–1986, 2021.

[58] Y. Wang, S. Cai, C. Lin, Z. Chen, T. Wang, Z. Gao, and C. Zhou, "Study of blockchains's consensus mechanism based on credit," *IEEE Access*, vol. 7, pp. 10 224–10 231, 2019.

[59] F. Kandah, B. Huber, A. Skjellum, and A. Altarawneh, "A blockchain-based trust management approach for connected autonomous vehicles in smart cities," in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, 2019, pp. 0544–0549.

[60] J. Zhang, Y. Rong, J. Cao, C. Rong, J. Bian, and W. Wu, "Dbft: A byzantine fault tolerance protocol with graceful performance degradation," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–14, 2021.

[61] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2016.

[62] Y. Ma, Z. Zhang, and A. Ihler, "Multi-lane short-term traffic forecasting with convolutional lstm network," *IEEE Access*, vol. 8, pp. 34 629–34 643, 2020.

[63] M. Li and Z. Zhu, "Spatial-temporal fusion graph neural networks for traffic flow forecasting," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 5, 2021, pp. 4189–4196.

[64] C. Ma, G. Dai, and J. Zhou, "Short-term traffic flow prediction for urban road sections based on time series analysis and lstm_bilstm method," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 5615–5624, 2022.

[65] D.-J. Lin, M.-Y. Chen, H.-S. Chiang, and P. K. Sharma, "Intelligent traffic accident prediction model for internet of vehicles with deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 2340–2349, 2022.