# Machine Learning-Based Fifth-Generation Network Traffic Prediction Using Federated Learning

Mohamed Abdelkarim Nimir Harir[1], Edwin Ataro[2],Clement Temaneh Nyah[3]
Electrical Engineering (Telecommunication option), Pan African University Institute of Basic Sciences Technology and Innovation (PAUSTI), Nairobi, Kenya[1]
School of Electrical and Electronic Engineering, Technical University of Kenya, Nairobi, Kenya[2]
Electrical and Computer Engineering, University of Namibia, Windhoek, Namibia[3]

***Abstract*—The rapid development and advancement of 5G technologies and smart devices are associated with faster data transmission rates, reduced latency, more network capacity, and more dependability over 4G networks. However, the networks are also more complex due to the diverse range of applications and technologies, massive device connectivity, and dynamic network conditions. The dynamic and complex nature of the 5G networks requires advanced and accurate traffic prediction methods to optimize resource allocation, enhance the quality of service, and improve network performance. Hence, there is a growing demand for training methods to generate high-quality predictions capable of generalizing to new data across various parties. Traditional methods typically involve gathering data from multiple base stations, transmitting it to a central server, and performing machine learning operations on the collected data. This work suggests a hybrid model of Long Short Term Memory (LSTM), Gated Recurrent Unit (GRU), and federated learning applied to 5G network traffic prediction. The model is assessed on one-step predictions, comparing its performance with standalone LSTM and GRU models within a federated learning environment. In evaluating the predictive performance of the proposed federated learning architecture compared to centralized learning, the federated learning approach results in lower Root Mean Square error (RMSE) and Mean Absolute Errors (MAE) and a 2.25 percent better Coefficient of Determination (R squared).**

***Keywords—5G Mobile network; machine learning; federated learning; parallel hybrid LSTM+GRU; network traffic prediction; centralized learning; dynamic network condition***

## I. Introduction

As witnessed the evolution of communication networks into the 5G era, the demand for high-speed, low-latency connectivity is growing exponentially. The development of 5G networks also increases data rates and complexity due to various services and multiplexed device connections; this makes network resource management of a 5G network a complicated task because of the diverse nature of network traffic conditions. The growth in the number of users and devices is increasing traffic exponentially, causing congestion in the network from many angles [1].

Due to most devices now being connected, the conventional 4G networks cannot meet the current demand. The advantages that the 5G network can provide are becoming more visible as its scope continues to expand. Compared to their 4G counterparts, 5G networks provide faster data transmission, reduced delay, expanded coverage, and greater reliability.

Managing 5G networks as they evolve and become more complicated is now one of the greatest difficulties with developing them. One critical element of this management is predicting network traffic, which has advanced greatly through machine learning techniques [1], [2].

With new elements such as millimeter waves, massive MIMO (Multiple Input, Multiple Output), and network slicing, 5G networks are much more sophisticated than their predecessors. This complexity requires sophisticated traffic control methods. Moreover, the various services of 5G, such as enhanced mobile broadband (eMBB), massive machine-type communication (mMTC), and ultra-reliable low-latency communication (URLLC) [3], imply different traffic dynamics that require prediction and control.

The huge increase in the number of devices connected leads to an increase in the volume of mobile traffic and adds stress to the system to cope with the volume of data [4]. Ericsson expects 5G subscriptions projected at approximately 610 million by the end of 2023, meaning that about one-fifth of all mobile subscriptions worldwide would be 5G [5]. The research predicted demand growth will increase to approximately 5.3 billion 5G subscriptions by 2029 [6].

5G networks allow the implementation of edge computing, which brings computation and data storage close to the network edge. By processing data at the edge, latency-sensitive applications might achieve lower network utilization levels while enjoying greater speeds, security, and privacy [7].

Network traffic forecasting was based on statistical models. Techniques such as time series analysis, regression analysis, and Markov models have been employed to forecast network behavior by leveraging historical traffic data. Time series models such as the Autoregressive Integrated Moving Average (ARIMA) are effective tools in identifying seasonal trends and irregular patterns in data over periods [8]. They make it possible to determine traffic cycles that could be predicted at intervals of a day, week, or even a month. Regression models, e.g., linear, polynomial, and multiple regression analysis, can explain such relations as the time of day, human activity, and external factors such as weather. Besides this, these models are good at understanding relationships influencing traffic volume.

Markov models such as the Hidden Markov Models (HMM) and the Markov Chain Monte Carlo (MCMC), on the other hand, apply a probabilistic technique to estimate different stages of the network, thus market appeal and ability to offer better traffic prediction because of considering the stochasticity of the network traffic [8], [9]. Some of these traditional models have also been used for forecasting traffic.

However, with the growth of the networks, the weaknesses of these models are becoming more of a concern. In most cases, these models obtrude the linearity, which, along with the inability to manage high dimensional data, high rate of change in patterns of networks, or manage an anomaly, which is not a common phenomenon in the network [9]. This highlights the requirement for more sophisticated predictive models. Today, machine learning (ML) methods are believed to help deal with the complexities of 5G traffic, especially concerning the predictive aspect. Unlike traditional models, ML models can handle large volumes of datasets, capture nonlinear relationships, and learn and update to changes occurring in real-time.

The advantage of machine learning models is that they can discover intersectional structures or relationships that can be captured through conventional statistical means such as models by training generally on large datasets [10]. For instance, deep learning models such as Recurrent Neural Networks (RNNs) and long short-term memory (LSTM) appreciate recognizing sequential information in a given network traffic data frame rate, improving precision forecast along the time scale. Similarly, incentive-based resource allocation has been demonstrated to integrate across existing resource conditions by learning sub-network policies with dynamic structures [11].

Federated learning (FL) is projected as one of the many machine learning paradigms appropriate for 5G networks. Federated learning is a novel approach for training models without needing a central server to host raw data on devices [12]. This is highly timely in the case of fifth-generation networks (5G), as data privacy and security are critical owing to the potential proliferation of personal gadgets and the IoT. Federated Learning helps model training with privacy, bandwidth, and data constraints by leveraging edge devices in a distributed manner.

FL advancements enable privacy and more efficient model training, as the processing of network traffic data can occur at the edge of the network [13], [14]. This paper studies how predictive machine learning models based on federated learning can be employed to predict traffic in 5G networks.

The structure of the paper is as follows: Section II reviews the related work, while Section III outlines the architectures of the prediction methods. Section IV delves into the prediction methodology. The experimental results are presented in Section V, and the paper concludes in Section VI.

## II. Related Work

Advancements in 5G networks have made efficiently managing their dynamic and complex nature challenging. One possible solution to the above is predicting the network traffic on such a network, which is steadily receiving support in the form of machine learning advances.

The author in study [15], through network Internet traffic analysis and forecasting of input traffic flow parameters to the model, developed a 5G network traffic prediction model that utilizes recurrent neural networks in their paper. They have employed Gated Recurrent Units (GRU) and (LSTM) to obtain a balance between optimality and viability. Such networks have acquired short-term traffic predictions since feature engineering was introduced to the model to reduce generalization errors and manage missing and corrupted data. Still, there is a need for more research on machine learning application techniques for network management and control in traditional distributed architectures.

In study [1], this paper presents a lightweight hybrid attention deep learning model for traffic prediction in 5G networks. The model integrates depthwise separable convolution with channel and spatial attention techniques to lower prediction costs. With its capacity to conserve computing resources, the model exhibits promise for use in integrated sensing, communication, and computation applications. The temporal and spatial properties of 5G network traffic data are revealed through data analysis, and the suggested model effectively addresses accuracy and complexity concerns using feature extraction and prediction capabilities.

To improve its prediction capabilities for 5G cellular network traffic flow, the authors in study [4] propose a deep learning model based on a Bidirectional Long Short-Term Memory (BiLSTM) architecture with hyperparameter optimization. The stated model demonstrates better prediction accuracy and shorter running time. Thus, it is helpful for real-time applications even though the authors did not discuss the practical limitations of deploying the model. The focus is on possible future research related to resource allocation schemes and IoT cloud architectures. Generally, the findings of the suggested Deep Learning Mobile Traffic Flow Prediction (DLMTFP) technique are encouraging for developing mobile traffic prediction in 5G networks.

In study [16], this paper proposes a Deep-Broad Learning System (DBLS) for traffic flow prediction in 5G cellular wireless networks. It explains that DBLS is suitable for 5G networks because it integrates deep representative and broad learning to provide accurate prediction while keeping the running time low. They showed that DBLS is more accurate and efficient than conventional deep neural networks. It is observed that enhancing the reasonable amounts of enhancement nodes adaptively can enhance the efficiency of the DBLS model and hence lead to high penetration prediction.

According to study [17], the study proposes to predict the traffic of the 5G network and its challenges, owing to the diversity and heterogeneous nature of the 5G traffic. To address these problems, a Smoothed Long Short-Term Memory (SLSTM) model is proposed to enhance prediction accuracy. Adjustments are made to the number of layers and hidden units based on the prediction accuracy, and seasonal time is based on the time series modeling techniques used to smooth the output sequences. This article recommends further research on other factors influencing 5G traffic to make it more applicable in practice.

In study [18], the study engages numerous cross-domain big data resources to construct a spatiotemporal cross-domain neural network model (STC-N) that enables deep learning in wireless cellular network regional traffic prediction. The method consists of the integration of feature fusion, multi-domain data integration, timestamp-based modeling, and spatiotemporal correlations. The paper also discusses a cross-domain transfer learning approach for improved prediction performance in traffic generation.

It focuses on how cross-domain datasets interact within the prediction model and how it affects the accuracy of the prediction. Nevertheless, the analysis of different kinds and volumes of cross-domain datasets, their synthesis, and association effects on wireless cellular traffic prediction accuracy deserves further attention. Although the reporting in this paper concerns the effect of many cross-domain datasets on prediction accuracy, there is scope for investigating the best combination and weighting of this dataset.

Paper in study [19] describes a novel method of estimating the traffic flow in cellular networks utilizing counters that monitor the performance of LTE radio frequency signals. It investigates a range of machine learning models that can forecast traffic in the network depending on time. It demonstrates that while ensembles such as Gradient Boosting produce the most accurate predictions and spend longer training time, linear models operate faster but depend on preprocessing. The analysis identifies the importance of having a large volume of good quality data necessary to train machine learning models and speaks to the challenges of deployment and solutions whereby autoML may be utilized during retraining, regularization, and feature engineering.

In study [20], this paper considers the issues of mobile network forecasting as applied in a distributed manner, specifically with forecasting traffic for base stations and 5G networks overall. It evaluates different aspects of the centralized and federated learning model, pointing out the strengths of federated learning for better generalization metrics, economies of computational resources, and less carbon dioxide emission. It also highlights the role of model aggregation algorithms and data preprocessing methods in improving the predictive power of the models. The models, which include LSTM and GRU, are quite effective in federated learning scenarios.

The research presented in the paper [21] investigates the efficiency of energy usage in augmented deep-learning model architecture. It discusses federated traffic prediction mechanisms for cellular networks for optimal energy usage. It shows how the difference in the region affects the performance of a wide variety of models, such as Transformer and Length short-term memory-based models. The results demonstrate that while complicated models are more demanding in energy, the expectation is also a high increase in accuracy. This study seeks to raise the understanding of Distributed AI Technologies' environmental impacts and their pose threats to communication systems. It advocates the merits of incorporating sustainability factors into model selection.

In study [22], the paper discusses the problems of implementing FL in vehicular IoT systems, such as variable mobility, limits to communication capability, and risks of non-IID data in combination with the management of resources. Working issues in FL for autonomous driving, intelligent transport systems, and resource-sharing developments are elaborated. The authors define the areas for further explorations, increasing FL advanced paradigms: scaling up and security on the background of the complex vehicular IoT scenarios.

## III. Prediction Method Architectures

### A. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a relatively sophisticated variant of the Recurrent Neural Network (RNN) model commonly employed in research today. One of the key aspects LSTM networks address is the long-range dependencies in sequential data, which results in higher performance in many practical applications [23]. Some use cases where LSTM networks have been highly successful include language translation, voice detection, and forecasting. This explains the popularity of LSTM networks in multiple applications and their efficiency in deep learning frameworks directed toward time series data. They encode the RNN memory with three gates alongside cell states, allowing the network to keep and erase information when necessary.

In the standard arrangement, an LSTM block consists of four extra layers and a hidden state in an RNN. Variables include Cell state ($C_t$), input gate ($i_t$), output gate ($o_t$), and forget gate ($f_t$). Each layer performs a specific operation on the others depending on how the information is created from the training data [24]. Fig. 1 shows the structure of LSTM.

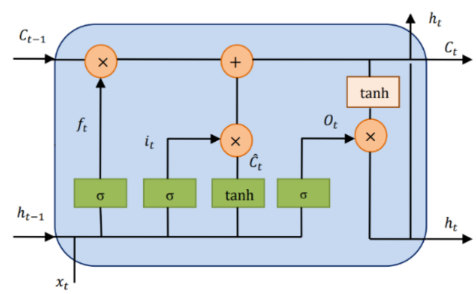


Fig. 1. Architecture of Long Short-Term Memory (LSTM) [23].

The memory of LSTM network networks is represented by the cell state, which is essential to LSTMs. The cell state process resembles a production line or conveyor belt. Except for a few linear interactions like addition and multiplication, the parameter information flows directly across the chain. These interactions determine the status of the information. The information will continue to flow without modifications if no interactions exist. Through the gates, which permit optional information to pass through, the LSTM block modifies or adds information to the cell state [24].

The forget gate eliminates data no longer needed in the cell state. The gate receives two inputs, $x_t$ (the input at that specific moment) and $h_{t-1}$ (the output of the previous cell), which are multiplied by weight matrices before bias is added. After being run through an activation function, the output is binary. When the output for a particular cell state is 0, the information is lost, and when the output is 1, it is saved for later use [25]. The nodal output equations of the LSTM are expressed as follows.

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

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

$$C_t^{\sim} = \tan h\,(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (3)$$

$$C_t = f_t * C_{t-1} + i_t * C_t^{\sim} \quad (4)$$

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

$$h_t = o_t * \tan h(c_t) \quad (6)$$

These equations describe how an LSTM unit works, distinguishing it from simple RNN, with several gates controlling information flow. The forget gate $f_t$ in (1) determines what portion of the previous cell state $C_{t-1}$ should be preserved, and the input gate it in Eq. (2) identifies how much new information from the current input $x_t$ and previously hidden state $h_{t-1}$ is added to the cell state. The candidate cell state $C_t^{\sim}$ in Eq. (3) is determined with a tanh function. The new cell state $C_t$ combines the old cell state and the new information added, as shown in Eq. (4). The output gate $o_t$ in Eq. (5) controls how much of the updated cell state $C_t$ is passed on to the hidden state $h_t$, influencing the output at this time step. The hidden state $h_t$ in Eq. (6) is finally computed, where the output gate $o_t$ is applied to the tanh of the new cell state, passing it on to the next time step, thus helping the LSTM keep long-term dependencies in sequential data.

Various parameters guide the internal mechanism of the network. $W_f$, $W_i$, $W_c$, and $W_o$ are the weight matrices multiplied by the forget gate, input gate, candidate cell state, and output gate, respectively, and they are used for both the previous hidden state, $h_{t-1}$, and current input $x_t$. Similarly, $b_f$, $b_i$, $b_c$, and $b_o$ are the biases related to these gates and states, added to the product sum of the inputs to bias the output. The sigmoid function σ is employed in the forget gate $f_t$, input gate $i_t$, and output gate $o_t$ to compress values between 0 and 1, indicating how much influence they should have (how much to forget, retain, and output, respectively).

### B. Gated Recurrent Unit (GRU)

GRU employs a gating mechanism to regulate the information passing through the network. The gates in LSTM determine which information to keep and which to discard at every step, enabling the network to learn long-range dependencies better. The GRU has two main components: the update and the reset gates.

The update gate decides how much new information to write to the memory now, and the reset gate decides how much old information to forget. The basic idea of GRU is that the network hidden state will be updated only by selecting time steps using gating mechanisms. The gates control what information joins and leaves the network. The GRU has two gating mechanisms: reset gate and update gate. The update gate specifies the proportion of the new input to add to the hidden state, and the reset gate specifies the extent to which the previous hidden state should be erased. The GRU output is computed based on the updated hidden state [23]. The architecture is shown in Fig. 2.

The update gate calculation is the first step in a GRU. It uses the current input and the previous hidden state to decide how much to update the previous hidden state; the sigmoid is used here [24]. Here are the GRU nodal output equations.

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z) \quad (7)$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \quad (8)$$

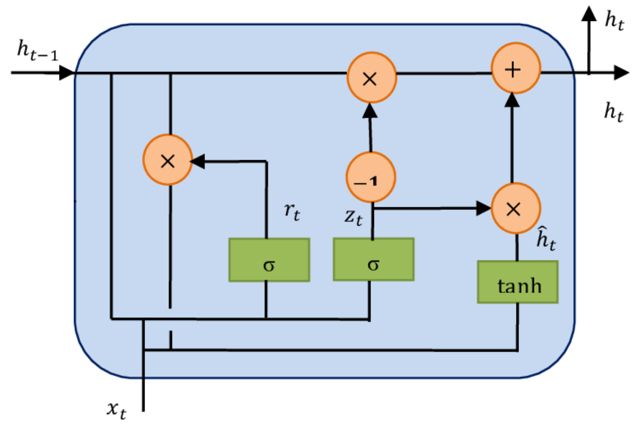


Fig. 2. Architecture of Gated Recurrent Unit (GRU) [23].

$$h_t^{\sim} = \tan h\,(W_h \cdot [r * h_{t-1}, x_t] + b_h) \quad (9)$$

$$h_t = z_t * h_{t-1} + (1 - z_t) * h_t^{\sim} \quad (10)$$

The GRU equations have several parameters that dictate how certain elements within the input data behave over the time steps. Where update gate $z_t$ defined in Eq. (7), uses weights Wz, biases bz, previously hidden state $h_{t-1}$, and current input $x_t$ to determine how much of the past hidden state will pass to the next step. In the same way, Eq. (8) also applies a reset gate $r_t$, which uses weights $W_r$, biases $b_r$, and a combined $h_{t-1}$ and $x_t$ to decide how much to "forget" of the past for computing the hidden state of the candidate [26].

The candidate hidden state $h_t^{\sim}$ is calculated from weights $W_h$, biases $b_h$, reset gate $r_t$ applied to $h_{t-1}$ and current input $x_t$, processed with the tannh function as shown in Eq. (9): Finally, the new hidden state $h_t$ in Eq. (10) is expressed as a weighted sum of the candidate hidden state $h_t^{\sim}$(scaled by $1 - z_t$) and the previous hidden state $h_{t-1}$ (scaled by $z_t$). The weights $W_z$, $W_r$, and $W_h$ and the biases $b_z$, $b_r$, and $b_h$ are learned during training and determine how inputs are decoded past and current information at each time step to adjust and combine the input information [27].

## IV. Proposed Method Description

### A. LSTM+GRU Parallel Network

In the proposed parallel hybrid model, the same input is applied to both LSTM and GRU layers. This enables the model to capture two different temporal representations simultaneously. This is especially beneficial because it combines the strengths of both architectures; thus, while the LSTM capability fortifies long-term dependencies, GRU computational efficiency and faster convergence make it an essential strength for more robust feature extraction in time series prediction tasks.

The input data processed through the LSTM and GRU branches come out as outputs from these branches. The outputs are then concatenated to form a combined feature representation [27]. After passing through dense layers, the final prediction is based on this combined representation. Fig. 3 shows the structure of the parallel hybrid model LSTM+GRU.

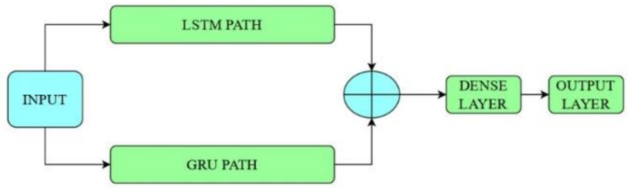


Fig. 3. Architecture of parallel hybrid model of LSTM+GRU.

As shown in Fig. 3, the model architecture consists of an input layer that receives the 5G traffic input data. The input is fed into both the LSTM and GRU paths simultaneously. Here, the LSTM (Long Short-Term Memory) network can capture long-term dependencies in time series data, which is important for identifying trends and patterns in 5G over long periods. However, the GRU (Gated Recurrent Unit) path takes the same input with fewer, faster-to-train steps in the architecture, allowing for efficient short-term dependency capture [27], [28]. Though both networks are good at processing sequential data, both contain different complementary strengths, such as LSTM being a long-term memory network and GRU being a memory-efficient network with fewer parameters.

As illustrated in Fig. 3, after processing the input via LSTM and GRU paths, the outputs are concatenated (as shown by the circle in Fig. 3). This merging step is performed to combine the information learned by the LSTM and GRU networks. This concatenated output is fed through a dense layer, and this layer helps further process the combined feature representation extracted from the LSTM and GRU branches [28]. Finally, the dense layer is connected to the output layer, which provides the model prediction.

## B. *Federated Learning*

Federated learning is an advanced machine learning approach that enables decentralized model training without sharing raw data between multiple devices or nodes. It is suitable for privacy-preserving scenarios, such as 5G network traffic prediction [20].

In a centralized learning setting, data brought in from different sources, such as base stations or user devices, would be aggregated in one place during model training, which can raise concerns about the privacy and security of data. With federation learning, each device or node learns a model based on its data at each device or node, and only the changes in the model (such as the weights or the gradients) are sent to the central server [21]. Afterward, this central server employs these updates to enhance the performance of the global model. Furthermore, in the context of 5G network traffic prediction, federated learning allows individual base stations or edge devices in the network to collaborate on training a predictive model without exchanging their raw traffic data. This keeps the users and network-sensitive data secure while providing realistic traffic pattern predictions [12].

In 5G networks, federated learning presents a valuable approach due to the large geographical distribution of data from numerous devices. Through the distributed learning of models locally trained on diverse data, federated learning can also improve the prediction about network congestion, traffic demand, and resource allocation for a particular network in the future while maintaining data privacy and low communication overhead in the network [21], [22]. Fig. 4 shows a round of the federated learning process.

The federated learning process involves multiple clients or base stations (BS) and a central server, as shown in Fig. 4. In step 1, the central server sends the global model to all clients. Step 2: Clients then update their local models by locally training the model with their private data. In step 3, clients return updated model parameters to the server (aggregator) without sharing raw data. These local model updates are then sent to a central server, which uses an aggregation function (that is, namely Federated Averaging) to aggregate these local model updates to produce an updated global model in Step 4. After updating the model, it repeatedly redistributes the new model to the clients for more training iterations. In this decentralized manner, clients update the global model in a privacy-preserving way by sending model updates rather than datasets.

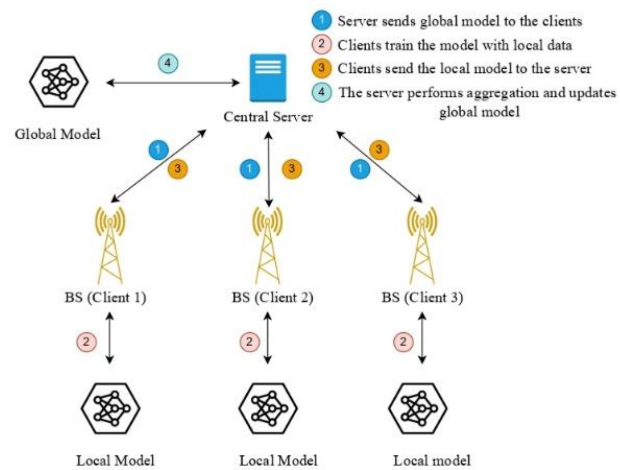


Fig. 4. Federated learning process [20].

Federated Averaging (FedAvg) is widely used due to its simplicity and effectiveness in handling non-iid (non-identically distributed) data across clients. This is common in real-world scenarios where different devices may have access to diverse datasets [12]. FedAvg also minimizes the communication overhead by reducing the frequency of interactions between the clients and the central server, making it well-suited for distributed environments.

## C. *Implementation of the LSTM+GRU Hybrid Model*

The flowchart in Fig. 5 illustrates the steps in implementing and evaluating a hybrid LSTM+GRU model for predicting 5G network traffic, including data preparation, model training, and model testing.

*1) Data preprocessing*: This stage addresses various data quality issues, such as outliers, missing values, and data splits. Missing values were handled based on the percentage of missing data in each feature. Features with more than 50% missing values were removed, while those with less than 50% were imputed using the mean of the column.

Outliers were managed using the Interquartile Range (IQR) capping method, which can limit the impact of extreme values and improve model robustness.

The data was split into training and testing sets with three different ratios (80:20, 85:15, and 90:10). In this study, the 90:10 ratio was used as it provided the best results. The 90-10 splits mean we used 90% of the data to train the model and 10% to test it. This ensures that most of the data is used for model training and that a different portion is set aside to evaluate it.

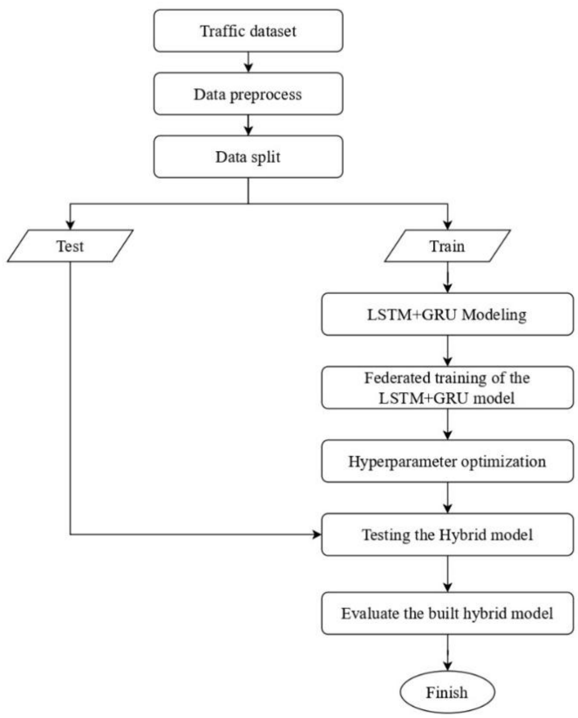


Fig. 5. Flowchart of modeling LSTM+GRU model for 5G network traffic prediction.

*2) LSTM+GRU modeling*: The hybrid LSTM-GRU model was implemented in Python and supported by TensorFlow libraries (i.e., Keras and TensorFlow Federated) using a Google Colab platform. The model had an LSTM layer of 128 units, then a GRU layer with the same number of units, and ReLU as an activation function. These layers were used in parallel (i.e., passed to an add() function). After that, the added output passes through a Dense layer with 64 units and ReLU activation. Finally, a Dense layer with 1 unit was added for the output. The Adam optimizer with a learning rate of 0.001 was used to optimize the model, and L1 and L2 regularizers (set to 0.05) were applied for overfitting prevention. The model was trained for 90 epochs with a batch size of 64.

*3) Federated training LSTM+GRU model*: The server starts the computation in federated training, and clients (base stations) join as participants. A subset of these clients are selected to receive the current global model from the server and use their local data to train [20], [21]. Once local training is done, the clients send the updated models and historical data (loss values and evaluation metrics) to the server. The server then aggregates the locally trained models, updates the global model, and repeats the process for several federated rounds, as shown in Fig. 4. Hybrid models combining LSTM and GRU have been proposed before [27], [28]. To the best of our knowledge, this work is the first to implement a parallel LSTM+GRU network and training using federated learning for time series prediction. The study created a flexible framework to make it more realistic for network traffic prediction scenarios. After the model is trained, it is tested on the held-out testing data to see how well it predicts 5G network traffic. This step checks how well the LSTM+GRU model can predict the traffic.

*4) Hyperparameter optimization*: Hyperparameter tuning is an important part of neural network development and is usually done through trial. Table I shows the model-specific hyperparameters.

TABLE I. MODEL HYPERPARAMETERS

| LSTM, GRU | LSTM+GRU |
|---|---|
| Activation: ReLU<br>Output layer: linear activation<br>No. of units: 128<br>Dense layer: 64 units<br>Optimizer: Adam<br>Regularizer L1, L2: 0.06<br>Learning rate: 0.001<br>Drop out: 0.4<br>Local Epochs: 3<br>Batch size: 64<br>Federated rounds: 10 | Activation: ReLU for both branches<br>Output layer: linear activation<br>No. of units: 128 for both branches<br>Dense layer: 64 units<br>Optimizer: Adam<br>Regularizer L1, L2: 0.05<br>Learning rate: 0.001<br>Dropout: 0.2<br>Local Epochs: 3<br>Batch size: 64<br>Federated rounds:10 |

A validation run was done for each model to finalize the hyperparameters that gave the best performance and fit before training the final model. The training data was split 90-10 for validation during the validation process. Keras search is used for hyperparameter optimization to get the parameters shown in Table I.

*5) Evaluation metrics*: To evaluate and analyze the network model prediction results, the evaluation metrics used are the Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Coefficient of Determination ($R^2$). The corresponding mathematical formulas are presented in Eq. (11,12,13). RMSE in Eq. (11) is particularly effective at measuring the model dispersion, where a lower RMSE indicates a higher concentration level and greater accuracy.

MAE in Eq. (12), measures the absolute differences between the predicted and actual results by taking the absolute values and then calculating the mean. A lower MAE signifies a smaller prediction error. R-squared in Eq. (13) is widely used as an optimal measure for assessing linear regression models, as it translates the prediction accuracy into a value between 0 and 1, offering an intuitive representation of the model accuracy [29]. When the model fit is ideal, the R-squared value approaches 1.

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{n}}\sum_{\mathrm{i}=1}^{\mathrm{n}}(f_i - y_i)^2} \tag{11}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|f_i - y_i| \quad (12)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - f_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (13)$$

## V. Experimental Results

### A. Dataset Description

The paper uses a 5G trace dataset from an Irish mobile telecommunication operator, as outlined in study [30]. It considers file downloading in a dynamic environment and uses the Download traffic bandwidth data produced from file downloads in a dynamic environment as the target variable. The data samples are aggregated for different days of the experimental period.

- 10,974 Samples collected from 2019/12/14/10:16:30 to 2019/12/17/08:16:23
- 4,106 Samples collected from 2020/01/16/07:26:43 to 2020/01/12:16:29
- 12,511 Samples collected from 2020/02/13/13:03:24 to 2020/02/27/20:50:06

Preprocessing steps were executed to clean and prepare the raw data for analysis. The preprocessing steps included feature normalization, missing value treatment, outlier treatment, and data samples collected on different days aggregated into one dataset. The integration finally resulted in 27,591 samples in the final dataset. Ten features were selected: GPS coordinates (longitude and latitude), timestamp, uplink bitrate, download bitrate and its download state, velocity, and several cellular signal indicators RSRQ (Reference Signal Received Quality), RSRP (Reference Signal Received Power), SNR (Signal-to-Noise Ratio), and CQI (Channel Quality Indicator). Cellular signal indicators are critical as they glimpse the network physical layer. For 5G systems, these features are pertinent since they correlate to how signal quality affects bandwidth and throughput. Velocity and geolocation information permits an exploration of how network performance may vary with mobility versus location environments; such considerations are crucial to many applications in 5G, where users typically move around a lot.

The dataset was gathered from the initial deployment phase of 5G, and it includes key performance indicators (KPIs) such as throughput, channel conditions, and context-related metrics. These metrics remain fundamental to understanding network performance, regardless of technological advancements. As 5G builds on similar foundational principles, the data provides insights that still apply today.

### B. Model Comparison

This study evaluates the performance of the proposed hybrid LSTM+GRU model against standalone LSTM and GRU models in the context of 5G network traffic prediction within the federated learning framework. Both LSTM and GRU units were specifically designed to capture temporal dependencies in sequential data; however, LSTM particularly excels in modeling long-term behaviors, while GRU gives a computationally efficient alternative for short-term dependencies with a simpler structure. The hybrid model is the parallel combination of these structures, thus permitting richer feature extraction by exploiting both strengths. Performance scores of the prediction models for 5G network traffic are shown in Table II.

TABLE II. Models Performance in 5G Network Traffic Prediction

| Models/Measures | RMSE | MAE | R2 |
|---|---|---|---|
| LSTM | 0.2360 | 0.3696 | 0830 |
| GRU | 0.2349 | 0.3656 | 0.833 |
| LSTM+GRU | 0.2291 | 0.3556 | 0.845 |

Table II highlights the superiority of the hybrid LSTM+GRU model in predicting 5G network traffic. The hybrid model achieved the lowest RMSE of 0.2291 and MAE of 0.3556, demonstrating its ability to minimize large and average prediction errors effectively. Furthermore, its $R^2$ value of 0.845, the highest among the models, indicates that it explains 84.5% of the variance in the data, making it the most accurate and generalized model for capturing both long-term and short-term traffic patterns. Although the stand-alone GRU model performed better than the LSTM model, the hybrid model always gave better results.

The hybrid model predictive performance is further corroborated by visualizations in Fig 6, 7, and 8, where its predictions closely align with the actual data, showing minimal deviations. This superior accuracy can be attributed to the combined architecture of LSTM and GRU. Despite the hybrid model superior performance, it is computationally more expensive due to its integrated architecture, which increases the number of parameters and requires more memory and processing power. The training time is also longer, as the model must optimize both LSTM and GRU layers.

However, in scenarios where computational resources are available, the hybrid model offers a worthwhile trade-off, as its enhanced accuracy and generalization make it ideal for applications like network optimization or capacity planning. The GRU model is a simpler yet effective alternative for environments with resource constraints or needing faster prediction. The standalone LSTM model, however, appears less suited for 5G traffic prediction due to its lower overall performance and difficulty adapting to the data's highly dynamic nature.

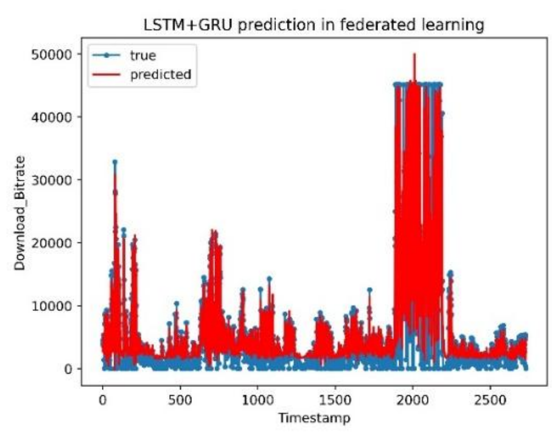


Fig. 6. Prediction of federated LSTM+GRU on test data.

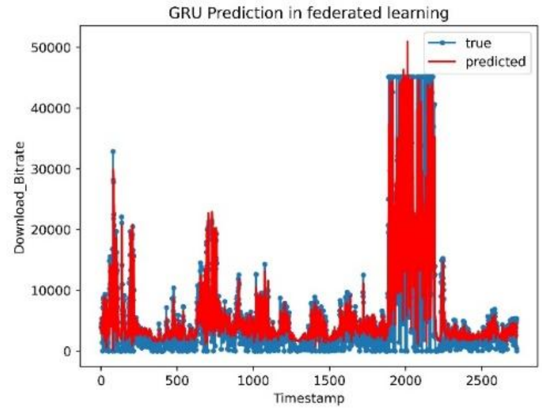


Fig. 7. Prediction of federated GRU on test data.

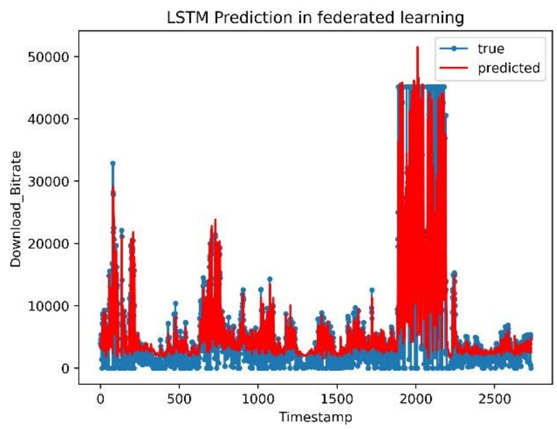


Fig. 8. Prediction of federated LSTM on test data.

### C. Learning Setting Comparison

The time conducted an exhaustive set of experiments on the 5G network traffic dataset to analyze the performance of deep learning models with a specific focus on the effectiveness of the federated learning framework. The study compared the LSTM+GRU hybrid model performance in centralized versus federated learning settings. A centralized learning environment involves training the model on the complete dataset on a single server, while federated learning trains local models at various participants (clients) that aggregate after each round. The model architecture consistency is maintained across both learning environments to achieve comparison equity. The LSTM+GRU hybrid model was applied in both scenarios. In the centralized learning setup, the model was trained for 90 epochs, letting the model traverse the entire dataset 90 times. Ten federated rounds and three local epochs on each client were executed for the federated learning setup. Since federated learning involves multiple clients, the total practical epochs across all clients is 90, obtained as (rounds × clients × local epochs). The two results obtained under different learning frameworks were compared upon completing the experiments, as shown in Table III.

TABLE III. PERFORMANCE OF FEDERATED LEARNING AND CENTRALIZED IN 5G NETWORK TRAFFIC PREDICTION

| Models/Measures | RMSE | MAE | R2 |
|---|---|---|---|
| Centralized | 0.2438 | 0.3687 | 0.826 |
| Federated | 0.2291 | 0.3556 | 0.845 |

Table III highlights a comparative analysis between centralized and federated learning in predicting 5G network traffic, emphasizing the advantages of federated learning. Federated learning achieved a reduced RMSE of 0.2291 and MAE of 0.3556, outperforming centralized learning, which yielded an RMSE of 0.2438 and MAE of 0.3687. It represents a 2.25% improvement in accuracy, underscoring the benefits of federated learning decentralized architecture. Federated learning ability to aggregate knowledge from diverse client models trained on local data allows it to capture a wider range of traffic patterns. This diversity introduces variations in local models, enhancing the global model ability to learn robust and generalized representations of network traffic behavior. In contrast, centralized learning lacks this diversity, relying on a single dataset, which limits its ability to generalize across varying traffic conditions.

One key benefit of federated learning is its scalability and privacy-preserving nature. Training models locally and aggregating updates at the server level avoids transferring raw data, making it an ideal solution for scenarios requiring strict data confidentiality, such as 5G networks. However, federated learning introduces complexity in synchronizing and aggregating models across multiple clients, which can increase computational complexity. Despite this, the distributed nature of federated learning ensures that the system remains scalable and capable of handling the demands of large-scale 5G networks while offering improved predictive accuracy.

Fig. 9 and Fig. 10 provide insights into the model predictions under centralized and federated learning setups. Both setups show the LSTM+GRU hybrid model performing well across a range of bitrate values. However, the performance in regions with lower bitrates reveals a notable challenge. The predictive accuracy drops near zero bitrates, indicating that the model struggles to detect meaningful patterns in this data range. This drop in performance is likely due to sparse or noisy data in these regions, where signal characteristics are less distinct. Such underfitting in low-bitrate areas highlights a common limitation in machine learning models when dealing with sparse or low-intensity data.

Addressing this challenge would involve strategies such as augmenting the training dataset to include more low-bitrate cases, ensuring the model encounters these scenarios during training. Another approach could involve using specialized techniques that enhance the model sensitivity to sparse data regions, such as weighted loss functions or regularization techniques tailored for imbalanced datasets. These enhancements would help mitigate underfitting and improve the model robustness, enabling more accurate predictions across the entire bitrate spectrum. By doing so, federated learning could further solidify its position as a scalable and effective solution for 5G network traffic prediction, particularly when paired with architectures like the LSTM+GRU hybrid model that captures complex patterns.

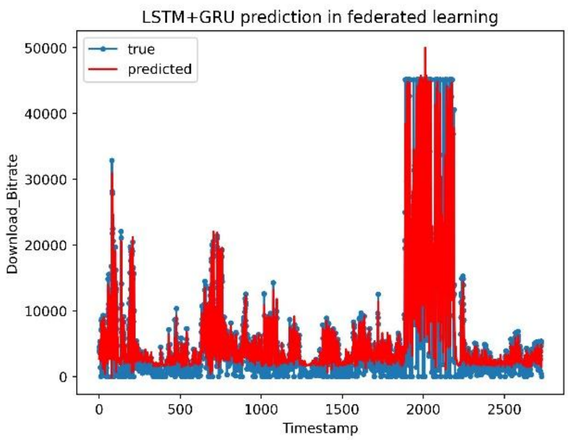


Fig. 9. LSTM+GRU model prediction in federated learning.

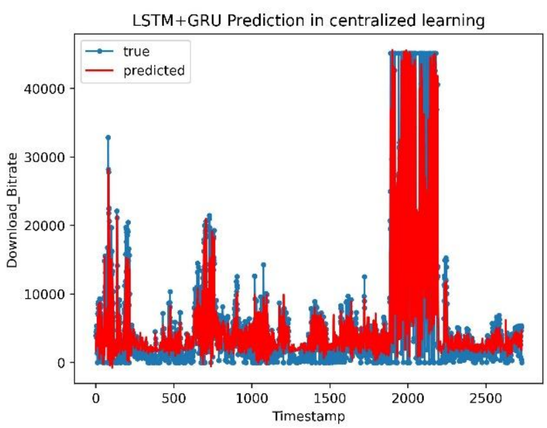


Fig. 10. LSTM+GRU model prediction in centralized learning.

### D. Data Splitting Comparison

The preprocessed data was divided into three different ratios (80:20, 85:15, and 90:10). The three different ratios were compared in federated learning and centralized learning, and the ratio 90:10; 90% of the data for training and the remaining 10% of the data for the test yielded the best results compared to the other two ratios, as shown in Table IV, this experiment demonstrated that training on a larger proportion of data allows the model to capture more patterns and nuances in the data, ultimately leading to a better understanding of the underlying structure and relationships. The model can generalize well and perform with lower prediction errors.

TABLE IV. PERFORMANCE OF THE MODEL IN DIFFERENT SPLITS OF THE DATASET

| Learning Setting | Federated Learning | | | Centralized Learning | | |
|---|---|---|---|---|---|---|
| Test Size/Measure | 10% | 15% | 20% | 10% | 15% | 20% |
| RMSE | 0.2291 | 0.3430 | 0.3553 | 0.2438 | 0.3444 | 0.3471 |
| MAE | 0.3556 | 0.4309 | 0.4448 | 0.3687 | 0.4363 | 0.4429 |
| R2 | 0.845 | 0.818 | 0.805 | 0.826 | 0.817 | 0.811 |

## VI. CONCLUSION

Building high-quality traffic prediction models with effective generalization is an inherently complex task, given the diverse data patterns that characterize 5G network traffic. This paper studies the challenge of predicting 5G network traffic using a hybrid LSTM+GRU model along with a federated learning approach. The hybrid model outperformed the standalone LSTM and GRU models, thus proving its capability to capture both long- and short-term dependencies within the data. At the same time, the federated learning approach adds another dimension to privacy by letting the system learn from varied data on different clients without compromising data privacy. In addition, it produced lower prediction errors with better generalization than centralized learning; thus, it would be an efficient and scalable solution under resource allocation optimization towards network performance enhancement and quality-of-service improvement in complex 5G environments while preserving data confidentiality.

## ACKNOWLEDGMENT

The authors extend their gratitude to the African Union Commission for their financial contribution and also to the members of the Pan African University Institute for Basic Sciences, Technology, and Innovation (PAUSTI), Nairobi, Kenya.

## REFERENCES

[1] Su, J., Cai, H., Sheng, Z., Liu, A., & Baz, A. (2024). Traffic prediction for 5G: A deep learning approach based on lightweight hybrid attention networks. Digital Signal Processing, 146, 104359. https://doi.org/10.1016/j.dsp.2023.104359.

[2] Saha, Sajal, "Toward building an intelligent and secure network: An internet traffic forecasting perspective" (2023). Electronic Thesis and Dissertation Repository. 9556. https://ir.lib.uwo.ca/etd/9556.

[3] (Alekseeva, Stepanov et al. 2021) Chih-Lin and J. Huang, "RAN revolution with NGFI (xHaul) for 5G," 2017 Optical Fiber Communications Conference and Exhibition (OFC), Los Angeles, CA, USA, 2017, pp. 1-4.

[4] M. Alias, N. Saxena and A. Roy, "Efficient cell outage detection in 5G HetNets using Hidden Markov Model," in IEEE Communications Letters, vol. 20, no. 3, pp. 562-565, March 2016.

[5] Yue, B., Fu, J., & Liang, J. (2018). Residual recurrent neural networks for learning sequential Representations. Information, 9(3), 56. https://doi.org/10.3390/info9030056.

[6] E. Selvamanju and V. B. Shalini, "Deep learning based mobile traffic flow prediction model in 5G cellular networks," 2022 3rd International Conference on Smart Electronics and Communication (ICOSEC).

[7] Mahajan, Smita & Ramachandran, Harikrishnan & Kotecha, Ketan. (2022). Prediction of network traffic in wireless mesh networks using hybrid deep learning model. IEEE Access. PP. 1-1. 10.1109/ACCESS.2022.3140646.

[8] Shafiq, M. O., Zhu, Z., Yu, X., & Liu, A. (2018). A survey on network traffic prediction techniques in communication networks. Journal of Network and Computer Applications, 100, 113-125.

[9] Deng, S., Zhao, H., Fang, W., Yin, J., Dustdar, S., & Zomaya, A. Y. (2020). Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence. IEEE Internet of Things Journal, 7(8), 7457-7469.

[10] Wang, S., Zhang, X., Zhang, Y., Wang, L., Yang, J., & Wang, W. (2019). A survey on mobile edge networks: Convergence of Computing, Caching, and Communications. IEEE Access, 7, 171676-171719.

[11] Qiao, G., Yu, G., & Ding, Z. (2020). Deep Reinforcement Learning for Cooperative Content Caching in 5G Networks. IEEE Transactions on Wireless Communications, 19(6), 3977-3991.

[12] Zhu, H., Gao, L., Li, W., & Yu, X. (2020). Federated Learning for 5G Vehicular Internet of Things: Opportunities and Challenges. IEEE Wireless Communications, 27(2), 12-18.

[13] Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., & Ivanov, V. (2019). Towards federated learning at scale: System design. Proceedings of the 2nd SysML Conference, 1-15.

[14] Wang, J., Kang, J., Lin, X., & Wang, S. (2019). When edge meets learning: Adaptive control for Resource-constrained distributed machine learning. IEEE INFOCOM 2019 - IEEE Conference on Computer Communications, 63-71.

[15] A. G. Reddy, S. Sinha, P. A. Reddy and C. Vimala, "Network traffic prediction for 5G network," 2023 International Conference on Recent Advances in Electrical, Electronics, Ubiquitous Communication, and Computational Intelligence (RAEEUCCI), Chennai, India, 2023, pp. 1-5, doi: 10.1109/RAEEUCCI57140.2023.10134497.

[16] M. Chen, X. Wei, Y. Gao, L. Huang, M. Chen and B. Kang, "Deep-broad learning system for traffic flow prediction toward 5G cellular wireless network," 2020 International Wireless Communications and Mobile Computing (IWCMC), Limassol, Cyprus, 2020, pp. 940-945, doi: 10.1109/IWCMC48107.2020.9148092.

[17] Gao Z. 5G Traffic prediction based on deep learning. Comput Intell Neurosci. 2022 Jun 24;2022:3174530. doi: 10.1155/2022/3174530. PMID: 35785055; PMCID: PMC9249458.

[18] Q. Zeng, Q. Sun, G. Chen, H. Duan, C. Li and G. Song, "Traffic prediction of wireless cellular networks based on deep transfer learning and cross-domain data," in IEEE Access, vol. 8, pp. 172387-172397, 2020, doi: 10.1109/ACCESS.2020.3025210.

[19] D. Alekseeva, N. Stepanov, A. Veprev, A. Sharapova, E. S. Lohan, and A. Ometov, "Comparison of machine learning techniques applied to traffic prediction of real wireless network," in IEEE Access, vol. 9, pp. 159495-159514, 2021, doi: 10.1109/ACCESS.2021.3129850.

[20] Perifanis, V., Pavlidis, N., Koutsiamanis, R., & Efraimidis, P. S. (2022). Federated learning for 5G base station traffic forecasting. ArXiv. https://doi.org/10.1016/j.comnet.2023.109950.

[21] V. Perifanis et al., "Towards energy-aware federated traffic prediction for cellular networks," 2023 Eighth International Conference on Fog and Mobile Edge Computing (FMEC), Tartu, Estonia, 2023, pp. 93-100, doi: 10.1109/FMEC59375.2023.10306017.

[22] Z. Du, C. Wu, T. Yoshinaga, K. -L. A. Yau, Y. Ji, and J. Li, "Federated learning for vehicular Internet of things: Recent advances and open issues," in IEEE Open Journal of the Computer Society, vol. 1, pp. 45-61, 2020, doi: 10.1109/OJCS.2020.2992630.

[23] Ali. M. A., Zhuang, H., Ibrahim, A.Rehman, O., Huang, M, and Wu, A. A machine learning approach for the classification of kidney cancer subtypes using miRNA. Genome Data. Appl. Sci. 2018, 8, 2422; doi:10.3390/app8122422.

[24] Dridi, Hinda & Ouni, Kais. (2020). Towards robust combined deep architecture for speech recognition: Experiments on TIMIT. International Journal of Advanced Computer Science and Applications. 11. 10.14569/IJACSA.2020.0110469.

[25] Okut, Hayrettin'' Deep Learning: Long-Short Term Memory''(2021).

[26] Yue, B., Fu, J., & Liang, J. (2018). Residual recurrent neural networks for learning sequential representations. Information, 9(3), 56. https://doi.org/10.3390/info9030056.

[27] E. Haque, S. Tabassum and E. Hossain, "A Comparative analysis of deep neural networks for hourly temperature forecasting," in IEEE Access, vol. 9, pp. 160646-160660, 2021, doi: 10.1109/ACCESS.2021.3131533.

[28] Toledo, M.L., & Rezende, M.N. (2020). Comparison of LSTM, GRU and hybrid architectures for usage of deep learning on recommendation systems. Proceedings of the 4th International Conference on Advances in Artificial Intelligence.

[29] Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: principles and practice (2nd ed.). OTexts.

[30] Raca, D., Leahy, D., Sreenan, C.J., & Quinlan, J.J. (2020). Beyond Throughput: The Next Generation a 5G Dataset with Channel and Context Metrics.