An efficient Multi-Layer Hybrid Neural Network and optimized parameter enhancing approach for traffic prediction in Big Data Domain

Supriya H.S[1], Dr. Chandrakala B. M[2]

[1]Research Scholar, Department of Computer Science, DSCE, Bengaluru[1]
[2]Associate Professor, Department of ISE, DSCE, Bengaluru[2]
[1]supriyahs_12@rediffmail.com, [2]chandrakalabm-ise@dayanandasagar.edu

ABSTRACT

Network Traffic Prediction and analysis is one of the essential tasks for cellular networks to reduce the network load. Moreover, traffic data are categorized as "Big Data" for developing a prediction framework to achieve higher metrics. Furthermore, the recent adoption of deep learning has been trending research, concerning cross-domain like Big Data and cellular networks; however, there are still several gaps considering the higher prediction rate and minimal error rate. This research work develops Multi-Layer Hybrid Network (MLHN) for network traffic prediction and analysis; MLHN comprises the three distinctive networks for handling the different inputs for custom feature extraction. The first network takes the network traffic sequence as the input for a given specific period. The second network takes input as the references along with data that corresponds to the date and time of traffic and the third input take the cross-domain parameter for understanding and exploiting the deep features. Furthermore, an optimized and efficient parameter-tuning algorithm is introduced to enhance parameter learning which tends to reduce error. MLHN is evaluated considering the "Big Data Challenge" dataset considering the Mean Absolute Error, Root Mean Square Error and R^2 as metrics; furthermore, MLHN efficiency is proved through comparison with state-of-art approach.

Keywords: Multi-Layer Hybrid Network, Big Data, Traffic Prediction.

1 INTRODUCTION

Big data is categorized into two types known as unstructured data and semi-unstructured data, which evolved with the rise of Internet technologies and computer science, generating huge amounts of big data from various domains [1]. Big data typically has been categorized into five Vs: volume, variety, velocity, and Veracity. In recent years, big data has been incorporated into a variety of study fields, stimulating advancements in the corresponding theories and technologies [1]. However, by utilizing the given knowledge the information has been made available for informative big data, supporting a new or greater awareness of the targeted problems, questioning and even transforming the fundamental hypotheses supported by conventional facts. The establishment of such a cellular network system is a crucial transition in Industry 5.0 with the advent of fifth-generation networks (5G). The most recent research indicates that by 2023, there will be 1.4 billion 5G connections, up from just around more than

25 million in 2022 [2]. The vision of the Internet of Everything and sixth-generation (6G) networks will enable this phenomenal growth trend to continue in the next years. The advanced cellular network is capable of enhancing the data and functioning of the system and its interoperability, which is one of the key enabling technologies in Industry 5.0, constructed on these substantial and dependable communication services as well as the ultra-low-latency services that are in their support.

Data traffic has caused significant issues because of mobile carriers' rising popularity and internet access because the network is under constant strain. As this network control and management along with provisioning service for network traffic analysis and prediction, this assumes to be a crucial part of cellular networks that reduces the load. Numerous studies have been done on cellular network traffic prediction, however, there are significant difficulties concerning the temporal volume and geographical dynamics that various Internet users' behavior is accommodated [3]. Cellular traffic is generated by a variety of devices used each day and businesses associated with them, particularly cellphones in these circumstances. Numerous innovative and cutting-edge studies make use of such technologies. Recently, new types of mobile traffic, such as mobile traffic classification [4], mobile traffic prediction and characterization [5], etc., have appeared. An accurate cellular network traffic prediction advances which makes it crucial each day and assist the network management system to deal with unexpected problems by proactively accommodating resources that are delivered to consumers along with industrial devices and high-end services [6]. Figure 1 shows the Intelligent Traffic prediction model that comprises the three blocks i.e. network application center, cellular usage data collection center, and Management center each block is interconnected with each other to predict the cellular traffic.

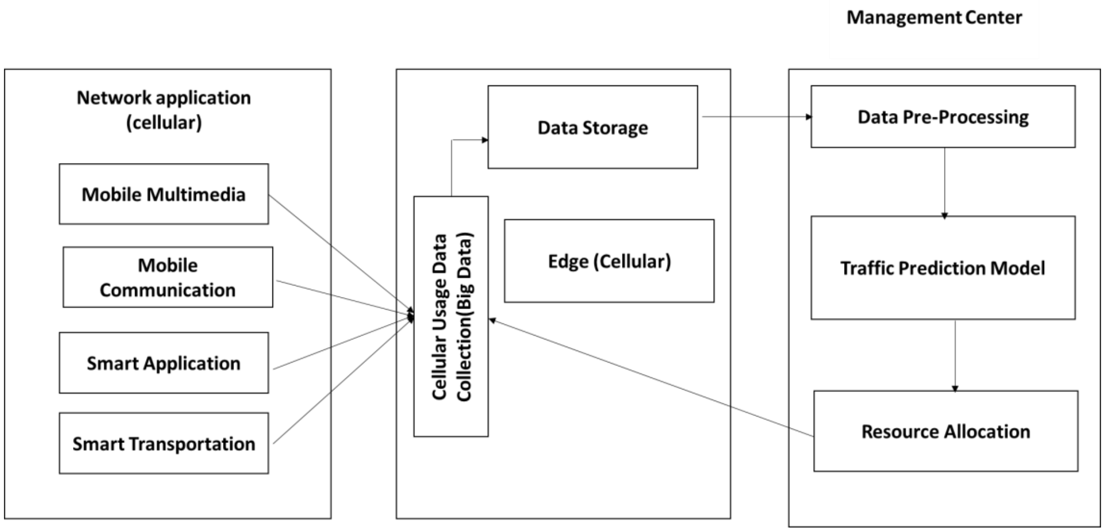

Figure 1 Intelligent Traffic prediction model

Network traffic prediction has become challenging with each passing day, the major points considered here are:

1) **Data heterogeneity**: Internet traffic is typically heterogeneous in the temporal dimension, and a particular graphical region may show distinct traffic patterns at different times of the day.

As a result, it is difficult to suggest a general prediction model that can accurately anticipate other traffic datasets.

2) **Anomalous data**: Internet traffic in the real world is subject to many internal and external influences, which results in non-stationary complex traffic patterns. Additionally, those events may contain data points that fall outside the parameters of the data distribution, which would limit the model's capacity to generalize during deployment. Therefore, before utilizing them to train the prediction model, the outliers or anomalous data points must be handled [7].

3) **Data scarcity**: In the first step, the prediction model is trained with a large dataset and ensures that it can generalize the pattern well; improved prediction on the testing set is required. However, managing a sizable history dataset to train the prediction model is difficult, resulting in subpar performance. The performance of cellular traffic prediction has depreciated, and artificial intelligence has been proposed as a prospective improvement. Time-series prediction has been shown to benefit from deep learning techniques that as long short-term memory (LSTM) and neural networks like convolutional neural networks (CNN). The traffic consumption information for cellular traffic is logged in as the time series in the spatiotemporal domain. Extensive studies employing deep learning methods, which are greatly improvised and enhance the performance of cellular traffic prediction in comparison with the traditional statistic-based approaches. However, the majority of this present work is to focus on processing network cellular traffic data in the temporal domain because of the constraints of neural network-based concepts. Recently, numerous studies have been combined for various convolution and recurrent operation techniques to focus on [8]-[11].

1.1 Motivation and Contribution of research work

Recent developments in machine learning models [7] are capable of solving complex challenges in comparison with traditional statistical models in cellular network traffic prediction [8], [9] by incorporating the complex and nonlinear dependencies embedded in cellular wireless traffic data. In [8] a deep belief network that is based on the prediction approach represents the cellular traffic's long-term reliance. By combining an auto-encoder and Long Short-Term Memory (LSTM) network, [9] we have developed a method to take advantage of distinct cells' spatial dependence [10]. However, lossy representations of the original data and the learned features from the auto-encoder may not accurately capture the spatial dependence of surrounding cells [11]. Additionally, the aforementioned techniques largely focus on estimating traffic for a single cell. If used on citywide scale networks, they are computationally expensive

Furthermore, motivated by the aforementioned phenomena, this research develops a novel deep learning architecture for traffic prediction and analysis, further contribution of research work is given as follows:

1. This research work develops a Multi-Layer Hybrid Neural Network (MLHN) architecture for traffic prediction and analysis; MLHN comprises multiple layers for the different types of input in the network for efficient feature extraction.

2. An optimized parameter-tuning algorithm is introduced that aims to automate the parameter for better prediction.
3. MLHN is evaluated considering the “Big Data Challenge “ dataset; evaluation is carried out on various services such as SMS, Call, and Internet; efficient analysis is carried out considering the sub-services like Incoming calls and outgoing calls, Received SMS, and sent SMS.
4. MLHN is compared with different state-of-art techniques considering the various metrics.

This particular research is organized as follows: The first section of the research starts with Big Data background and deployment of Big Data in cellular traffic prediction; the section ends with motivation and contribution of research work. The second section focuses on various existing traffic prediction models along with their disadvantages. The third section presents the proposed methodology of MLHN along with algorithm and mathematical architecture; the Fourth section presents the MLHN evaluation considering different metrics.

2 RELATED WORK

The deep learning technique on network traffic prediction, however, the research conducted in both fields of academics and industry has dedicated a lot of effort to this approach for wireless cellular network traffic prediction. The deep learning approach extracts the complex patterns, which are hidden via the cellular data, also known as DenseNet [11]. A temporal model known as the spatial-temporal neural network model is proposed in [12]. This exhibits an accurate cellular traffic forecasting systematic architecture. A Double STN (D-STN) model is implemented in this paper, which employs a lightweight mechanism responsible to solve the purpose of STN output along with this the historical statistics, henceforth improvising the performance of the model. On the other hand, the meta-learning model can adapt or extend to new situations as well as tasks. Because of this, we introduce model-agnostic meta-learning in this letter (MAML) [13] for predicting wireless traffic using the FL mechanism to effectively estimate wireless traffic at the edge. The deep learning mechanism is proposed here as an ensemble mechanism for different predictors of prediction type mechanisms.

However, the deep reinforcement learning mechanism is used to appropriately predict the accurate performance of prediction. [14] A cross-service-based mechanism for fusion transfer learning strategy to make use of many cross-domain datasets to improve the performance of prediction. In [15] a spatiotemporal time-independent deep learning mechanism for cellular network traffic prediction mechanism incorporates the attention mechanism for architecture design purposes. By consideration, in [16] a network traffic prediction model, known as STGCN-HO, the approach is proposed by using the graph along this residual network structure to enhance the prediction performance. A novel approach is proposed in [17] for network traffic prediction known as FedDA, which employs a federated learning mechanism that trains the wireless network traffic prediction model collectively. The dual mechanism is known as the intra-cluster and the inter-cluster is known as the global model.

In [18] five various kinds of deep learning methods are used for resolving wireless network traffic usage forecasting known as one AP for substantial spatial correlations exist. The researchers collaborate the statistical tools along with these deep learning models as well as predicting the accuracy. In [19] a single-cell level cellular traffic prediction mechanism by the LSTM approach with Gaussian Process Regression (GPR) enhances the performance. In this approach, periodic components are extracted based on the utilization analysis carried out. Here LSTM is used for this learning process as well as a long-term dependency for random values and GPR is applicable here to estimate the residual random components for the network traffic prediction mechanism as proposed in [20]. This aims at enhancing the accuracy of fluctuation based on pattern clustering. An LSTM-based novel approach known as TPBLN, this mechanism is used for the prediction of baseline components, wherein the remaining component is predicted by a probability model, and this parameter is evaluated based on incorporating maximum likelihood determination.

3 PROPOSED METHODOLOGY

Network Traffic Prediction and modeling is considered the high-quality telecom service for cellular networks and has attracted fair attention due to the application. However, it has been proved one of the challenging tasks due to several reasons as discussed earlier. Furthermore, considering the diverse network, cellular traffic prediction possesses a dynamic range and most of the existing model considers traffic patterns as the probabilistic distribution, which makes it beneficial to form the closed solution. However, the prediction error has been quite high; also earlier research indicated the adoption of deep learning could be highly efficient, thus, this section develops MLHN (Multi-Layer Hybrid Network) for Traffic Prediction.

3.1 Dataset initialization and problem formulation

The network traffic here depicts a particular service type known as $u \in$ (msg, call, Internet) denoted as a time-dependent sequence for target points, $T_U = \{ T_{U,V} | \text{ v } = 1, 2 ... \text{V}\}$, here V is the sum of time intervals. $T_{U,V}$ Is the set of cellular network traffic matrix for vth period in a geographical position as depicted by $X * Y$ cells depicted as

$$T_{U,V} = \begin{bmatrix} t_{u,v}^{(1,1)} & t_{u,v}^{(1,2)} & \cdots & t_{u,v}^{(1,Y)} \\ t_{u,v}^{(2,1)} & t_{u,v}^{(2,2)} & & t_{u,v}^{(2,Y)} \\ t_{u,v}^{(X,1)} & t_{u,v}^{(X,2)} & & t_{u,v}^{(X,Y)} \end{bmatrix} \quad (1)$$

Here $t_{u,v}^{x,y}$ estimates the network prediction in a cell with ordinates $(x * y)$ the sequence is determined as the parameter $T_U \in H^{V \times \text{X} \times \text{Y}}$, the initial analysis apart from the time-dependent area is appropriate for the specific kind of network cellular traffic dataset used. This ensures readability the service type notion is omitted here in this equation. $T_{U,V} = T_v$ and $t_{u,v}^{x,Y} = t_v^{x,Y}$ until not specific. By, careful analysis of network cellular traffic dataset, the time-dependent dynamics adjacent correlation analysis to be detected, temporal dynamics for various kinds of network cellular traffic detected in target cells.

$$U_a = \frac{\sum_{v=1}^{V-a}(N_v^{(r,s)} - N^{-(r,s)})(N_{v+a}^{(r,s)} - N_v^{-(r,s)})}{\sum_{v=1}^{V-a}(N_v^{(r,s)} - N_v^{-(r,s)})^2} \qquad 0 \leq \text{a} \leq \text{V} \quad (2)$$

Here, $N^{-(r,s)}$ this denotes the mean value of the cell over the specific time duration. The network cellular traffic shows non-zero autocorrelations for the time-duration that indicates the traffic volume prediction via previous analysis. The metric to estimate the spatial correlation is articulated as:

$$\delta = \frac{covariance(N^{(r,s)}, d^{(r',s')})}{\partial_{N^{(r,s)}} \partial_{d^{(r',s')}}} \tag{3}$$

The $covariance(.)$ denotes the covariance function and ∂ denotes the standard deviation function; however, this evidence shows that the spatial correlation exists amongst various groups of cells.

The network traffic volume states that it's not only associated with spatiotemporal factors along the external factors like the estimated count like Interface and Base station; here we select two distinctive parameters for Base station and Interface as B_{st} and $Infs$.By considering the sum of B_{st}a particular cell determines the traffic load that is carried out. The number of B_{st}in a cell estimates the network traffic load, which is carried out thoroughly. The network cellular traffic load reaches the maximum value, which shows that the traffic volume is static however; the additional users enter the cell. To accomplish the estimated cellular traffic prediction, via multiple aspects that influence the basis of cross-domain datasets. The various datasets needed to be characterized the network prediction from various aspects. The number of B_{st} and $Infs$. Decide the actions of a cell that is responsible for directly affecting the user's requests for its ability in telecommunication services. The three type's cross-domain datasets are considered here, the B_{st} data, the $Infs$. Distribution of social activity level. The dataset here uses the B_{st} is determined from Open-Cell ID. The software gathers data from cellphones around the world. The dataset consists of different types of data about the B_{st} like the position of the mobile where it is located the country code the accurate series of each B_{st}. The geographical positioning of the individual cell which maps the position of every B_{st} to each cell where the $B_{st}$is positioned once preprocessing is done. The total number of B_{st} for all the cells is determined, however B_{st} is given by the equation below:

$$T_{B_{st}} = \begin{bmatrix} t_{B_{st}}^{(1,1)} & t_{B_{st}}^{(1,2)} & \cdots & t_{B_{st}}^{(1,Y)} \\ t_{B_{st}}^{(2,1)} & t_{B_{st}}^{(2,2)} & & t_{B_{st}}^{(2,Y)} \\ t_{B_{st}}^{(X,1)} & t_{B_{st}}^{(X,2)} & & t_{B_{st}}^{(X,Y)} \end{bmatrix} \tag{4}$$

The POIs data is distributed and used across the google places API. The different mechanisms used for the data include subway stations, stores, restaurants, etc. the elaborated analysis of the category is summed up or the final representation. The matrix produced through Interface is given as:

$$T_{POI} = \begin{bmatrix} t_{Infs.}^{(1,1)} & t_{Infs.}^{(1,2)} & \cdots & t_{Infs.}^{(1,Y)} \\ t_{Infs.}^{(2,1)} & t_{Infs}^{(2,2)} & & t_{POI}^{(2,Y)} \\ t_{Infs.}^{(X,1)} & t_{Infs.}^{(X,2)} & & t_{Infs.}^{(X,Y)} \end{bmatrix} \tag{5}$$

The Data collected gathers user-generated data via Twitter, such as the position or keywords. The Pre-processing of sociality is denoted through the matrix as:

$$T_{soc}\begin{bmatrix} t_{soc}^{(1,1)} & t_{soc}^{(1,2)} & \cdots & t_{soc}^{(1,Y)} \\ t_{soc}^{(2,1)} & t_{soc}^{(2,2)} & & t_{soc}^{(2,Y)} \\ t_{soc}^{(X,1)} & t_{soc}^{(X,2)} & & t_{soc}^{(X,Y)} \end{bmatrix} \quad (6)$$

Here, $t_{soc}^{(x,y)}$ determines the sociality of the cell(x, y). The complexity to determine the various types of cross-domain datasets that is adjacent and consists of the standard data. The static datasets are considered here because it frequently changes over a specific period. The model is trained here using the datasets and the heat maps produced along three types of datasets with the city topology. The cross-domain dataset for similar spatial distribution in comparison with the network traffic for various services. More traffic is generated in the city center than in rural areas.

The spatial correlation across the datasets used for network cellular traffic. This spatial correlation is calculated by equation (3), network cellular traffic is determined which is utilized at its peak value. The different types of cellular traffic for specific similarity levels, the pattern knowledge learning the type of network traffic data. The Bst is distributed across the correlated cellular traffic in comparison with other datasets. The number of Bst in a cell is indeed the traffic comparison via other datasets. The sum of Bst is used by the traffic generated by using features that facilitate network traffic prediction the Interface categories and sociality.

3.2 Problem Formulation

The communication network is decentralized across geographically distributed areas. Across every region, the client holds the wireless network traffic data by considering the local model update. In each region, a local client records the traffic that pairs the local model update $F = \{1, \dots \dots, t \dots., T\}$ depicts this client set, and then t is the index and T the total count of adjacent clients. This adjacent traffic information is segmented into g time slots, T_m is the parameter depicting the network traffic volume and the close dependence shown by $K_m = \{T_{m-n}, T_{m-n+1,}, \dots, T_{m-1}\}$ given as the input feature, wherein n is shown as the set of data points. However, T_m is given as the estimated target, which is labeled as given as the output j_m , by considering the prediction ahead of one step. The input-output pair is summed as { K_m, j_m) determined by the sliding –window mechanism. The randomly created types of samples N_k differs apart each client to client and zero sample approximation aimed at each client, the sample is responsible for each client to another. The training dataset of the $m - th$ client, β_g is partitioned along the support setβ_g^u, the collection of queries given asβ_g^r. The knowledge is secured by internally transferring it by β_g^u to β_g^r. By aggregation of the local models communicate with the central server and accommodate the features of the global model to the central server for each model of the local client. The main aim here is to determine the network traffic prediction to generate a model via a constraint α that reduces the loss function for the dataset depicted as, j_m and the comparison value as j_m.

$$ {}^{Red}\frac{1}{\tau^{G}} \sum_{g=1}^{G} T\left(\alpha; \beta_g\right), \tag{7} $$

Here $T\left(\alpha; \beta_g\right)$ shows the loss function determining the network traffic volume j_m . By considering the mean squared error (MSE) as the metric, the loss function is determined by:

$$ T\left(\alpha; \beta_g\right) = \frac{1}{M_g} \sum_{\{a_m, b_m\} \in \beta_g} (j_m - j_m)^2 \tag{8} $$

The traditional network traffic prediction targets and train the model by accommodating all clients, the main aim here is to determine a model, which adapts to a rapid heterogeneous distribution circumstance. By managing and minimizing the loss occurrence amid the two values of each client's network traffic value predicted based on this proposed model. This technique is built by preceding one among the few steps responsible for fine-tuning the dataset. This is stated by below equation:

$$ {}^{Red}\frac{1}{\tau^{g}} \sum_{g=1}^{G} T(\alpha - \partial), \sum_{k=0}^{K-1} \rho_\alpha T(\alpha_{g,s}; \beta_g^u); \beta_g^r), \tag{9} $$

Here $\rho_\alpha T(\alpha_{g,s}; \beta_g^u)$ depicts the adjacent gradient of the $s - th$ steps for the update. $s \in [0, \mathrm{S})$. $\alpha - \partial \sum_{k=0}^{K-1} \rho_\alpha T(\alpha_{g,k}; \beta_g^u)$ Is the fine-tuned model obtained preceding the query-set β_g^u. The equation (9) is used for minimizing the average loss for the fine-tuned model on the query-set β_g^r, by adapting new tasks for the model to adapt in.

3.3 Data Processing

The network traffic prediction at a certain period goes missing while transmission or the data storage error. The data completed needless to be completed before moving to the next stage for cellular network traffic prediction. A standard approach considered here is, to sum up, the missing values of each cell as (x, y). The parameters depicted here show the mean network traffic volume values around its adjacent cells. This is given by

$$ T_v^{(x,y)} = \sum_{k \in [-1,1]} \sum_{m \in [-1,1]} \frac{T_v^{(x+k, l+m)}}{8} \tag{10} $$

3.4 Prediction Model

Our proposed model is fed by three inputs. the first input is the ($T_{v-1}, \ldots, T_{v-o}, o \in \text{Ƥ}$) this sequence of the network traffic matrix before the specified period. The second input sequence here (j) references along the data for date-time corresponding to the time-duration on a particular hour on a day of a week. The last input provided here depicts the cross-domain datasets by incorporating $T_{B_{st}}$, and Interface distribution as T_{INFS} as well as the T_{soc} as sociality. To handle various inputs relevant to the data and their features, three types of neural networks are formulated as below:

3.4.1 First Layer Modelling

The first input fed here is data that consists of o number of frames where each frame consists of a one-channel image. This is a CNN model that consists of strong capabilities to model this spatial interdependence, which efficiently fuses the localized information area that necessarily features extraction for particular tasks. This time sequence of the data is specifically not caught via the CNN model. These LSTM networks model the sequence of information for the cellular traffic. A dual layer customized-Integrated Neural Network (CINN) network is formulated via a parallel developed model for Spatio-temporal interdependencies for the particular sequence.

Each cell here consists of a memory cell μ that gathers the relevant state information. This cell is accessed and altered via three controlling gates the input gate IG, forget gate FG, and the OG. However here a new input is fed to the CINN unit, and the data gathered here is stored in the μ if the IG is activated here. Parallelly the past cell status is adapted here and is forgotten via the process if the FG is switched on. ß is controlled via the output gate, which is the final input state. This states whether the output of the cell μ is proliferated via a final state or not. The main functionality here states that for a particular frameT_{v-m}, here $m \in$ {1, 2,….., o} where ∂(.) shows the activation function. , whereas * denotes the convolutional operation and ∘ is the Hadamard product

$$IG^{C} = \partial(WT_{tI} * T_{\Upsilon} + WT_{xI} * ß_{\Upsilon-1} + T_{cI} \circ \mu_{\Upsilon-1} + q_I), \qquad (11)$$

$$FG^{C} = \partial(WT_{tF} * T_{\Upsilon} + WT_{xF} * ß_{\Upsilon-1} + T_{cF} \circ \tanh(WT_{t\mu} * T_{\Upsilon} + q_F),$$

$$\mu_{\Upsilon} = \partial(FG^{C} \circ \mu_{\Upsilon-1} + IG^{C} \circ\circ \tanh(WT_{t\mu} * T_{\Upsilon} + T_{xc} * \mu_{\Upsilon-1} + q_c),$$

$$OG^{C} = \partial(WT_{tO} * T_{\Upsilon} + WT_{xO} * ß_{\Upsilon-1} + T_{cO} \circ T_{\Upsilon} + q_O),$$

$$ß^{C} = \partial \circ \tanh(T_{\Upsilon}).$$

In the above-stated equation $WT(.)$ and $q(.)$ are the biased weights that are learned. Besides tanh(.) here indicates the hyperbolic tangent function that works as a non-linear transformation input. The gatesIG^{C}, FG^{C}, μ_{Υ}, OG^{C}, $ß^{C}$ here in this CINN unit along the tensors that are dimensioned. The output stated here of the CINN is shown as the $\delta_{zv} \in H^{FM*X*Y}$, where FM shows the Feature Maps.

3.4.2 Second Layer Modelling

When the mobile users ask for services, the date and time of this network cellular traffic information are recorded, the data is extracted and shown as the features, four types of data are extracted as i.e., which day of the week, which hour of the day, whether it is a weekend or a feature vector e . This feature vector here is fed to a dual-layer network where the dimensionality of e is upgraded from 4 to$FM * X * Y$. This equation for feature vector is mathematically stated as δ_{data}

$$\delta_{data} = \partial(WT_{Data}^{2} \partial(WT_{Data}^{1} e + q_{Data}^{1}) + q_{Data}^{2}), \qquad (12)$$

WT_{Data}^{p} and q_{Data}^{p} are the learning parameters at the p-the layer where $p \in \{1, 2\}$. After this operation is reframed the output is given by δ_{data}.

$$\delta_{data} = RF_{\delta_{data}}, \quad (13)$$

3.4.3 Third Layer modeling

The model is affected via external factors by traffic generated and learns these representations with the cross-domain datasets, a dual-layer CNN architecture is built here. In the dual-layer CNN architecture, the parameters T_{POI}, T_{soc} T_{cross} fed into a tensor through this concatenation function. Once the non-linear transformation is carried out on T_{cross}, The features are represented initially given by cross-domain datasets as δ_{cross} written as:

$$\delta_{cross\_dataset} = comp\_func(WT_{cross\_dataset} * T_{cross\_dataset}) \quad (14)$$

$$T_{cross\_dataset} = T_{Bst} \oplus T_{POI} \oplus T_{Soc}$$

Here $\oplus$ indicates the concatenation operator parallel, $WT_{cross\_dataset}$ are the weights associated with the learning parameters along optimization $comp\_func(.)$ is the composite function, which implements this batch normalization (Batch norm) function, rectified linear units (ReLU), and the convolution operation (Conv) consecutively.

3.5 Feature Learning

The outputs are fused here via the concatenation function shown below:

$$\delta = \delta_{zv} \oplus \delta_{data} \oplus \delta_{cross\_dataset}, \quad (16)$$

δ Shows the overall model of this initial feature map which serves as the input given to this DenseNet. The additional function here is processed without the recommendation for the process because it combines various information and does not gain advantage to this efficient feature learning mechanism, this model consists of B layers where each layer implements a composite function as $comp\_func_B$ similar to the cross-domain data modeling which is, $comp\_func_B = comp_{func}(.)$ other than this B indexes these layers.

To completely extract the temporal features affected by external dependencies, which affect the cellular network traffic volume, the connectivity here is organized with this component. The connection here denotes the existing direct connections by each layer with the adjacent layers. The B layers receive feature maps from all the layers below them. $\delta_0, \dots \dots . \delta_1, \dots . \delta_{B-1}$ this serves as the input.

$$\delta_b = comp_{func_b}(\delta_0 \oplus \dots . . \oplus \delta_{b-1}), \quad (17)$$

Here $\delta_0 = 0$ the output of this last layer for this component is depicted as $\delta_B \in H^{X*Y}$. After this, the final prediction is given as

$$\hat{Z} = \partial(\delta)_B \qquad (18)$$

Hence, the objective function for the proposed model is to reduce the Frobenius norm of error matrix amongst the prediction and ground truth for these cells, shown as

$$Ƥ(\acute{\Gamma}) = {}^{Min}_{\acute{\Gamma}} |\hat{Z} - Z|_{comp\_func} \qquad (19)$$

Here $\acute{\Gamma}$ represents the values of the proposed model that is trained by the optimization mechanisms.

3.6 Efficient-parameter learning and tuning

Table 1 presents efficient parameter learning and tuning algorithm, which learns the parameter tuning automatically for efficient prediction and error reduction.

Table 1 efficient parameter tuning and learning

Input	Data set α, with parameters ∂ *and* β, for the fraction of specific client Υ
Step 1	Random initialization of $\boldsymbol{\tau}$
Step 2	for each iteration v = 0,1, 2,.., do
Step 3	$\mu = Maximum(\Upsilon G, 1)$
Step 4	Sampling a set Υ_v for CL clients
Step 5	for each client $CL \in \gamma\Upsilon_v$ parallelly do,
Step 6	Upload global model parameter: $\tau^v_{CL,0} = \tau^v$
Step 7	Sampling the tasks β^r_{CL} from α^r_{CL}
Step 8	for each step $k = 1,2, \ldots \ldots, K$ *do*
Step 9	$\tau^v_{CL,k} = \tau^v_{CL,k-1} - \partial þ_{\tau_v} T(\tau^v_{CL,k-1;}\beta^u_{CL})$
Step 10	Sampling a batch of tasks from β^r_{CL} from α^r_{CL}
Step 11	Update the model with (5)
Step 12	Improvising the individual model for spatial dependencies as: $\partial^{v+1}_{CL} = \sum_{h\in\Upsilon_v} \omega^{-v+1}_{CL,h} \tau^{v+1}_{h,0}$
Step 13	Update the model: $\partial^{v+1} = \frac{1}{CL} \sum_{\mathrm{CL}\in\Upsilon_v} \tau^{v+1}_{CL}$
Output	Learned parameters $\boldsymbol{\tau}$

To initialize the parameters a global parameter $\boldsymbol{\tau}$ is initialized here, A set of $\mu = Maximum(\Upsilon G, 1)$ clients depicted as Υ_v this is randomly chosen at each iteration during the training phase, here Υ is depicted as the hyper-parameter to qualify this fraction for all the clients. For each client $CL \in \Upsilon_v$, which fits the current value parallelly initialized for the local model parameter given by $\acute{\Gamma}^v_{CL,0}$ which is reproduced by the global parameter $\acute{\Gamma}^v$. Hence a traffic prediction mechanism such as α^u_{CL} is sampled via a support set β^u_{CL} . K the gradient steps for the sampling as given by α^u_{CL}, the model is updated and transmitted along the personal knowledge. The parameters are modeled and updated along the k-this step is estimated as given below:

$$\tau_{CL,k}^{v} = \tau_{CL,k-1}^{v} - \partial þ_{\tau_v} T(\tau_{CL,k-1;}^{v}\beta_{CL}^{u}) \tag{20}$$

Parallel, a batch of tasks obtained upon sampling the query set, the local model has improvised rapidly by adjusting the sampled query data sets. The local model is updated and uploaded to the central server, which integrates the information via heterogeneous situations, given by:

$$\tau_{CL,0}^{v+1} = \tau_{CL,0}^{v} - \acute{\Gamma} þ_{\tau_v} T(\tau_{CL,k-1;}^{v}\beta_{CL}^{r}) \tag{21}$$

Here $T\left(\tau_{CL,k-1;}^{v}\beta_{CL}^{r}\right)$ determines the second-order gradient descent mechanism working on the query tasks by combining this along with this local model is updated correspondingly and internally transmitting the model parameter $\tau_{CL,k}^{v}$. By taking into account the equation (21), the sec, the meta-learning model is capable of adapting or extending to new situations and tasks, for predicting wireless traffic using this FL mechanism that efficiently estimates the wireless cellular network traffic alongside the edge of the second-order gradient descent is shown as:

$$\begin{aligned} þ_{\tau_v} T\left(\tau_{CL,k-1;}^{v}\beta_{CL}^{r}\right) &= þ_{\tau^v{}_{CL,k}} T\left(\tau_{CL,k;}^{v}\beta_{CL}^{r}\right). þ_{\tau^v}\tau_{CL,k;}^{v} \\ &= þ_{\tau^v{}_{CL,k}} T\left(\tau_{CL,k;}^{v}\beta_{CL}^{r}\right). þ_{\tau^v{}_{CL,k-1}}\tau_{CL,k}^{v}. þ_{\tau^v}\tau_{CL,k-1}^{v} \\ &= þ_{\tau^v{}_{CL,k}} T\left(\tau_{CL,k;}^{v}\beta_{CL}^{r}\right). \vartheta_{k=1}^{K} þ_{\tau^v{}_{CL,k-1}}\tau_{CL,k}^{v} \\ &= þ_{\tau^v{}_{CL,k}} T\left(\tau_{CL,k;}^{v}\beta_{CL}^{r}\right) \quad \vartheta_{k=1}^{K}(\text{Ƥ-} \quad \partial þ_{\tau^v{}_{CL,k-1}} \\ &. þ_{\tau^v}. T(\tau_{CL,k-1;}^{v}\beta_{CL}^{r})) \end{aligned} \tag{22}$$

4 PERFORMANCE EVALUATION

Big data plays an important role in network traffic management, as it comprises a huge number of data along with various factors. Moreover, considering the network traffic management concerning big data, MLHN is designed. This section evaluates MLHN considering the metrics like MAE, RMSE and R^2. Furthermore, MLHN is designed considering python as a programming language with several deep learning libraries. MLHN evaluation is proved by comparing with a model like STCNET [21] and the existing model CNNLSTM-2D [22].

4.1 Dataset Details

A wide range of experiments are taken into consideration here to consider the Telecom Italia Big Data challenge dataset [23], the call detail records from Milan city are taken into consideration by considering 100*1000 grids of the size of 235*235 square meters. There exist five types of call detail records in the dataset, these consist incoming calls, outgoing calls, SMS received sent SMS, and the internet. One call detail record is generated at each particular time for the internet connection given as one the user issue or receives a call or SMS. For internet calls, details records are generated each time the internet connection begins or ends for a duration period for data that exceeds 5 MB. The interval is given between the duration of call detail records for ten minutes. Here the length of the period for utilizing multiple resource allocation is said to be one hour and not 10 minutes. Here because the period is set for 1 hour basis and not the original CDRs are used for training and testing the proposed model. The original data shape for wireless network traffic data for each type of wireless traffic is. To utilize the spatial-temporal aggregation mechanism, the original wireless traffic data is combined into an equivalent of two parts. The input data as well as the target output data.

4.2 Metrics evaluation

MLHN (Multi-Layer Hybrid-Network) is evaluated considering the three distinctive metrics MAE (Mean Absolute Error), RMSE (Root Mean Square Error) and R^2, these are computed as below:

4.2.1 Mean Absolute Error

Mean Absolute Error is defined as the summation of absolute difference among predicted and computed values.

$$\boldsymbol{MAE} = \left(\sum_{\boldsymbol{j=1}}^{\boldsymbol{l}} \left|\boldsymbol{w_j} - \boldsymbol{v_j}\right|\right)(\boldsymbol{l})^{-\boldsymbol{1}} \tag{23}$$

In the above equation, j indicates the variable, l indicates the non-missing data points, $\boldsymbol{w_j}$ as the predicted value and $\boldsymbol{v_j}$ as actual value.

4.2.2 Root Mean Square Error

Root Mean square Error or RMSE is defined as the measure of difference among values predicted through MLHN and the value observed.

$$RMSE = \sqrt{\left(\sum_{j=1}^{l} (v_j - v_j')^2\right)(N)^{-1}} \tag{24}$$

In above j indicates the variable, l indicates the non-missing data points, v_j' indicates the time series (estimated) and v_j indicates the time series (actual)

4.2.3 R-squared

$\boldsymbol{R^2}$ or R-squared can be defined as the ratio of unexplained variation denoted as ρ and total variation ρ'

$$R^2 = 1 - \frac{\rho}{\rho'} \tag{25}$$

4.3 Incoming and Outgoing call evaluation

In this subsection, the network traffic prediction is evaluated here for incoming calls and outgoing calls; Figure 1 shows the MAE comparison of various models considering incoming and outgoing calls as service. Moreover, figure 2 suggests that Densenet and DenseNet-fusion model observes high MAE with 16.1291 and 11.91 for incoming call service, and considering outgoing call service; both models observe 15.4935 and 9.49 respectively. Other models like STCNet perform average. Furthermore, the LSTM model shows good performance especially 3.29 and 2.91 respectively for incoming and outgoing. However, in comparison with this entire model, the proposed model MLHN observes 2.75 and 1.75 respectively.

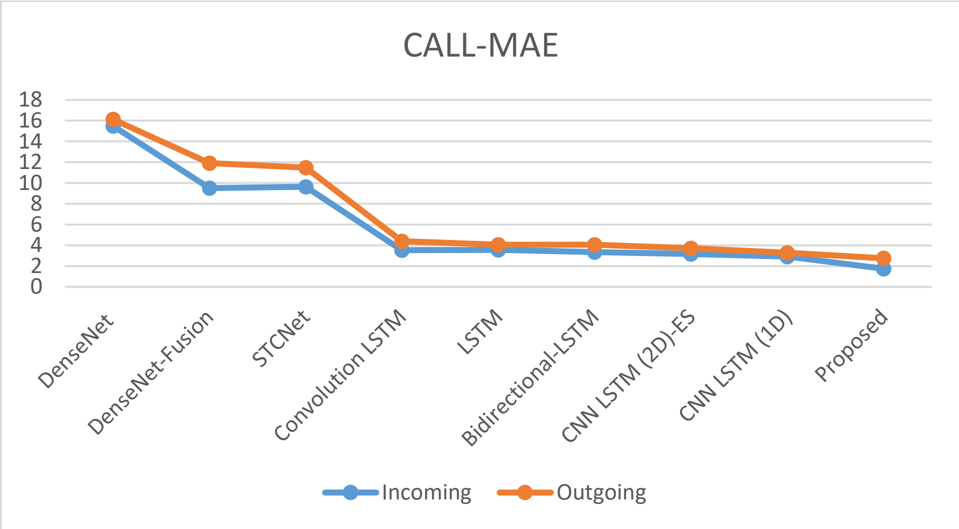

Figure 2 Incoming and outgoing call comparison considering MAE metrics

Figure 3 shows the RMSE comparison for incoming and outgoing calls, it shows that model like denseNEt, Densenet-Fusion, STCNEt observes high RMSE in both scenarios i.e. incoming and outgoing. LSTM model tends to perform well especially the existing CNN-LSTM model 10.69 and 9.27 for incoming and outgoing whereas MLHN observes 8.95 and 8.65 respectively.

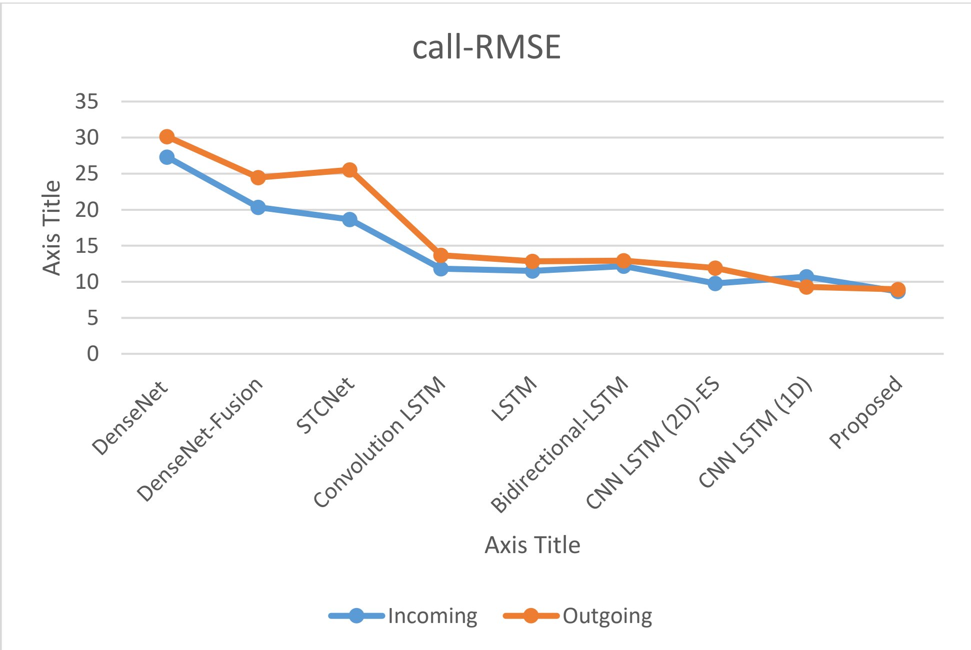

Figure 3 RMSE comparison

Figure 4 shows the R squared metrics comparison for incoming and outgoing calls, model like DenseNet, DenseNet-Fusion and STCnet observes less R-squared error, out of other LSTM based model, CNN LSTM (2D) observes 89.71 and 93.44 for incoming and outgoing call whereas proposed model observes 94.62 and 95.25 respectively.

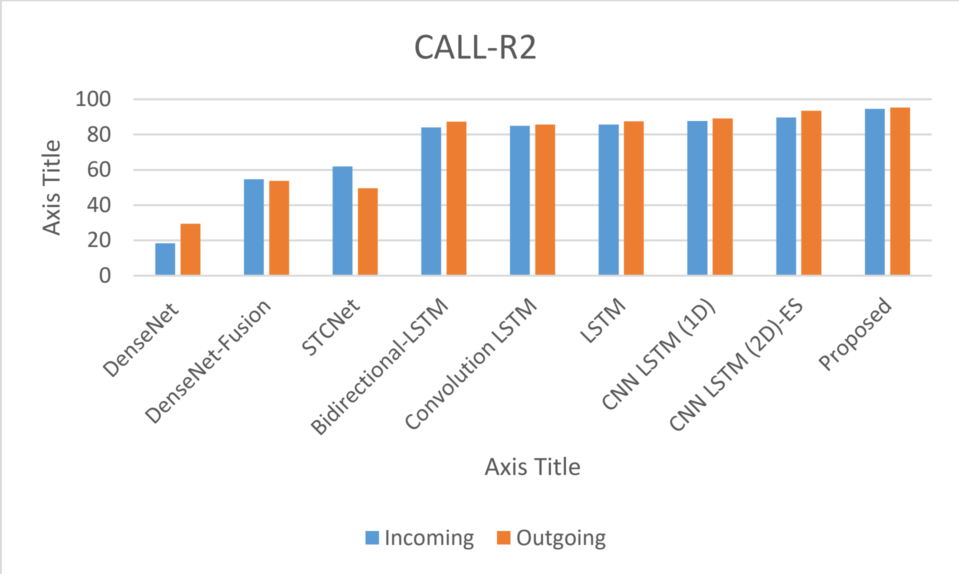

Figure 4 R-squared Call

4.4 Receiving and sending SMS

SMS plays an important part in communication; figure 4 presents the MAE comparison for receiving and sending SMS. It is observed that DenseNet, DenseNet-Fusion, and STCNet observe higher MAE, and another model from LSTM observes optimal results. LSTM model like CNN LSTM (1D) observes MAE of 6.40 and 6.3 whereas MLHN observes 5.8 and 5.15 respectively.

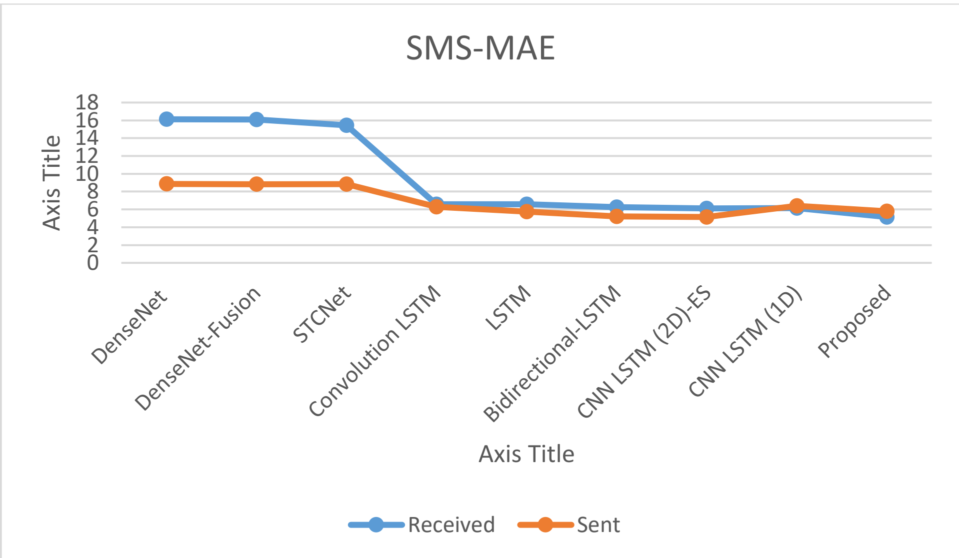

Figure 5 MAE comparison of SMS service

Figure 6 presents an RMSE comparison for receiving and sending SMS considering RMSE metrics, DenseNet and DenseNet-Fusion observe higher RMSE, the LSTM model performs, and CNN-LSTM(2D) observes RMSE values of 17.88 and 12.72 for received and sent SMS respectively. In comparison with another model, MLHN observes an RMSE value of 14.653 and 8.56 respectively.

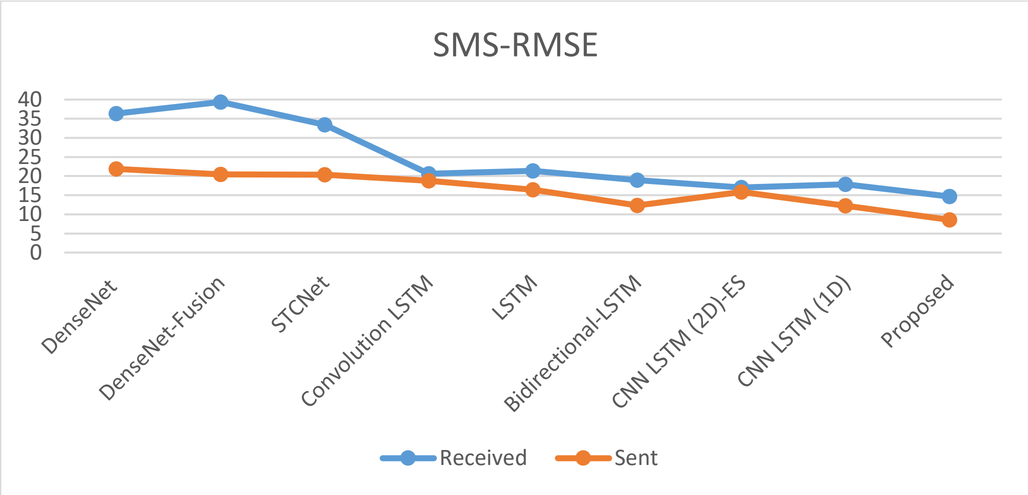

Figure 6 RMSE comparison of SMS service

Figure 7 shows the comparison of various models over SMS service considering R-squared metrics; furthermore, CNN LSTM (2D) observes better than another model with R-squared values of 89.12 and 60.32 respectively for Received SMS and sent SMS whereas MLHN observes 94.25 and 75.56 respectively for same.

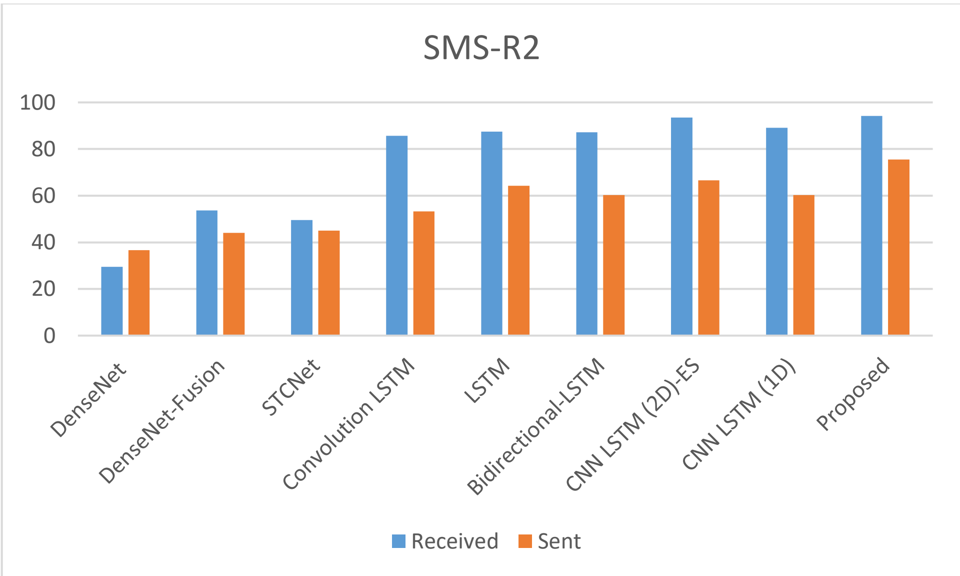

Figure 7 R2 Comparison on SMS services

4.5 Internet Traffic

Internet traffic is considered one of the major concern these days, from the below table we can conclude that Dense Net shows a value for MAE as 71.1731, whereas LSTM and Bidirectional-

LSTM shows a value of 55.445 whereas the existing system shows a value of 61.7307 wherein the proposed model depicts a value of 58.59.

Table 2 error metrics comparison on Internet service

Methodology	MAE	RMSE	R2
DenseNet	71.1731	180.2818	83.38
DenseNet-Fusion	80.2907	220.0689	75.24
STCNet	100.6001	220.2162	75.21
Convolution LSTM	93.7289	212.9368	77.1
LSTM	69.3479	198.4925	80.1
Bidirectional-LSTM	55.4445	154.2303	87.99
CNN LSTM (2D)-ES	61.7307	135.5715	86.95
CNN LSTM (1D)	64.9617	160.7744	90.72
PS	58.59	127.653	94.56

4.6 Comparative Analysis

An existing model like CNN LSTM aims to perform optimally and tends to improvise over the other single architecture like STCNet and DenseNet. However, the Existing model fails to keep consistency over the various metrics; furthermore, the proposed model MLHN keeps consistent improvisation over the various services. Table 3 presents the improvisation over the existing model considering different metrics; moreover, considering Incoming call traffic, MLHN observes improvisation of 39.86, 19.08, and 1.93 (in percentage) in comparison with the existing model. Similarly, considering outgoing call traffic MLHN observes improvisation of 16.60%, 3.45%, and 5.47% for MAE, RMSE, and R2 respectively.

Table 3 Improvisation of the existing model

Call Service	MAE (in percentage)	RMSE(in percentage)	R2(in percentage)
Incoming	39.86	19.08	1.93
Outgoing	16.60	3.45	5.47

Table 4 presents the Receive and Sent SMS traffic; Considering the SMS sending service, MLHN observes improvisation of 16.42%, 30.24%, and 0.86% for MAE, RMSE, and R2 respectively. Furthermore, considering SMS sending service, MLHN observes improvisation of 16.60%, 3.45%, and 13.47% for MAE, RMSE, and R2 respectively.

Table 4 Improvisation of the existing model

SMS Service	MAE(in percentage)	RMSE(in percentage)	R2(in percentage)
Receive	16.42	30.24	0.86

Sent	16.60	3.45	13.47

Table 5 presents MLHN improvisation over the existing model; considering MAE, RMSE, and R2 it observes improvisation of 9.80 %, 20.599%, and 8.75% respectively.

Table 5 Improvisation of the existing model

Service	MAE(in percentage)	RMSE(in percentage)	R2(in percentage)
Internet	9.80	20.599	8.75

5 CONCLUSION

Cellular Traffic prediction and analysis over the Big Data domain has been trending research with the demand for efficient resource distribution and network management, existing protocol same has not been cost-effective due to a higher error rate. This research work introduced MLHN architecture for traffic prediction; MLHN comprises multiple layers, which is utilized for multiple input. MLHN aims at extracting the custom feature; also an efficient parameter tuning and learning algorithm are designed that can reduce the error through an automated learning approach. MLHN has been evaluated considering the “Big Data Challenge” dataset which comprises three services i.e. SMS, CALL, and INTERNET; further analysis is carried out on an incoming call and outgoing call, Received SMS, and sent SMS considering the various state-of-art technique. MLHN performance is proved through performing the comparative analysis and it suggests that MLHN observes marginal improvisation over the other model.

Moreover, every year telecommunication domain observes a major revolution like 5G, 6G, and many more to come this makes more traffic and efficient management, which makes it more complicated. Hence, future work should focus on complexity reduction.

REFERENCE

[1] H. Ahmed and M. A. Ismail, "A Structured Approach towards Big Data Identification," in IEEE Transactions on Big Data, doi: 10.1109/TBDATA.2021.3139069.

[2] S. Song, F. Gao, R. Huang and C. Wang, "Data Dependencies over Big Data: A Family Tree," in IEEE Transactions on Knowledge and Data Engineering, doi: 10.1109/TKDE.2020.3046443.

[3] A. Azari, F. Salehi, P. Papapetrou and C. Cavdar, "Energy and Resource Efficiency by User Traffic Prediction and Classification in Cellular Networks," in IEEE Transactions on Green Communications and Networking, vol. 6, no. 2, pp. 1082-1095, June 2022, doi: 10.1109/TGCN.2021.3126286.

[4] Siyun Feng, Jiashuang Huang, Qinqin Shen, Quan Shi, Zhenquan Shi, "A Hybrid Model Integrating Local and Global Spatial Correlation for Traffic Prediction", IEEE Access, vol.10, pp.2170-2181, 2022.

[5] Z. Rao, Y. Xu, S. Pan, J. Guo, Y. Yan and Z. Wang, "Cellular Traffic Prediction: A Deep Learning Method Considering Dynamic Non-Local Spatial Correlation, Self-Attention, and Correlation of Spatio-Temporal Feature Fusion," in IEEE Transactions on Network and Service Management, 2022, doi: 10.1109/TNSM.2022.3187251.
[6] A. Rago, G. Piro, G. Boggia, and P. Dini, “Multi-task learning at the mobile edge: An effective way to combine traffic classification and prediction,” IEEE Transactions on Vehicular Technology, vol. 69, no. 9, pp. 10 362–10 374, 2020.
[7] G. Aceto, G. Bovenzi, D. Ciuonzo, A. Montieri, V. Persico, and A. Pescap´e, “Characterization and prediction of mobile-app traffic using markov modeling,” IEEE Transactions on Network and Service Management, vol. 18, no. 1, pp. 907–925, 2021.
[8] L. Yu et al., "STEP: A Spatio-Temporal Fine-Granular User Traffic Prediction System for Cellular Networks," in IEEE Transactions on Mobile Computing, vol. 20, no. 12, pp. 3453-3466, 1 Dec. 2021, doi: 10.1109/TMC.2020.3001225.
[9] Chen, Aaron, Jeffrey Law, and Michal Aibin. "A Survey on Traffic Prediction Techniques Using Artificial Intelligence for Communication Networks." Telecom. Vol. 2. No. 4. MDPI, 2021.
[10] Lohrasbinasab, Iraj, et al. "From statistical-to machine learning-based network traffic prediction." Transactions on Emerging Telecommunications Technologies 33.4 (2022): e4394.
[11] C. Zhang, H. Zhang, D. Yuan, and M. Zhang, “Citywide cellular traffic prediction based on densely connected convolutional neural networks,” IEEE Commun. Lett., vol. 22, no. 8, pp. 1656–1659, Aug. 2018.
[12] C. Zhang and P. Patras, “Long-term mobile traffic forecasting using deep spatio-temporal neural networks,” in Proc. 18th ACM Int. Symp. Mobile Ad Hoc Netw. Comput., 2018, pp. 231–240.
[13] Q. He, A. Moayyedi, G. Dán, G. Koudouridis, and P. Tengkvist, “A meta-learning scheme for adaptive short-term network traffic prediction,” IEEE J. Sel. Areas Commun., vol. 38, no. 10, pp. 2271–2283, Oct. 2020.
[14] 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,” IEEE Access, vol. 8, pp. 172387–172397, 2020.
[15] N. Zhao, Z. Ye, Y. Pei, Y.-C. Liang, and D. Niyato, “Spatial-temporal attention-convolution network for citywide cellular traffic prediction,”IEEE Commun. Lett., vol. 24, no. 11, pp. 2532–2536, Nov. 2020.
[16] S. Zhao et al., “Cellular network traffic prediction incorporating handover: A graph convolutional approach,” in Proc. 17th Annu. IEEE Int. Conf. Sens. Commun. Netw. (SECON), Como, Italy, 2020, pp. 1–9.
[17] C. Zhang, S. Dang, B. Shihada, and M.-S. Alouini, “Dual attentionbased federated learning for wireless traffic prediction,” in Proc. IEEE Conf. Comput. Commun., Vancouver, BC, Canada, 2021, pp. 1–10.
[18] S. P. Sone, J. J. Lehtomäki, and Z. Khan, “Wireless traffic usage forecasting using real enterprise network data: Analysis and methods,” IEEE Open J. Commun. Soc., vol. 1, pp. 777–797, 2020.

[19] W. Wang, C. Zhou, H. He, W. Wu, W. Zhuang, and X. S. Shen, “Cellular traffic load prediction with LSTM and Gaussian process regression,” in Proc. IEEE Int. Conf. Commun. (ICC), Dublin, Ireland, 2020, pp. 1–6.

[20] X. Xing, Y. Lin, H. Gao, and Y. Lu, “Wireless traffic prediction with series fluctuation pattern clustering,” in Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops), Montreal, QC, Canada, 2021, pp. 1–6.

[21] C. Zhang, H. Zhang, J. Qiao, D. Yuan, and M. Zhang, “Deep transfer learning for intelligent cellular traffic prediction based on cross-domain big data,” IEEE J. Sel. Areas Commun., vol. 37, no. 6, pp. 1389–1401, Jun. 2019.

[22] Y. Fu and X. Wang, "Traffic Prediction-Enabled Energy-Efficient Dynamic Computing Resource Allocation in CRAN Based on Deep Learning," in IEEE Open Journal of the Communications Society, vol. 3, pp. 159-175, 2022, doi: 10.1109/OJCOMS.2022.3146886.

[23] G. Barlacchi et al., “A multi-source dataset of urban life in the city of Milan and the Province of Trentino,” Sci. Data, vol. 2, no. 1, pp. 1–15, Oct. 2015.