Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606

Volume 07 Issue 01
2024

Research Article
https://doi.org/10.56979/701/2024

Blockchain-Based Decentralized Federated Learning for Privacy-Preserving Traffic Flow Prediction: A Case Study with PeMS-8 Data

Shaharyar Asad[1], Tehreem Masood[1,2], Shamim Akhter[3], Muhammad Naushad Ghazanfar[3], Sumbul Azeem [4], Iftikhar Naseer[1,2], and Hafiz Muhammad Tayyab Khushi[1]

[1]Faculty of Computer Science & Information Technology, Superior University, Lahore 54000, Pakistan.
[2]Department of Software Engineering, Superior University, Lahore, 54000, Pakistan.
[3]School of Information Management, Minhaj University, Lahore , 54000, Pakistan.
[4]Lahore College for Women University, Jail Road, Lahore, 54000, Pakistan.
[*]Corresponding Author: Hafiz Muhammad Tayyab Khushi. Email: muhammad.tayyab@superior.edu.pk

Received: March 01, 2024 **Accepted:** May 21, 2024 **Published:** June 01, 2024

Abstract: A timely traffic flow information is essential for traffic management, traffic prediction has become a key component of intelligent transportation systems. However, current centralized machine learning-based traffic flow prediction algorithms need the collection of raw data for the train model, which poses significant privacy breach hazards. Federated learning, a recent innovation that effectively protects privacy by sharing model changes without transferring raw data, has been developed to solve these issues. The current federated learning frameworks, however, are built on a centralized model coordinator that continues to experience serious security issues, such as a single point of failure. In this paper, we proposed BDFL a block-chained decentralized federated-learning (DFL) architecture for traffic flow prediction using PeMS-8 data. The suggested technique provides decentralized model training on PeMS-8 while ensuring the privacy of the underlying data. The long short-term memory (LSTM) model, which is often employed for signal and time-series data, was used in this study. A total of 3035520 traffic locations were covered by the PeMS-8 dataset, which was obtained from the Caltrans Performance Measurement System (PeMS). These locations' data included timestep and junction details as well as traffic flow, occupancy, and speed. The total accuracy of our long-short-term memory (LSTM) model is 99.0, with a loss of 3.296.

Keywords: Federated learning; Decentralized Federated Learning; Traffic flow; Blockchain; LSTM.

1. Introduction

To reduce traffic congestion and its detrimental implications (such as inaccurate travel time estimates, increased fuel consumption, and negative environmental effects), a traffic forecast is essential. Traffic congestion has recently become a major social issue as vehicle miles traveled (VMT) keep increasing yearly. For instance, according to the 2019 Urban Mobility Report, the United States' traffic jams added Eighty-eight billion hours to commute time and Three billion gallons to fuel usage just in 2019. As a result, real-time traffic prediction can reduce congestion by giving companies and road users accurate real-time traffic conditions for route planning and providing rerouting choices once the cars are on the road.

Deep learning (DL) has recently gained popularity as a method for traffic flow prediction (TFP), showing great promise in the literature [3] and achieving prediction accuracy of up to 93%. Existing DL models, on the other hand, are centrally trained in the context of the TFP problem, necessitating the collection and aggregation of enormous amounts of data by a data center or the cloud for processing. If all of the obtained data must be explicitly shared to create the model, or when crowdsourcing approaches are employed, it becomes challenging to adhere to the data privacy laws [5].

We suggest a Blockchain-based decentralized federated learning (BDFL)-based TFP technique to overcome the issue. According to [6], BDFL is a cutting-edge machine-learning framework that enables participants to collectively train a model that is shared globally. This model is then distributed by a decentralized server. During each iteration of the global model training, each participant is given an initial global model. Afterward, they train a model at the local level and subsequently send updates of the local model (namely, gradient parameters) to the decentralized server, without uploading the raw data. The decentralized server then compiles all of the local model modifications into a new global model before releasing it. Our global DFL (Model) training approach described is carried out repeatedly until convergence through local TFP task execution, BDFL application can effectively resolve privacy concerns for cars. On the other hand, because conventional BDFL mostly relies on the decentralized server can collect and Handel locally model updates, the presence of a single point of failure in security systems is an unavoidable vulnerability [7]. Additionally, if malicious cars submit fake or subpar content. Additionally, BDFL eliminates the requirement to exchange or aggregate locally obtained traffic data, improving stakeholder (e.g., data collector) communication efficiency, privacy, and security.

Consensus procedures are used by blockchain, a distributed and decentralized public ledger system, to synchronize P2P network changes without the need for external parties or centralized organizations. Blockchain's incredible features of decentralization, inalterability, traceability, and anonymity have led to its widespread application in many different industries [10, 11]. While doing so, we also considered tracking and managing all local model modifications and addressing the single-point-of-failure issue by using a decentralized blockchain instead of a centralized server. This effort aims to use blockchain with federated learning (FL) in the TFP system to enhance privacy protection. To be more precise, we create a collaborative blockchain for a decentralized federated learning-based traffic flow prediction (TFP) system. We utilize a compact yet highly efficient neural network model termed Long Short-Term Memory (LSTM), as described in reference [12], to demonstrate the implementation of federated learning for TFP by analyzing time-series traffic data. The subsequent sections of the paper are organized as follows. Section II focuses on the existing research conducted on BDFL privacy studies and short-term TFP. In Section III, a security parameter aggregation approach is suggested along with a definition of the Decentralized blockchain Federated learning and Dataset. The LSTM algorithm and experiment results are explained in section IV. The Conclusion is explained in section V.

2. Literature Work

To predict traffic flow, Y. Liu describes the FedGRU method in this study, which is based on the privacy-preserving machine learning methodology known as federated learning. FedGRU updates universal learning models via a safe parameter aggregation approach as opposed to simply sending raw data to organizations, which is how it differentiates from other centralized learning systems. Y. Liu employs a Federated Averaging technique in the secure parameter aggregation mechanism to lower the cost of communication while transmitting the model parameters. Y. Liu also creates a Joint Announcement Protocol to boost FedGRU's scalability. Y. Liu also provides an alternative to the FedGRU algorithm for estimating traffic flow by clustering organizations: an ensemble clustering-based approach. Research studies performed using a real dataset reveal that FedGRU is capable of delivering rapid and precise traffic prediction while upholding the confidentiality and safety of raw data. Furthermore, its forecast accuracy surpasses that of intricate deep-learning models by 90.96 percent. To more accurately forecast future traffic flow data and better capture the spatial-temporal relationship among the data, they will apply the Graph Convolutional Network (GCN) in the federated learning architecture.

In this article, Liu, Yi suggests FedGRU, a design for a gated recurrent unit neural network based on federated learning. (TFP). To manage the communication cost during parameter transmission, Liu, Yi apply a Federated Averaging technique in the secure parameter aggregation mechanism. The effectiveness of the FedGRU model in generating precise and timely traffic estimates while maintaining privacy compliance is proven through thorough case studies conducted using the Performance Measurement System (PeMS) dataset. We describe a novel privacy-preserving method for predicting traffic flow that blends federated learning development with an operational GRU neural network. A locally trained model and the absence of raw data exchange in this method guarantee trustworthy data privacy protection. Liu, Yi introduce the Federated Averaging (FedAVG) technique, which implements Stochastic Gradient Descent, in the safe parameter aggregation method (SGD).

In this paper, Zhang, Chenhan, et al provide a distinctive federated learning model. Zhang, Chenhan, et al. present a differential privacy-based adjacency matrix preservation technique that is specifically designed to safeguard topological data. Zhang, Chenhan, et al. also offer an adjacency matrix aggregation method that enables regional Utilize gnn-Based model to integrate into the global network for enhanced training. In addition, Zhang, Chenhan, et al. define attention-based spatial-temporal graph neural networks and an ASTGNN model to forecast traffic speed. Zhang, Chenhan, and his colleagues utilize the proposed federated learning system, known as FASTGNN, which incorporates ASTGNN, to predict traffic speeds. Multiple empirical analyses of a dataset from the actual world provide evidence that FASTGNN can generate precise predictions while maintaining confidentiality. Following that, ablation research is carried out to assess the key components of FASTGNN. Zhang, Chenhan, et al want to focus on further fine-grained research on FASTGNN in the future, including communication overhead and generalization capabilities on other data sets.

In this paper Wang, Naiyu, et al present BPFL, a Blockchain-based Privacy-Preserving Federated Learning scheme that, in this study, they uses blockchain as the foundational distributed architecture of FL. They also developed an incentive system based on reputation. To show that the recommended system can satisfy the security requirements while simultaneously enhancing the performance of the FL model, it is put through a security analysis and performance evaluation. Wang, Naiyu, et al assess BPFL's performance in three areas: model correctness, verification computation cost, and the effectiveness of the incentive system. The Hyper Ledger Fabric (v-1.0) platform is used to create the blockchain. Wang, Naiyu, et al run simulations in a single channel to keep things simple. The smart contracts are written in Go (v-1.10.3) and its accompanying libraries. Our future study will be focused on communication challenges caused by vehicle dynamics, which primarily consist of two pacts: 1) Create an asynchronous federated learning model that takes into account the straggler issue. 2) To optimize the flexibility of the proposed system in the IoV, design a flexible client adaptive selection technique that considers the previously specified communication overhead and latency limits. This article [5] provides an outline of the underlying ideas as well as an examination of the FL chain's potential in MEC networks. Communication costs, resource allocation, incentive systems, security, and privacy protection are important design considerations for FL chains. The essential answers for FL chain design are presented, as well as the lessons gained and prospects. Finally, the importance of major research issues and future perspectives are discussed. An overview of FLchain, a developing paradigm in MEC enabled by the convergence of FL and blockchain, has been provided.

The research [6] demonstrates a bias in the aggregation ratio towards fast nodes, and it highlights the challenge of removing old weights of slow nodes quickly in asynchronous federated learning based on exponential moving averages. This poses a significant difficulty in practical applications. The algorithm has been examined on the MNIST and CIFAR-100 datasets. By addressing the problem of slower model convergence time brought on by variations in node training speed, this study increases the effectiveness of asynchronous federated learning and training. Furthermore, the blockchain-enabled infrastructure is capable of efficiently dealing with some hostile threats. Chai wants to investigate the overhead and transaction throughput of the proposed hierarchical blockchain system in a real-world IoV scenario in the future.

Shahriar Badsha provides a lightweight vehicular blockchain architecture to increase trust, verifiability, and non-repudiation in networked vehicular collaboration. Shahriar Badsha created a vehicle model-sharing architecture based on blockchain. For automotive networks, a new deterministic PoVS-BFT is suggested as a scalable and effective consensus approach. Additionally, they built a practical two-step verification mechanism and addressed transaction verification concerns with model sharing. This Blockchain’s design can be modified for a variety of uses, including large-scale automotive networks. Future work will involve extending the architecture to make the consensus mechanism leaderless while maintaining the qualities of determinism, scalability, and linear complexity.

In this paper [8], the authors present a blockchain-based federated learning approach for smart healthcare, where the edge nodes maintain the blockchain and the MIoT devices use the federated learning to fully utilize the distributed clinical data. To protect data privacy and thwart data poisoning attempts, they explicitly develop a gradient verification-based consensus protocol and an adaptive differential privacy approach. They contrast our approach with two other related techniques using a diabetic dataset obtained from real-world data. Promising experimental results show that our approach can effectively lower the privacy budget consumption, fend off attempts to poison it and achieve high model accuracy in a tolerable operating time.

To transport data reliably and securely for the Internet of Things (ECC), the paper proposes a novel approach called Secure Ant Colony Optimization with Multi Kernel Support Vector Machine (ACOMKSVM) and Elliptical Curve Cryptosystem [9]. This program employs cutting-edge technology to create safe ACOMKSVM training algorithms by utilizing partial views of IoT data acquired from many data providers. Furthermore, it utilizes blockchain technology to ensure the privacy and precision of certain data. The ACOMKSVM safe learning process, which produces precise and effective privacy, is then safeguarded by ECC The authors developed a safe and dependable framework for data interchange across several data sources using the Internet of Things data in this study. Using a blockchain, the Internet of Things data was encrypted and entered into a distributed ledger. Two benchmark datasets from the UCI AI repository, the Heart Disease Data Set (HDD) and the Breast Cancer Wisconsin Data Set (BCWD), are used to evaluate the

Table 1. Comparison of Proposed State of the art approaches

Title and Author	Dataset	Model Details	Accuracy
Privacy-preserving traffic flow prediction: A federated learning approach [1]	PeMS database	FedGRU algorithm	90.96
FedGRU: Privacy-preserving traffic flow prediction via federated learning [2]	(PeMS) dataset	FedGRU model	99.32
A topological information protected federated learning approach for traffic speed forecasting [3]	PeMSD7	GNN model	95.6
A blockchain based privacy-preserving federated learning scheme for Internet of Vehicles [4]	Mnist Belgium Emnist	BPFL	70%, 70%/90%
Federated learning meets blockchain in edge computing: Opportunities and challenges [5]	FLchain		88%
A hierarchical blockchain-enabled federated learning algorithm for knowledge sharing in internet of vehicles [6]	MNIST, CIFAR10	ADMM-based algorithm	97
A light-weight blockchain architecture for v2v knowledge sharing at vehicular edges [7]	California state highways	PoVS-BFT,	87.6%
A blockchain-based federated learning method for smart healthcare [8]	diabetes dataset	CNN	78.5% and 82.7%,

Privacy preserving blockchain technique to achieve secure and reliable sharing of IoT data [9]	Breast Cancer Wisconsin Data Set (BCWD) and Heart Disease Data Set (HDD)	SVM classifiers	90.65% and 92.47

3. Materials AND Methods

In this section, we developed a DBFL framework using the Tensorflow framework and provided a blockchain verification platform using the consortium blockchain. To assess the effectiveness of the suggested methodology, we use data from. The Caltrans performance measurement system is a tool used by California Highways to assess performance. The dataset including traffic flow data collected in California from January to March 2013 is a component of the Caltrans performance monitoring system. We selected the first two months of data for training and the final month of data for testing. The method's general model architecture is depicted in Figure 1.

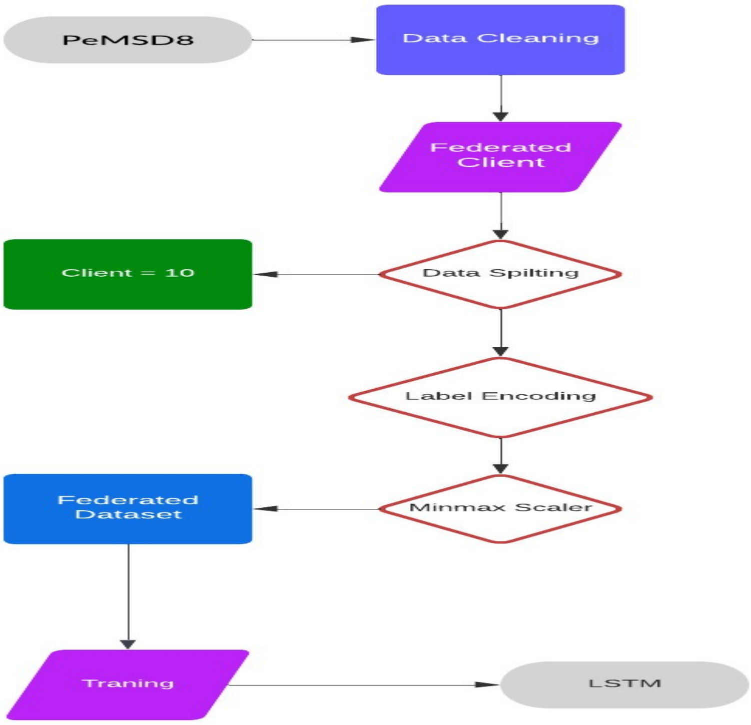

Figure 1. Research Methodology

3.1. Dataset

In this paper, The Caltrans Performance Measurement System of the California Department of Transportation PeMS-8 dataset is used. Each time series in the three data files (PeMS-4W, PeMS-8W, and PeMS-12W) contains 288 (5-minute) time points per day for the first four weeks (PeMS-4W), first seven weeks (PeMS-7W), first eight weeks (PeMS-8W), and first twelve weeks (PeMS-12W) of 2018, respectively [19]. This PeMS-8 dataset contains the traffic data from July and August of 2016 in San Bernardino, California. There are 170 detectors spread across 8 roadways at 5-minute intervals. As a standard dataset for traffic forecasting, the PeMS-8 dataset is widely used. Over 39,000 detectors have been installed by the system on the highways in California's biggest cities. The databases contain geographic data about the sensor stations. In our trials, we take into account three different types of traffic measurements: total flow, average speed, and average occupancy.

3.2. Preprocessing

3.2.1. Data Cleaning

The PeMS-8 dataset is in nps formatted so firstly I converted it into a CSV file using the pandas library. After the conversion of the file, I check the missing value from the dataset to achieve high accuracy during the model training. In the below table, I will show the five rows of the dataset.

3.2.2. Data Splitting

The dataset may have been split into 90% for training and 10% for testing using a machine learning sample method. To tackle this issue, we partitioned the training dataset into training and validation, with 90% of the dataset being utilized for training and 10% for validation [53]. Fig 2 shows the snore of data splitting.

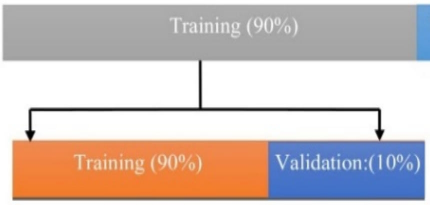

Figure 2. Shown the data training and testing ratio

3.2.3. Data Normalization

In our PeMS-8 dataset, the value attribute value is redundant so we need the scale of over dataset. We used the MinMax Scale machine learning method for scaling the dataset. So the MinMax scaler scales the dataset value between 0 and 1

3.3. Federated Client

The dataset is split up across several clients in decentralized federated learning, with each client retaining a subset of the data locally. Data encryption and differential privacy are two privacy-preserving methods that can be used to protect client data while it is being used in the Federated Learning process. Some clients could choose not to actively engage in every training session. A subset of clients can be selected for each training round using subset selection procedures. This subset can be chosen based on variables like client performance, network conditions, or device availability. Based on the dataset's total length, the decentralized federated learning divides the dataset into 10 clients in the first step. Each customer has a dataset, which is equally distributed among 10 clients. There are 30 total datasets, and they are shared across 10 clients. Fig 3 displays the entire federated learning client dataset. The dataset can be distributed among clients as shown in the following Figure 3.

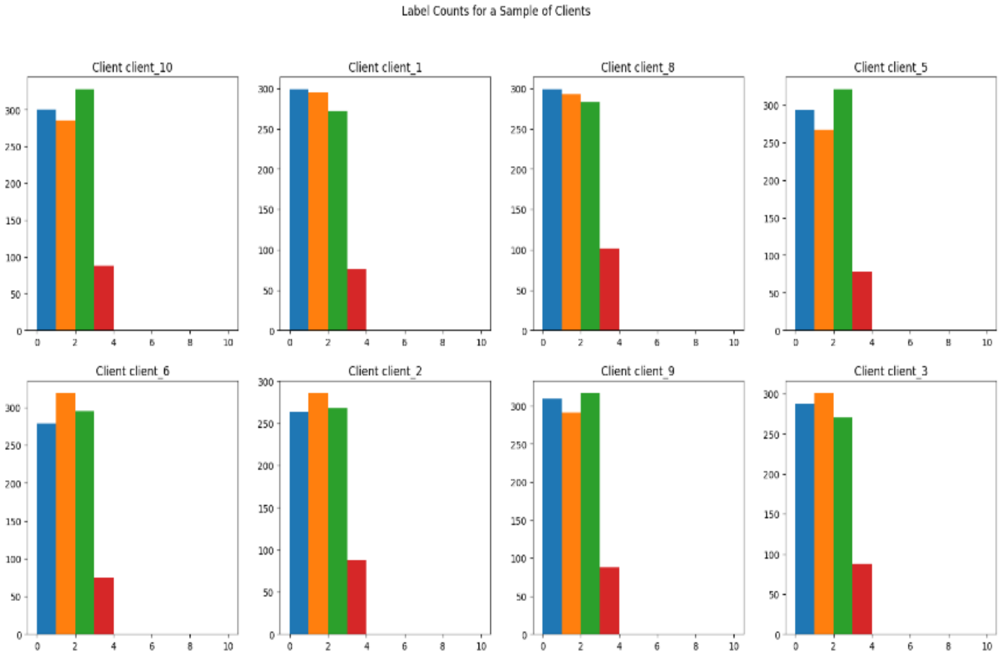

Figure 3. Federated Client Data

3.4. Federated Dataset

We must first turn the federated dataset into a federated dataset after the dataset has been pre-processed. Using the Tensor Flow library, the label and features are reshaped during the pre-processing. Based on client data and client id, this federated dataset can then be split into federated train and federated test. The lstm model is trained using the federated data we collected. In our dataset, 2 clients are utilized for testing, while 8 clients are used for training.

Table 2. Number of Client for Train and Testing

Number of client datasets	Number of client datasets
8	2

3.5. Federated Dataset

To predict traffic flow, I created decentralized federated learning utilizing blockchain. The Caltrans Performance Measurement System of the California Department of Transportation provided the dataset for this thesis. The two main libraries we used for our implementation of the block-chained FL extended from the Block FL [8] architecture are hashlib and Tensor Flow federated learning. We use Tensor Flow federated learning tensors as the data structure to house our data sample since it provides APIs for efficiently carrying out matrix operations like addition and multiplication while computing gradients and carrying out weights updates. The nodes that accept uploaded gradients from worker nodes, validate them, and place them in a candidate block are referred to as role miners. They then mine this block and request that its worker associates download it to obtain the gradients stored in it. The block is then given access to all of the train weights. After completing this training, the local model was aggregated with the global model to estimate traffic flow, as seen in the diagram below [7].

We made the data available to PeMS-8 for this work. The global model is updated with the data from each of California's transport departments using the Caltrans Performance Measurement System. Because the data comes from a range of sources, we have devised a normalization technique to effectively handle the numerous diverse types of data. Upon completion of the data normalization procedure. To train and disseminate a collaborative model, we employ a Federated Learning (FL) architecture that is built upon Blockchain Technology (BCT). The purpose of FL is to integrate the weights of models learned locally. Once the weights of the locally trained models are integrated, the global model is updated with the results [5].

Actions is the second objective. The weights of the local models are combined and then updated in the block chain FL solution that has been presented. Using a technique known as spatial normalization, we create a local model for the heterogeneous or unbalanced data [6].

The proposed model consists of two separate parts. The FL framework based on BCT comes after the local client model. We start by attempting to solve the forecasting of traffic flow. In the last phase, the local model weights are transmitted to the blockchain network so they may be used to train the blockchain governs both the confidentiality and disclosure of personal data. In the blockchain ledger, we document two unique types of transactions: data retrieval transactions (B) and data sharing transactions (A). This study employs a blockchain that requires authorization to access to regulate data accessibility while preserving data privacy. The main benefit of a blockchain is its capacity to log every transaction and provide permissioned blockchain for access to data sourced from a global model. Coordination of data-related global model. In this section of the thesis, we examine a hypothetical scenario in which data across PEMS-8 are exchanged and decentralized. Our approach facilitates the diffusion of the model throughout a decentralized network while making user data concealment easier. Each PEMS-8 decided to give the locally produced model weights. Additionally, FL is used to aggregate the net consequences of the shared models as a whole. This project's major goal is to use FL to communicate data between datasets in a method that doesn't compromise patient privacy. Original records must be encrypted because they are confidential and take up a lot of disk space Due to the limited amount of storage space available, putting data on the blockchain is costly regarding both the required resources for processing and the financial expenditure. As a result, it is easier to acquire trained models when blockchain technology and traffic data are used [6].

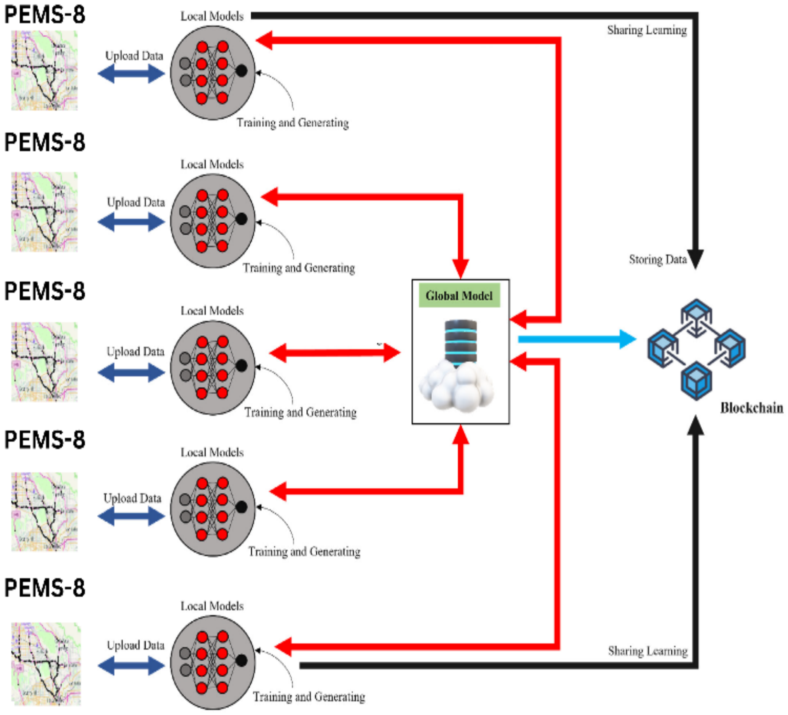

Figure 4. Proposed block

4. Results and Discussions

In this section, we utilize the Tensorflow Federated framework to construct a Federated Learning. We utilize a consortium blockchain to provide a blockchain verification platform within our architecture. We validate the effectiveness of the proposed framework by using the Caltrans performance measurement system dataset gathered from California roadways. For training purposes, we used the traffic flow data from January and February 2013 acquired in California, which is included in the Caltrans performance measurement system dataset. The data from March 2013 was used for testing. We employ the traffic flow data from the previous two months as input to predict the traffic flow in the next month. The simulation is configured with the following values: the number of vehicles (V) is 10, the learning rate (α) is 0.001, the training round (T) is 500, the mini-batch (m) is 128, and the local training epochs (B) is 1.

4.1. Measuring Model Metrics

We evaluated long-short-term memory accuracy using proposed decentralized federated learning and local learning for 10 training rounds, respectively. The model is trained locally on each client via 10 epochs for each round of decentralized federated learning, and model changes are communicated to the decentralized server to create a global model. In the studies, we examined three more metrics in addition to accuracy, such as precision, recall, and F1- score. Recall is the ratio of accurately categorized positive cases to the total number of positively classified occurrences, whereas precision is the proportion of accurately classified positive instances to all positively classified instances. Using both decentralized federated learning and blockchain-based lstm, we evaluated these three criteria. In a 10-round training process, we examined the model's metrics after each round. I created a 10-layer LSTM model, and after each layer was fitted, another Softmax layer was added to help reduce error rates. I used Softmax on the dense layers. To determine the inaccuracy in our training of the lstm model, I build a loss and accuracy structure after defining the lstm model. For loss and accuracy find, I used the SparseCategoricalCrossentropy approach. When creating the FedAVG function in Tensorflow, I had to find both a client and a server optimizer. For the client optimizer, I utilized SGD, which has a learning error rate of 0.02 on the centralized server side, and a learning rate of 0.01 on the decentralized server side. The table below lists the model's 7 training rounds.

4.2. Model Accuracy

The training set may vary between several devices because each device trains the model using its data. The devices receive the updated and improved global model from the server after it has been created. The global model is then updated by each device using its data, potentially enhancing the model's accuracy.

Until the model reaches the appropriate degree of accuracy, the aforementioned process is repeated. Decentralized federated learning may generally increase accuracy and scalability for a variety of use cases by training models locally and aggregating them globally, especially in situations where data is dispersed across numerous devices or locations. Over model shows best accuracy accuracy=0.999 and loss=3.296 due to decentralized global model aggregation on a block basis, which improves privacy and scalability.

4.3. Training Loss

A machine learning model's performance throughout the training process is assessed using the statistic known as training loss. It computes the discrepancy between the predicted outcomes of the model and the real target values obtained from the training dataset. To minimize training loss and maximize prediction accuracy, the model iteratively modifies its parameters throughout training. The type of the problem and the model architecture selected determine the precise shape of the training loss. For instance, the mean squared error (MSE) is frequently used as the training loss in regression problems, but the cross-entropy loss or the binary cross-entropy loss may be employed in classification problems. Averaging the training loss over the whole training set or batch is commonly done after computing it for each training example or batch of instances. Reducing the training loss shows that the model is improving its prediction abilities and is the intended outcome of training. The y-axis of the train loss graph displays the number of rounds that the model performs.

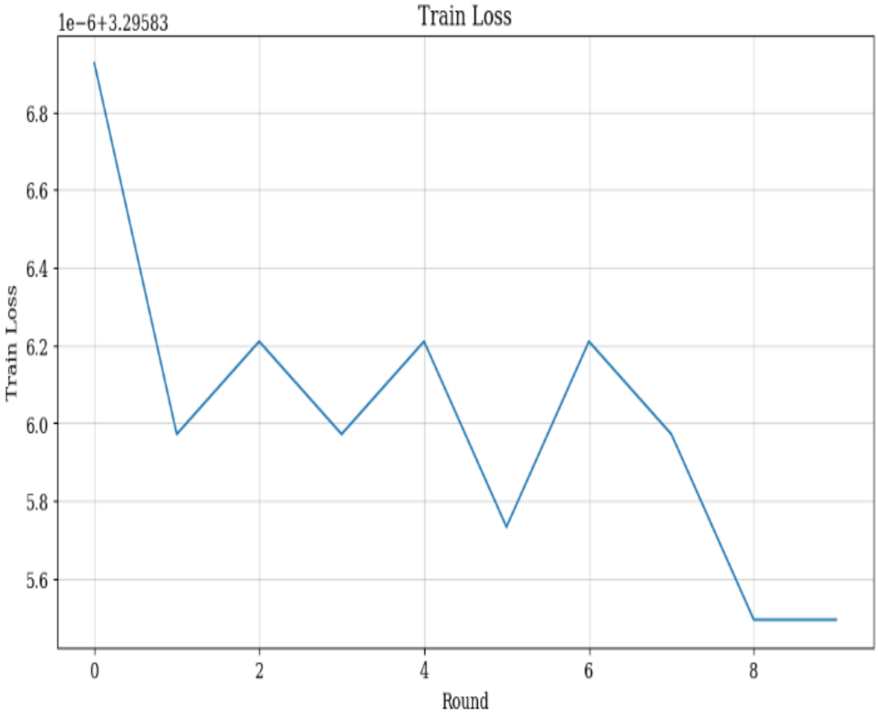

Figure 5. LSTM Model Training loss on the PeMS-8

4.4. Train Accuracy

A training accuracy graph shows how well an LSTM model performs during training. It shows how the model's accuracy changes throughout various training iterations or epochs. A training accuracy graph will have an x-axis for the quantity of training iterations or epochs and a y-axis for training accuracy. Typically, the accuracy is represented as a percentage, with values ranging from 0% to 100%. As the model makes random predictions in the early phases of training, the accuracy may begin at a low value. The model learns from the data as the training goes on and modifies its parameters to increase accuracy. Consequently, with each iteration or epoch, the training accuracy typically increases. The train accuracy graph's y-axis displays the number of rounds the Model performed, while the x-axis displays the accuracy of training on each round.

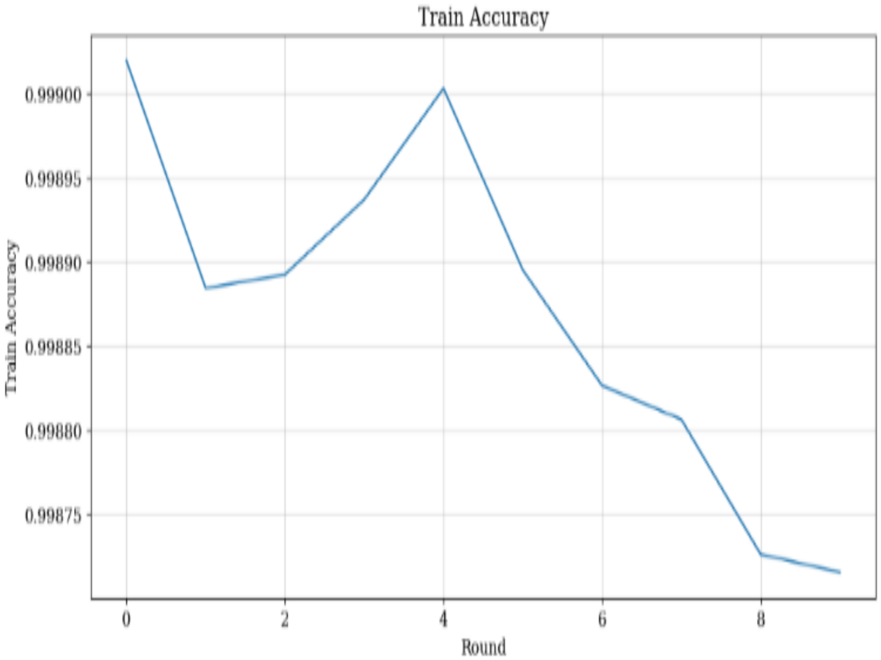

Figure 6. LSTM Model Training Accuracy on the PeMS

5. Conclusions

The study introduces a new method called BDFL, which is an architecture for decentralized federated learning utilizing blockchain technology. This method is specifically designed for predicting traffic flow using PeMS-8 data. This method tackles the privacy issues linked to centralized machine learning-based traffic prediction algorithms by employing federated learning approaches. The suggested design distributes the process of training models, guaranteeing the confidentiality of the data being used. Applying the LSTM model to the PeMS-8 dataset from Caltrans yields encouraging outcomes, boasting a remarkable accuracy rate of 99.0% and a small loss of 3.296. The BDFL architecture utilizes blockchain technology with decentralized federated learning to provide a secure and ensuring privacy-preserving method for predicting traffic flow in intelligent transportation systems.

References

1. Liu, Y., James, J.Q., Kang, J., Niyato, D. and Zhang, S., 2020. Privacy-preserving traffic flow prediction: A federated learning approach. IEEE Internet of Things Journal, 7(8), pp.7751-7763.
2. Liu, Y., Zhang, S., Zhang, C. and James, J.Q., 2020, September. Fedgru: Privacy-preserving traffic flow prediction via federated learning. In 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC) (pp. 1-6). IEEE.
3. Zhang, C., Zhang, S., James, J.Q. and Yu, S., 2021. FASTGNN: A topological information protected federated learning approach for traffic speed forecasting. IEEE Transactions on Industrial Informatics, 17(12), pp.8464-8474.
4. Wang, N., Yang, W., Wang, X., Wu, L., Guan, Z., Du, X. and Guizani, M., 2022. A blockchain based privacy-preserving federated learning scheme for Internet of Vehicles. Digital Communications and Networks.
5. Nguyen, D.C., Ding, M., Pham, Q.V., Pathirana, P.N., Le, L.B., Seneviratne, A., Li, J., Niyato, D. and Poor, H.V., 2021. Federated learning meets blockchain in edge computing: Opportunities and challenges. IEEE Internet of Things Journal, 8(16), pp.12806-12825.
6. Chai, H., Leng, S., Chen, Y. and Zhang, K., 2020. A hierarchical blockchain-enabled federated learning algorithm for knowledge sharing in internet of vehicles. IEEE Transactions on Intelligent Transportation Systems, 22(7), pp.3975-3986.
7. Islam, S., Badsha, S. and Sengupta, S., 2020. A light-weight blockchain architecture for v2v knowledge sharing at vehicular edges. In 2020 IEEE International Smart Cities Conference (ISC2) (pp. 1-8). IEEE.
8. Chang, Y., Fang, C. and Sun, W., 2021. A blockchain-based federated learning method for smart healthcare. Computational Intelligence and Neuroscience, 2021.
9. Le Nguyen, B., Lydia, E.L., Elhoseny, M., Pustokhina, I., Pustokhin, D.A., Selim, M.M., Nguyen, G.N. and Shankar, K., 2020. Privacy preserving blockchain technique to achieve secure and reliable sharing of IoT data. Computers, Materials & Continua, 65(1), pp.87-107.
10. Yanez Parareda E. Federated learning network: Training distributed machine learning models with the federated learning paradigm.
11. Lundberg O. Decentralized machine learning on massive heterogeneous datasets: A thesis about vertical federated learning.
12. Felix Johannes M. Hardened Model Aggregation for Federated Learning backed by Distributed Trust Towards decentralizing Federated Learning using a Blockchain.
13. Li W. PRACTICAL IMPLEMENTATION OF BLOCKCHAIN ENABLED FEDERATED LEARNING (Doctoral dissertation, University of Delaware).
14. Venir E, Ruzza L. Decentralized federated machine learning with blockchain and zero knowledge proofs.
15. Teunissen G. Machine learning for all: a methodology for choosing a federated learning approach.