Federated-Learning-Based Wireless Traffic Prediction

Feng Li[13], Xiaona Chen[1], Kwok-Yan Lam[2], Bowen Shen[2], Li Wang[4]
[1]School of Information and Electronic Engineering, Zhejiang Gongshang University, Hangzhou, China
[2]College of Computing and Data Science, Nanyang Technological University, Singapore
[3]Strategic Centre for Research in Privacy-Preserving Technologies & Systems, Nanyang Technological University, Singapore
[4]College of Marine Electrical Engineering, Dalian Maritime University, Dalian, China
li_feng@ntu.edu.sg, cxn1878289355@163.com, kwokyan.lam@ntu.edu.sg, bowen010@e.ntu.edu.sg, liwang2002@dlmu.edu.cn

***Abstract*—Accurate traffic prediction is crucial for enhancing the performance of intelligent cellular networks, as it directly impacts the effective allocation of network resources and user satisfaction. The instability and dynamics of traffic data pose challenges to centralized prediction methods. These methods are limited in prediction accuracy and may lead to issues concerning data privacy breaches and response delays. Furthermore, traditional Federated Learning (FL) typically employs a simple averaging strategy during model aggregation, which does not consider the heterogeneity of client data, thereby affecting the generalization performance of the global model. Therefore, this study proposes a wireless traffic prediction model that combines FL, Long Short-Term Memory (LSTM), and Kolmogorov-Arnold Networks (KAN). The aim is to enhance prediction accuracy while ensuring data privacy. The model leverages edge computing and FL techniques, enabling multiple edge devices to train LSTM models locally. By integrating KAN to explore the inherent complexity of traffic data, it enhances the model's adaptability. Furthermore, the FedNova model is introduced, which thoroughly considers the heterogeneity of data among devices and optimizes the process of updating model parameters. The extensive experimental results conducted on the Milano real dataset demonstrate that the FedNova-LSTM-KAN model exhibits significant advantages compared to existing centralized and FL methods when dealing with uneven client traffic data.**

***Index Terms*—Traffic Prediction, Federated Learning (FL), Kolmogorov-Arnold Networks (KAN), Data Privacy**

I. Introduction

In the current transition to 6G networks, mobile traffic prediction is crucial for enhancing the efficiency of mobile network management. Currently, traffic prediction primarily relies on centralized learning methods, which necessitate transferring a large amount of raw data to data centers to train a unified prediction model [1]. During this process, data transmission and signal overhead may rapidly consume network resources and affect data transmission efficiency. At the same time, the complex spatiotemporal correlations caused by user mobility pose higher requirements for model capture and modeling techniques. Accurately predicting data transmission at the network level is a challenge that needs to be addressed. The main contributions of the model proposed in this article include the following three aspects.

- Introducing the FedNova framework fully utilizes its advantages in distributed learning environments. Through its unique standardized averaging strategy, it effectively solves the heterogeneity problem of different devices in the optimization objective, improving the model's generalization ability and convergence speed.
- By combining LSTM with KAN techniques, a traffic prediction model tailored for heterogeneous base station data has been designed. The application of KAN techniques enhances the model's ability to identify and predict network traffic fluctuation patterns, particularly demonstrating significant advantages in analyzing and fitting traffic characteristics among different base stations.
- Extensive experimental validations of the FedNova-LSTM-KAN model were conducted on three real-world datasets. The experiments were conducted from three different perspectives, and the results confirmed that our framework exhibits higher accuracy and generalization ability in traffic prediction tasks.

II. Related Work

Traffic prediction is currently a hot research topic. With the development of technology, deep learning, especially RNN and LSTM, has shown great potential in mobile traffic prediction. A neural network capable of identifying cyclic patterns in various indicators was designed and implemented in reference [2]. Experimental results showed that the neural network can process prediction tasks faster and more accurately in a custom architecture. In practical applications, transferring a large amount of data to a single data center may raise concerns about data confidentiality, privacy protection, and data transmission requirements. FL technology provides an effective solution to protect data privacy across multiple devices by avoiding sharing raw data.

In practical applications, parallel modeling of community traffic reaching tens of millions of orders of magnitude faces enormous challenges. FL technology [3], [4] provides an effective solution to protect data privacy across multiple devices by avoiding sharing raw data. For example, the FedProx algorithm [5] limits the distance between local model updates and global models by introducing adjustable regularization terms. This method helps to maintain consistency in model updates in heterogeneous networks, prevent local models from deviating too far from the global optimal solution, and thus improve the convergence of FL. The FedAtt algorithm [6] introduces an attention mechanism to consider the contribution of different clients to the global model. This mechanism can assign different weights to clients, allowing clients who contribute more to the global model's generalization ability to have a higher influence in model updates. Although these methods address heterogeneity issues in the FL framework, there is still room for improvement. Our method utilizes the FedNova algorithm to consider client personalized features during model aggregation, and responds to client personalized needs through adaptive local updates, weight adjustments, and asynchronous update support. In addition, integrating LSTM enables the model to capture time series features and improve prediction accuracy. Finally, by using KAN to model nonlinear relationships, the adaptability and generalization ability of the model to heterogeneous data were enhanced, making it perform better in complex problems such as mobile traffic prediction.

III. Proposed Framework for Traffic Forecasting

The steps for constructing the FL-based traffic prediction framework proposed in this paper are as follows: Initially, on each mobile edge computing node, the LSTM-KAN model is used to train local traffic data. Subsequently, with the support of the cloud server, leveraging the FedNova strategy, the independently trained LSTM-KAN models are aggregated into a unified and more powerful traffic prediction model, namely FedNova-LSTM-KAN. This section will comprehensively describe the training process of the LSTM-KAN model deployed on the mobile edge computing nodes, as well as how the cloud server utilizes the FedNova algorithm to construct the FedNova-LSTM-KAN model.

A. Overall Framework

This section provides a detailed introduction to the proposed traffic prediction framework that combines the FedNova and LSTM-KAN models. The framework consists of a central server and multiple edge clients, as illustrated in Fig. 1. Firstly, the data from each client is preprocessed to ensure its suitability. Subsequently, the server initializes a global model and sends the initial parameters to all clients. Each client independently trains the LSTM-KAN model based on local data and adjusts parameters through optimization algorithms to enhance model performance. During the training process, after completing model updates, the clients upload their parameters to the server. The server utilizes the FedNova algorithm to weight and integrate the uploaded model parameters based on each client's data characteristics and quality, forming new parameters for the global model. These updated parameters are then distributed back to the clients for further optimization of their local models. Through multiple rounds of iterative training, the model undergoes continuous optimization between the clients and the server until it meets the specified performance standards. Finally, the clients evaluate the trained global model using their local test datasets to assess its prediction accuracy.

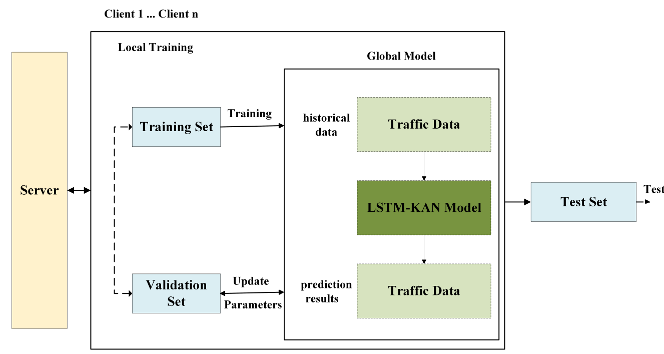

Fig. 1. Federated Learning Traffic Prediction Framework based on LSTM-KAN Model.

B. Client Model

LSTM is able to identify long-term dependencies in time series data, mainly due to its unique internal mechanisms, including the gate system of forgetting mechanism, input mechanism, and output mechanism, which work together to maintain or update cellular states. Such architecture endows LSTM with strong performance in tasks involving sequential data processing, language modeling, and time series prediction.

KAN is a technique that integrates the principles of Kolmogorov complexity and Arnold transformation into neural network architectures. Integrating the Arnold transformation into neural networks can enhance the ability of the KAN technique to simulate temporal dynamics and internal relationships within sequential data. As a revolutionary neural network structure, KAN differs from conventional multilayer perceptrons by applying activation functions to the weights of the network. This makes each weight a trainable one-dimensional function, typically represented using spline functions, replacing traditional linear weights. In the KAN network, nodes only perform summation operations on input signals without involving nonlinear processing. This simplified design not only makes the model more concise but also enhances its accuracy and parameter utilization in tasks such as data modeling and solving partial differential equations.

Combining LSTM with KAN can build a powerful predictive model that not only inherits the advantages of LSTM in handling sequential data but also integrates the ability of KAN to enhance the network in capturing complex patterns and regularities in the data. Fig. 2 illustrates the basic framework structure of the LSTM-KAN model.

The equations involved in the LSTM-KAN model are as follows:

Forget gate:

$$f_t = \sigma\left(W_f\left[h_{t-1}, x_t\right] + b_f\right) \tag{1}$$

Input gate:

$$i_t = \sigma\left(W_i\left[h_{t-1}, x_t\right] + b_i\right) \tag{2}$$

$$g_t = \tanh\left(W_g\left[h_{t-1}, x_t\right] + b_g\right) \tag{3}$$

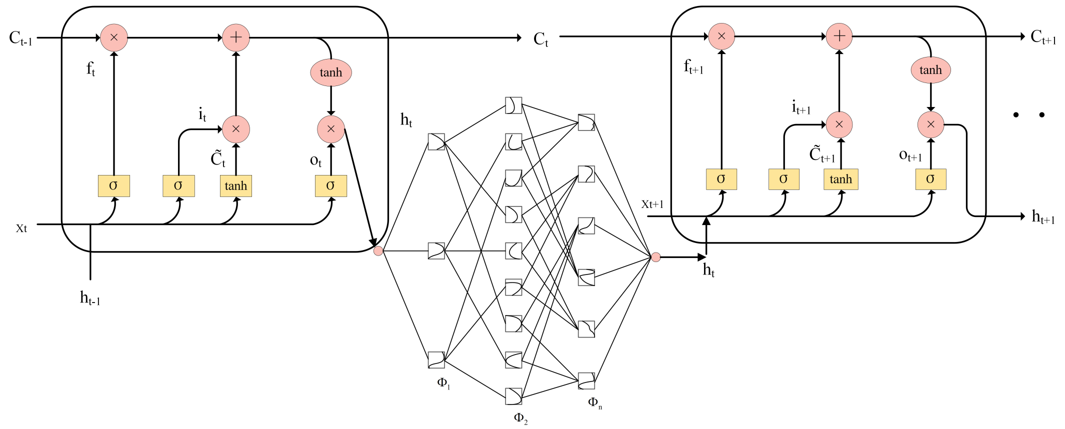

Fig. 2. Traffic Prediction Architecture based on the LSTM-KAN Model.

Cell state:

$$c_t = f_t \times c_{t-1} + i_t \times g_t \tag{4}$$

Output gate:

$$o_t = \sigma \left(W_o \left[h_{t-1}, x_t \right] + b_o \right) \tag{5}$$

$$h_t = o_t \times \tanh(c_t) \tag{6}$$

Kolmogorov-Arnold Networks:

$$KAN(h_t) = \left(\Phi_{l-1} \circ \Phi_{l-2} \circ \cdots \circ \Phi_2 \circ \Phi_1 \right) h_t \tag{7}$$

$$\Phi = \{\phi_{q,p}\} \tag{8}$$

$$\phi(h_t) = w \left(b(h_t) + \sum_i c_i B_i(h_t) \right) \tag{9}$$

$$b(h_t) = \frac{x}{1 + e^{-x}} \tag{10}$$

The forget gate determines which information to retain or discard from the cell state. This decision is based on the current time step's input x_t and the previous time step's hidden state h_{t-1}, implemented through a sigmoid function with weights W_f and biases b_f. The input gate is responsible for determining which information in the current time step is important and should be added to the cell state. This includes the input gate i_t calculated by the sigmoid function and the candidate values g_t generated by the $\tanh$ function, where the weights are denoted by W_i and W_c, and the biases are denoted by b_i and b_c.

The update of the cell state C_t combines the outputs of the forget gate and the input gate, as well as the cell state from the previous time step C_{t-1}. The output gate determines the value of the next hidden state, which involves the current time step's input x_t and the previous time step's hidden state h_{t-1}, implemented through a sigmoid function with weights W_o and bias b_o, along with a $\tanh$ function. Φ is the functional matrix, Φ_l is the functional matrix of the l-th layer of KAN, q is the input dimension, p is the output dimension, and $\sum_i c_i B_i(h_t)$ represents the linear combination parameterized by B-splines, where $b(h_t)$ represents the activation function.

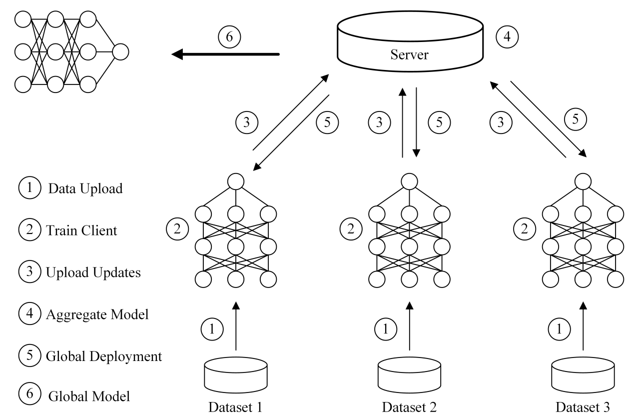

Fig. 3. Workflow of Federated Learning.

Through these computational steps, the client model is capable of effectively handling long sequence data and demonstrating superior performance in various time series analysis tasks.

C. Server-side ModelClient Model

In conventional approaches, centralized machine learning algorithms or centralized computing methods typically require data to be sent from various nodes to a central server for model training and processing. This approach is not only inefficient but also poses risks of data leakage. The core advantage of FL lies in its ability to overcome the limitations of traditional centralized algorithms, especially when dealing with large-scale distributed data. The workflow of FL is depicted in Fig. 3, detailing the entire process from local model training to server-side model aggregation.

The clients are responsible for preprocessing and segmenting local data, while the server oversees the configuration and initialization of the global model. The server sends the initial parameters of the global model to the clients, based on which the clients train and optimize their local models. After optimization, the clients send back the updated model parameters to the server, which integrates these parameters to update the global model. The updated global model is then distributed to the clients for the next round of training. This process will be repeatedly iterated through multiple rounds until the model training achieves satisfactory results. Ultimately, the clients will evaluate the performance of the model using their local test datasets.

In most FL algorithms, this process iterates continuously until the model training converges or the number of iterations reaches a predetermined limit. The FedNova algorithm employed in this paper adopts the core idea of normalizing the local gradients of each client by averaging them based on the number of local updates performed by the client, rather than simply averaging the unnormalized local gradients. This approach helps address data heterogeneity issues, enhances model performance, and reduces communication overhead. The detailed procedures of FedNova and other related algorithms used in this study are outlined in explicit algorithmic forms, such as Algorithm 1.

Algorithm 1 Federated Learning Algorithms

Input: local datasets D^i, number of clients N, number of federated rounds T, number of local epochs E, learning rate η, beta, $\beta_1, \beta_2 \in [0, 1)$ for FedAvgM, FedYogi, λ degree of adaptivity
Output: Final global model parameters vector w^T

Server executes:
Initialize w^0
for $t = 0$ to $T - 1$ **do**
  Sample a set of parties S_t
  $n \leftarrow \sum_{i \in S_t} \left|D^i\right|$
  for all $i \in S_t$ in parallel **do**
    Send the global model w^t to client C_i
    $\Delta w_i^t, \tau_i \leftarrow$ **LocalTraining**(i, w^t)
  end for
  $\Delta W \leftarrow \sum_{i \in S_t} \frac{\left|D^i\right|}{n} \Delta w_k^t$
  if FedAvg/FedProx **then**
    $w^{t+1} \leftarrow w^t - \eta \Delta W$
  else if FedNova **then**
    $w^{t+1} \leftarrow w^t - \eta \frac{\sum_{i \in S_t} \left|D^i\right| \tau_i}{n} \sum_{i \in S_t} \frac{\left|D^i\right|}{n \tau_i} \Delta w_k^t$
  else if FedAvgM **then**
    $u_t \leftarrow \beta u_{t-1} + \Delta W$
    $w^{t+1} \leftarrow w^t - u_t$
  else if FedAdagrad **then**
    $u_t \leftarrow u_{t-1} + \Delta W^2$
    $w^{t+1} \leftarrow w^t + \eta \frac{\Delta W}{\sqrt{u_t} + \lambda}$
  else if FedYogi **then**
    $m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1) \Delta W$
    $u_t \leftarrow u_{t-1} - (1 - \beta_2) \Delta W^2 sign(u_{t-1} - \Delta W^2)$
    $w^{t+1} \leftarrow w^t + \eta \frac{m_t}{\sqrt{u_t} + \lambda}$
  else if FedAdam **then**
    $m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1) \Delta W$
    $u_t \leftarrow \beta_2 u_{t-1} + (1 - \beta_2) \Delta W^2$
    $w^{t+1} \leftarrow w^t + \eta \frac{m_t}{\sqrt{u_t} + \lambda}$
  end if
end for
return w^T
Client executes:
For every algorithm: $L(w; b) = \sum_{(x,y) \in b} l(w; x; y)$
For FedProx: $L(w; b) = \sum_{(x,y) \in b} l(w; x; y) + \frac{\mu}{2} \left|w - w^t\right|^2$

IV. Experiments and Analysis

The experimental section conducts comparative analysis from three dimensions. 1) Comparing the proposed FedNova-LSTM-KAN model with traditional traffic prediction models validates that FL, represented by FedNova, outperforms traditional centralized learning. 2) Contrasting the proposed FedNova-LSTM-KAN with FedNova-LSTM and LSTM validates that FL based on LSTM-KAN outperforms conventional methods. 3) Comparing the FL method based on LSTM-KAN with five FL strategies: FedAvg, FedAdagrad, FedYogi, FedAvgM, and FedAdam, the FedNova algorithm demonstrates its advantage in solving the problem of ”target inconsistency” caused by differences in client data distribution and computing speed.

A. Dataset

The dataset used in this study is sourced from the Telecom Italia Big Data Challenge [7]. This dataset encompasses detailed communication traffic records from two regions in Italy: Milan and Trentino. The focus of the study was on the traffic data of Milan, which was divided into 10,000 regions. Each region has widely recorded the traffic usage of users in three different communication activities: SMS, voice call and Internet service. The data covers a two-month period from November 1, 2013 to January 1, 2014 with the raw data recorded at 10-minute intervals. To enhance the efficiency and accuracy of model training, this study resampled the data to record once per hour, reducing the potential impact of inadequate data volume within individual time intervals.

B. Experimental Environment and Parameter Settings

During the model training process, the dataset is initially split with 20% set aside for validation and the remaining 80% used for training. The LSTM network model parameters are set with a hidden size of 128 and the number of layers set to 1. Gradually adjusting parameters during the training process to achieve an optimal model. The learning rate is set to 0.001, with a Batch size of 32 and Epoch size of 300. The optimizer selected is Adam, with FL rounds set to 100.

The hardware setup for the experiment includes a device running the Windows 11 operating system, equipped with an Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz processor, 8 GB of memory, an Intel(R) UHD Graphics 620 GPU, and two additional GPUs. In terms of software environment, Python version 3.10.14 and Torch version 2.3.0 are used.

C. Evaluation Metrics

This study employed two evaluation metrics, Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), to measure the accuracy of the model's predictive performance. Specifically, the considered metrics are defined as follows:

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{11}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \tag{12}$$

where y_i represents the actual value of the i-th traffic, $\hat{y}_i$ represents the predicted value of the i-th traffic, and n represents the number of data points included in the test set.

D. Comparison of Federated Learning and Centralized Learning Algorithms

In this section, the proposed FL algorithm is compared with other popular centralized learning algorithms in terms of performance. On the same dataset, we evaluated the predictive performance of CNN, RNN, GRU, Transformer, Bi-LSTM, LSTM-KAN, Bi-LSTM-KAN, and Encoder-Decoder models, comparing their predictive errors with the FedNova-LSTM-KAN model proposed in this study. During this process, centralized learning algorithms aggregate the traffic data from the three areas, utilizing a global model for overall traffic prediction. The RMSE and MAE of all these customized models are averaged to form a single metric, "Avg.", to measure the predictive accuracy of the FedNova-LSTM-KAN model. Specific data can be found in Table I.

On the selected three traffic datasets, compared to the best-performing traditional centralized method, the Bi-LSTM model, the FedNova-LSTM-KAN model achieved a reduction of 12.2% in RMSE and 13.2% in MAE. Compared with existing models, the FedNova LSTM-KAN model proposed in this study exhibits superior performance. This achievement is attributed to the successful integration of LSTM's expertise in processing sequential data and KAN's advantage in function approximation, effectively capturing the temporal and spatial features of traffic data. In addition, the model also possesses interpretability and interactivity that MLP lacks, and can achieve higher prediction accuracy with fewer parameters, demonstrating its excellent parameter efficiency.

E. Ablation Study

To evaluate the stability and robustness of the proposed model in this study, we conducted ablation experiments to identify the key components that influence the model's performance. In the ablation experiments, we specifically focused on centralized learning methods conducted within the same region, conventional FL strategies, and the FL strategy proposed in this study that integrates KAN.

As shown in Table II, we conducted an analysis of the contributions of FedNova and KAN. FedNova-LSTM-KAN performed the best in terms of predictive accuracy, followed by FedNova-LSTM, with LSTM showing relatively poorer performance. From Table 3, it is evident that our method outperforms centralized learning and standard FL methods in traffic prediction performance across the three datasets. Compared to centralized LSTM learning, after training with our method, the RMSE values for the three datasets all show a decreasing trend. The RMSE performance for data1, data2, and data3 decreased by approximately 27.7%, 52.3%, and 14.8% respectively. The MAE performance for data1, data2, and data3 decreased by approximately 32.2%, 72.2%, and 17.4% respectively.

F. Comparison of Different Federated Learning Strategies

To further validate the effectiveness of the proposed research method, especially in examining the performance of different FL strategies when dealing with heterogeneous data, this section conducts a comparative analysis between five FL strategies (FedAvg, FedAdgrad, FedYogi, FedAdam, and FedAvgM) in traffic data prediction and the predictive results of the model proposed in this paper. Through this comparison, we can more accurately assess the performance of the proposed model in real-world applications.

Table III presents a comparative analysis of six FL algorithms, namely FedAvg, FedAdagrad, FedYogi, FedAvgM, FedAdam, and FedNova, in terms of their performance on traffic prediction tasks, as measured by RMSE and MAE metrics. The experimental results indicate that compared to FedAvg, FedAdagrad, FedYogi, FedAvgM, and FedAdam, FedNova achieved a decrease in RMSE of 3.0%, 9.2%, 8.0%, 22.8%, 12.9% respectively, and a decrease in MAE of 6.2%, 13.8%, 15.6%, 28.1%, 22.5% respectively. The improvement in model performance can be attributed to the FedNova algorithm, which, after initial training at the server end, allows fine-tuning on individual client devices using local data, thereby enabling the model to better adapt to their respective data characteristics.

V. Conclusion

In response to the limitations of traditional centralized traffic prediction methods in terms of data throughput and prediction accuracy, this study proposes a traffic prediction method based on FedNova-LSTM-KAN. Through a series of simulation experiments, we have derived several key conclusions:

- Compared to centralized models, our model achieves lower prediction errors while reducing data exchanges. This indicates that FedNova can enhance prediction accuracy while reducing data transmission requirements.
- Through ablation experiments, we validated the key contributions of KAN technology and the FedNova algorithm in enhancing model performance. In the experiments, incrementally integrating FedNova with KAN significantly and consistently improved the efficiency of the model, highlighting the superiority of the proposed model in traffic data prediction and time series modeling.
- Among numerous comparisons of FL algorithms, the combination of LSTM-KAN and FedNova demonstrates significant performance advantages. The FedNova algorithm effectively integrates individual differences and improves the adaptability and generalization ability of the

TABLE I. Comparison between this method and traditional deep learning methods.

Model	RMSE				MAE			
	data1	**data2**	**data3**	**Avg**	**data1**	**data2**	**data3**	**Avg**
CNN	13.5073	15.5959	49.85	26.3177	12.9927	12.6491	31.6009	19.0809
RNN	6.068	14.7363	47.9939	22.9327	3.9935	10.6626	28.6122	14.4228
GRU	5.5909	10.7789	49.3275	21.8991	3.5843	6.2459	29.6809	13.1704
Transformer	24.599	18.7123	49.0107	30.7740	24.1948	16.1726	29.7974	23.3883
Bi-LSTM	**4.4755**	10.801	48.1306	21.1357	**2.9493**	8.2449	25.3446	12.1796
LSTM-KAN	4.6393	9.3991	47.3947	20.4777	2.9966	6.1	27.584	12.2269
Bi-LSTM-KAN	16.4564	15.8133	46.2226	26.1641	16.1308	14.3116	26.7727	19.0717
Encoder-Decoder	15.6929	16.3037	57.5878	29.8615	5.8126	7.3247	33.8788	15.6720
FedNova-LSTM-KAN	5.2439	**7.9341**	**42.464**	**18.5473**	4.2337	**4.1061**	**23.3855**	**10.5751**

TABLE II. Comparison of Ablation Experiment Data.

Model	RMSE				MAE			
	data1	**data2**	**data3**	**Avg**	**data1**	**data2**	**data3**	**Avg**
LSTM	7.2533	16.6293	49.8247	24.5691	6.2455	14.7814	28.3003	16.4424
FedNova-LSTM	5.3853	9.6705	44.878	19.9779	4.3652	6.9931	25.7207	12.3597
FedNova-LSTM-KAN	**5.2439**	**7.9341**	**42.464**	**18.5473**	**4.2337**	**4.1061**	**23.3855**	**10.5751**

TABLE III. RMSE and MAE of Different Federated Learning Methods.

Model	RMSE				MAE			
	data1	**data2**	**data3**	**Avg**	**data1**	**data2**	**data3**	**Avg**
FedAvg-LSTM-KAN	5.4026	8.0526	43.9209	19.1254	4.4308	4.692	24.6922	11.2717
FedAdagrad-LSTM-KAN	5.3704	8.4594	47.4269	20.4189	4.3033	4.5283	27.9576	12.2631
FedYogi-LSTM-KAN	6.9765	8.7394	44.7925	20.1695	6.2806	5.5071	25.8135	12.5337
FedAvgM-LSTM-KAN	12.9249	13.5322	45.6592	24.0388	8.2225	9.4794	26.4097	14.7039
FedAdam-LSTM-KAN	5.5404	13.0488	45.2734	21.2875	**4.1597**	9.8154	26.9731	13.6494
FedNova-LSTM-KAN	**5.2439**	**7.9341**	**42.464**	**18.5473**	4.2337	**4.1061**	**23.3855**	**10.5751**

model through personalized fine-tuning and normalized averaging techniques.

Overall, the traffic prediction method proposed in this study has demonstrated outstanding performance across multiple base station traffic prediction tasks. Future research plans to expand the range of traffic data, incorporating factors such as weather and base station density to improve prediction accuracy, and optimize federated learning algorithms to enhance accuracy.

ACKNOWLEDGMENT

This research is supported by the National Research Foundation, Singapore and Infocomm Media Development Authority under its Trust Tech Funding Initiative, Strategic Capability Research Centres Funding Initiative and Future Communications Research & Development Programme. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore and Infocomm Media Development Authority.

REFERENCES

[1] W. Jiang, J. Luo, "Graph neural network for traffic forecasting: A survey," *Expert systems with applications*, vol. 207, pp. 117921, 2022.

[2] D. Anestis, L. Malamati, K. George, "Traffic forecasting in cellular networks using the LSTM RNN" in *Proceedings of the 22nd Pan-Hellenic conference on informatics*, pp. 28-33, 2018.

[3] Y. Liu, J. J. Q. Yu, J. Kang, D. Niyato and S. Zhang, "Privacy-Preserving Traffic Flow Prediction: A Federated Learning Approach," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7751-7763, 2020.

[4] Z. Qin, G. Y. Li and H. Ye, "Federated Learning and Wireless Communications," *IEEE Wireless Communications*, vol. 28, no. 5, pp. 134-140, 2021.

[5] T. Li, Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., and Smith, V. "Federated optimization in heterogeneous networks." in *Proceedings of Machine learning and systems 2*, pp. 429-450, 2020.

[6] S. Ji, S. Pan, G. Long, X. Li, J. Jiang and Z. Huang, "Learning Private Neural Language Modeling with Attentive Aggregation," in *2019 International Joint Conference on Neural Networks (IJCNN)*, pp. 1-8, 2019.

[7] G. Barlacchi, M. De Nadai, R. Larcher et al, "A multi-source dataset of urban life in the city of Milan and the Province of Trentino," *Sci Data 2*, pp. 150055, 2015.