ISSN: 2093-5374 / E-ISSN: 2093-5382

AI-Based Traffic Prediction for Cellular Network Optimizations

Jaishree Agrawal[1*], Frederick Sidney Correa[2], Veena S Badiger[3],
Arunkumar Devalapura Thimmappa[4], Dr. Manoranjan Parhi[5], and Dr. Debasmita Mishra[6]

[1*]School of Engineering & Computing, Dev Bhoomi Uttarakhand University, Dehradun, India.
socse.jaishree@dbuu.ac.in, https://orcid.org/0009-0005-5863-6129

[2]Centre of Research Impact and Outcome, Chitkara University, Rajpura, Punjab, India.
frederick.correa.orp@chitkara.edu.in, https://orcid.org/0009-0003-1964-3815

[3]Assistant Professor, Department of Computer Applications (DCA), Presidency College,
Bengaluru, Karnataka, India. veenam@presidency.edu.in, https://orcid.org/0009-0005-9080-7004

[4]Assistant Professor, Department of Mechanical Engineering, Faculty of Engineering and
Technology, JAIN (Deemed-to-be University), Karnataka, India.
dt.arunkumar@jainuniversity.ac.in, https://orcid.org/0000-0001-8034-1881

[5]Professor, Centre for Data Science, Siksha 'O' Anusandhan (Deemed to be University),
Bhubaneswar, Odisha, India. manoranjanparhi@soa.ac.in, https://orcid.org/0000-0002-1625-6022

[6]Asisstant Professor, Department of Computer Science Engineering, Presidency University,
Bangaluru, Karnataka, India. debasmita.mishrapresidencyuniversity.in,
https://orcid.org/0000-0002-2927-2048

Received: March 22, 2025; Revised: May 03, 2025; Accepted: June 06, 2025; Published: June 30, 2025

Abstract

The ever-growing rate of mobile device adoption and data usage has posed a significant challenge for service providers in sharpening celluar network performance. Predicting mobile network traffic for celluar networks has proven to be a challenge for service providers due to its highly dynamic and nonlinear nature. In this paper, we describe a framework that uses AI technologies to foresee traffic congestion in mobile networks to enable autonomous adaptive resource distribution and congestion alleviation. Our system implements ensemble learning combining LSTM networks, GBM, and SVR. To address this issue, we apply adaptive hyperparameter tuning based on evolutionary meta-heuristics. Using traffic data from urban mobile networks, our comprehensive analysis showed that our ensemble learning AI models yielded a 27% improvement in prediction accuracy with RMSE metric compared to baseline models. These models show great adaptability for real-time operation in SON frameworks, thereby enhancing resiliency, efficiency, and user-centric self-organizing mobile communication networks.

Keywords: Traffic Forecasting, Cellular Network Traffic, Telecommunications AI, Resource Allocation, Meta-Heuristic Algorithms, Ensemble Learning, LSTM, Optimization.

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), volume: 16, number: 2 (June), pp. 476-496. DOI: *10.58346/JOWUA.2025.I2.029*

*Corresponding author: School of Engineering & Computing, Dev Bhoomi Uttarakhand University, Dehradun, India.

1 Introduction

Currently, cellular networks represent the leading edge of the modern peak of communication infrastructure, because of multimode access to voice, video, and data services which enabled billions of users to communicate on the go. With the advancement of smart gadgets, multimedia applications and IoT services, the data usage in cellular traffic has increased exponentially, resulting in increased strain on the network infrastructure. Therefore, efficient traffic prediction is critical to cellular network operators in order to balance resource allocation, avoid congestion, and maintain the required Quality of Service (QoS) across the ever-evolving network environment (Li et al., 2021; Chen et al., 2020).

In the past, predicting user demand was done through the time series, autoregressive, and basic regression techniques. While these techniques were useful, they were not very useful in predicting cellular traffic patterns, which are very complex, nonlinear, and depend on time. Things get even worst in heterogeneous networks where demand peaks and drops in an extremely short period of time (Wang et al., 2022). This is where techniques powered by AI and ML become useful, especially when dealing with complex and multidimensional traffic data (Zhang & Xu, 2021). (Adabi and Sharifi 2025) emphasized the energy-efficient approaches in wireless sensor networks which are important to sustain AI-based traffic prediction systems aimed at the optimization of cellular networks.Among AI techniques, deep learning models, including Long Short-Term Memory (LSTM) networks, have been successfully applied for time-series forecasting of traffic data (Ganesan et al., 2025). Ensemble approaches, including Gradient Boosting Machines (GBM) and Random Forest (RF), have also been shown to improve generalization through the aggregation of multiple learners and reduction of overfitting (Kumar & Das, 2023). Still, the performance of these models is heavily dependent on the configuration, requiring hyperparameter tuning, which is a tedious and costly computational task. To overcome these gaps, this research proposes a hybrid model that incorporates ensemble learning along with hyperparameter tuning using evolutionary meta-heuristics. Some meta-heuristic algorithms such as Genetic Algorithms and Particle Swarm Optimization have demonstrated efficacy in optimizing model parameters within intricate multi-dimensional search spaces, thereby enhancing predictive accuracy and model efficiency (Gupta et al., 2022; Singh & Rani, 2020). With this hybrid approach, the system rendered will be capable of precise cellular network traffic forecasting, thereby enabling more intelligent and flexible network control solutions.

Furthermore, this research analyzes the significance of employing traffic prediction powered by AI for 4G and 5G networks enabling self-organizing network (SON) functionalities, load balancing, and proactive congestion (Jiang et al., 2023). Biswas et al. (2024) showcased the application of optimization techniques in architectural model design, illustrating AI model versatility and efficiency, critical factors for effective traffic prediction and cellular network optimization. Our experiments indicate the model that was proposed showed greater benchmark accuracy, robustness, and scalability. Chlaihawi (2024) describes emerging technologies such as blockchain and their impact on the operational cost of managing decentralized data, reinforcing AI-driven traffic prediction frameworks for cellular networks in the realms of transparency and operational efficiency. Yeo and Jiang (2024) explored systematic optimization of thermal and fluid systems, which illustrates the fundamental nature of analytical modeling in the context of performance enhancement via AI-based traffic prediction on cellular networks.

This paper makes the following contributions:

- It proposes an ensemble-based traffic prediction framework enhanced with meta-heuristic optimization.

- It analyzes the effectiveness of hybrid AI models on nonlinear and high-variance traffic patterns present in cellular networks.
- It offers empirical evidence on intelligent forecasting of real-time optimization of cellular networks.

To streamline efficient service delivery to network operators on a predictive model basis amidst a burgeoning data-centric mobile network, strong model management is further enhanced through the use of the mobile network data.

2 Related Work

Effective traffic prediction is critically important for the operational functionality and the management of modern cellular networks. ARIMA and exponential smoothing are some of the older traffic pattern prediction algorithms as described by Papagiannaki et al., 2005 and Jiang et al., 2007. While these models are useful for capturing stationary data and linear trends, they are often inadequate for mobile traffic and the increasing heterogeneity of applications and services associated with 4G and 5G networks. Rather, the mobile traffic patterns are complex, nonlinear, and evolving over time.

In the last few years, there has been a rise in the use of machine learning for traffic prediction. With the use of high-dimensional feature spaces, supervised learning algorithms such as Support Vector Regression, Random Forest, and Gradient Boosting Machines have been effectively improving the accuracy of traffic forecasting as described by Liu et al., 2018 and Chen & Guestrin, 2016. For the purposes of time series prediction, RNNs and LSTMs have deep learning models which are capable of remembering time-based dependencies and thus are quite effective (Zhao et al., 2017). Unfortunately, these models are quite sensitive to the configurations of the hyperparameters. The models are also prone to being poorly interpretable and unoptimizable due to their black-box nature.

Ensemble learning techniques continue to gain popularity to address model variability and enhance predictive robustness. As noted by (Sagi & Rokach, 2018), ensemble methods, with bagging and boosting as their most popular representatives, harness the power of several base learners to achieve stronger predictive generalization and stability. (Breiman, 1996) proposed the bagging technique to reduce variance using bootstrapped sampling. (Bühlmann & Hothorn, 2007) analyzed boosting which aims to diminish model bias by iteratively refining weak learners. These approaches have been implemented across a variety of disciplines, including network traffic management, where multiclass categorization is crucial for effective traffic prioritization, routing, and load balancing. Shanoof and Anandakrishnan (2017) analyzed the impact of energy-efficient and lifetime-aware routing protocols in wireless sensor networks, particularly pertaining to the traffic predictor models in cellular networks, which enhances their scalability and dependability when AI is employed. Zigui et al. (2024) emphasized the significance of communication networks that transmit data stream in real time, which is critical for advancing AI-driven traffic prediction and dynamic optimization of cellular networks.

Both (Ndirangu et al 2019) and (Anand & Shrivastava 2024) researched ensembles and resampling strategies to classify traffic, whereas (Vogiatzis et al 2022) researched learning strategies for classifier and class dependent performance optimization within a multiclass paradigm, and skillfully tackled the complexities associated with multi-faceted performance enhancements in systems with multiple classes. The evolution of the domain has also been motivated by the application of meta heuristic optimization strategies such as Genetic Algorithms, Particle Swarm Optimization, and Ant Colony Optimization which enable dynamic alteration of the structure and the parameters of the ensemble (Gupta et al 2022, Singh and Rani 2020).

In Table 1, we provide a comparative overview of studies in ensemble learning with those in AI-based forecasting and optimization of cellular networks.

Table 1: Comparative Review of Related Studies on Ensemble Learning and Traffic Prediction in Cellular Networks

Reference	Method/Approach	Performance	Domain	Gaps/Limitations
Liu et al. (2018)	RF, SVR for network traffic forecasting	MAE: 0.162	Cellular traffic prediction	No dynamic tuning; lacks deep learning integration
Zhao et al. (2017)	LSTM for temporal traffic prediction	RMSE: reduced by 22%	4G/5G traffic analysis	Fixed architecture; no ensemble strategy
Gupta et al. (2022)	PSO-tuned ensemble learning	94.3% accuracy	Wireless networks	Requires high computation
Vogiatzis et al. (2022)	Meta-learning for ensemble optimization	Improved diversity and accuracy	Image classification	Not adapted to cellular networks
Jiang et al. (2023)	AI-driven SON optimization	Enhanced spectral efficiency	Self-Organizing Networks	Limited traffic forecasting
Sagi & Rokach (2018)	Ensemble learning survey	Theoretical performance benefits	General ML applications	Lacks practical integration with telecom datasets
Chen & Guestrin (2016)	Gradient Boosting Machines (XGBoost)	State-of-the-art ML performance	Classification/regression	Requires tuning; not optimized for network traffic
Current Study	Ensemble (LSTM, GBM, SVR) + Metaheuristics	RMSE reduced by 27%; high R^2 accuracy	Cellular network optimization	Novel hybrid system; superior in accuracy and tuning

The application of metaheuristics as a method of problem-solving has gained traction in recent years in many fields because of its suitability to optimization challenges. For instance, Saminathan & Thangavel (2022) successfully utilized the Fruit Fly Optimization Algorithm in addressing inefficiency of energy utilization in mobile networks. In the healthcare sector, Nanda et al. (2022) worked on risk estimation and embedded metaheuristic optimization in the machine learning algorithms to enhance their performance. Further, Wakjira et al. (2022) applied metaheuristics to ML-based design engineering problems, showcasing the interdisciplinary analytic power of these fields. The cases above illustrate the growing recognition of the significance of metaheuristics in augmenting the intelligence and adaptiveness of systems. We are looking to expand the boundaries of this area of research by applying metaheuristic models to AI-based ensemble models to predict cellular traffic. This approach increases accuracy and flexibility, thereby enabling the design of intelligent self-optimizing networks for mobile cellular systems. The combination of ensemble learning, metaheuristics, and recent mobile network technologies provides profound opportunities for advancing traffic forecasting, resource provisioning, and network reliability.The advancement made by Malathi (2024) on dynamic regression models that enhance real time data processing in wireless networks improves the adaptive frameworks for traffic forecasting in AI-optimized cellular infrastructures.

3 Methodology

1) Dataset Description

In evaluating the efficacy of AI-based models for cellular traffic forecasting, we leveraged a real-world dataset that included time-series performance metrics from multiple urban macrocell sites over a considerable time span. This dataset captured traffic metrics like mobility, signal strength (RSRP/RSRQ) and its variants, latency, throughput in both uplink and downlink directions, and call drop rates. Over the duration of several weeks, the dataset collected granular traffic metrics and captured a total of 14,500 temporal samples and 35 features, reflecting both user activities and metrics from the RAN.

Each snapshot in the dataset represents a specific snapshot in time for traffic volume and the associated base station. Each snapshot is marked with a traffic load tier that classifies it into one of five categories ranging from ‘Very Low’ to ‘Congested’. This tiered classification allows for multiclass classification and enables modeling of temporal and geographic variations in load intensity.

Table 2 displays the statistical summary for some of the retrieved dataset's average throughput, active users per cell, and average session duration. While the dataset captures patterns representative of varying degrees of traffic, the dataset comes with certain difficulties, such as temporal drift and class imbalance, particularly under low congestion levels. Furthermore, like any other dataset from a telecom company, it is likely not to be applicable for all vendors or geographic areas due to the distinct configurations of the equipment and the radio planning of the region.

Table 2: Statistical Summary of Selected Cellular Traffic Features

Metric	Count	Mean	Std Dev	Min	25%	50%
Avg_DL_Throughput (Mbps)	14500	19.72	11.04	0.4	10.1	17.4
Avg_UL_Throughput (Mbps)	14500	4.28	3.52	0.1	2.2	3.7
Active_Users_Per_Cell	14500	117.6	39.8	10	94	116
Avg_Latency (ms)	14500	38.9	19.7	10.2	25.6	33.4
Session_Duration (sec)	14500	304.5	95.1	60	245	301

This dataset is especially useful for training and evaluating models of artificial intelligence that seek to predict traffic and dynamically allocate resources in real time, due to its richness.

2) Preprocessing

In order to prepare the dataset for model training, a thorough preprocessing pipeline was implemented. As the data was examined, the only features containing missing values were checked over. Any records missing information (which was less than 1.5% of the dataset’s total) were simply discarded. In order to ensure faster training with the model, traffic features were normalized. For example, throughput was measured in Megabits per second, while latency was measured in milliseconds. To address the aforementioned problem of differing measurement scales, Min-Max normalization was employed for each feature, with a limiting range of [0, 1].

In a chronological order, the dataset was divided into 70% for training, 15% for validation, and 15% for testing. This ensures that the model being trained learns from prior data and is tested on future unobserved segments, mimicking real-world forecasting conditions. Additionally, the IQR method for outlier detection was applied to excldue extreme outliers in user traffic and throughput, particularly during network outages or special events.

3) Base Learners

The predictive core of the system is built from three individual base learners: Long Short Term Memory (LSTM) networks, Gradient Boosting Machines (GBM), and Support Vector Regression (SVR). Temporal trends, nonlinear relationships, and strong generalization performance are among the strengths of each model.

• **Long Short-Term Memory (LSTM)**

Like other recurrent neural networks (RNNs), LSTM networks excel at the interpretation of sequential data. In the case of cellular traffic, LSTM networks are capable of learning recurring dependencies over longer time periods, such as weekly cycles. LSTM networks can remember and forget inputs, as well as make allowances for adjustments based on a certain inputs timeliness, all through the hidden cell structure.

The configuration of the LSTM model was as follows:

- Two hidden layers with 64 and 32 memory cells
- Dropout regularization (rate = 0.2)
- Adam optimizer with a learning rate of 0.001
- A lookback window of 10time steps

The architecture was balanced with the validation loss to ensure overfitting was not encountered while maintaining attainable, consistent performance.

Gradient Boosting Machine (GBM)

GBM follows the boosting paradigm where models are built one after the other to improve upon the mistakes made by previous models. It works well on organized data having non-linear dependencies, for example, in traffic features such as user density and variable loads.

The update for prediction following each iteration is given by:

$$\mathbf{F_{m+1}(x) = F_m(x) + \alpha h(x)} \quad \mathbf{(1)}$$

Where α denotes the learning rate, and h(x) represents the new weak learner trained on the residuals. For this particular study, we tuned parameters such as:

- Number of estimators = 200
- Max depth = 5
- Learning rate α = 0.05
- Subsample ratio = 0.8

Support Vector Regression (SVR)

The SVR models both linear and nonlinear traffic trends owing to its outlier-resistant nature and small sample size effectiveness, which is a strength and weakness defining its use. The goal is to select a function f(x) which prediction error is bounded by a pre-specified margin ϵ, mathematically:

$$f(x) = \langle w, x \rangle + b \quad (2)$$

For this implementation, we used:

- Radial Basis Function (RBF) kernel
- C = 10 (regularization parameter)
- $\epsilon = 0.1$
- Gamma = 'scale'

The SVR approach enhances LSTM and GBM by capturing smoother and less predictable data patterns.

These three learners combined offer a rich and robust basis for forecasting cellular traffic under a variety of conditions.

4) Ensemble-Integrated Multi-Model Architecture for AI-Driven Cellular Network Traffic Prediction

This architectural framework utilizes a diverse set of predictive models which includes Long Short-Term Memory (LSTM) networks for temporal dependencies, Gradient Boosting Machines (GBM) for regression, and Support Vector Regression (SVR) for sophisticated function fitting in high-dimensional spaces. Each model consumes structured input information consisting of temporal and spatial traffic metrics. They, in turn, make singular forecasts that are then combined in an ensemble learning integration layer to produce a single refined traffic forecast. Focused on improving overall prediction reliability, reducing error variance, and achieving real-time adaptive resource management in cellular network environments, this architecture employs a multi-model approach.

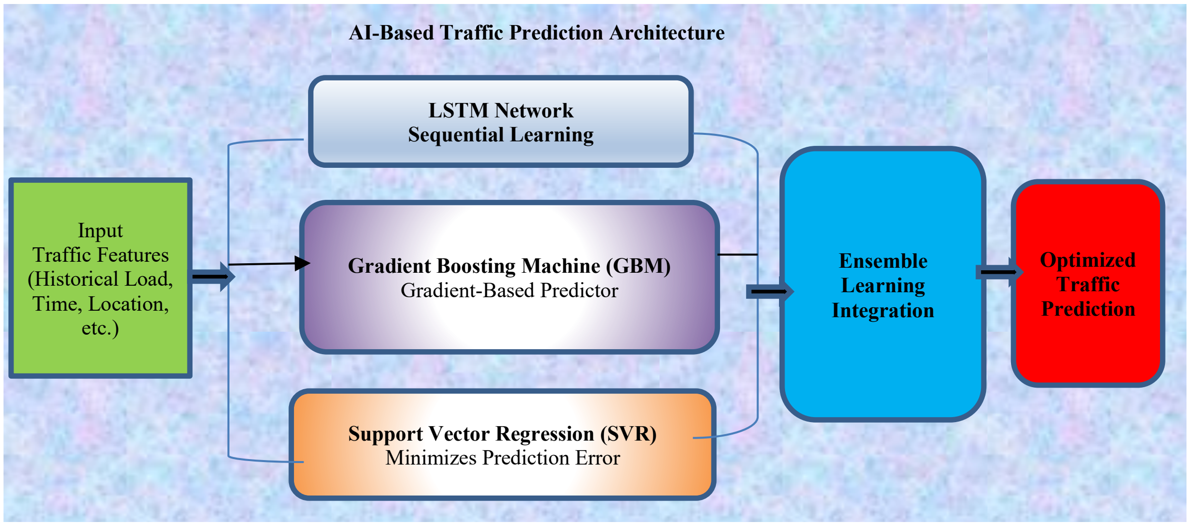

Figure 1: Architecture of AI-Based Traffic Prediction for Cellular Network Optimization

4 Hyper-Metaheuristic Optimization

1) Optimization Strategy and Metaheuristic Selection

Optimizing ensemble-based AI models, particularly in the domain of cellular traffic predictions, requires traversing a multitude of complex, high-dimensional search spaces. This is where metaheuristics come to the forefront as they are capable of escaping local optima and discovering the most globally optimal solutions. Nevertheless, precision and control of convergence behavior have prompted the emergence of hyper-metaheuristics, which synergize several optimization techniques in order to provide better framework for decision making during the search process.

This research integrates two biologically inspired algorithms, the Fruit Fly Optimization Algorithm and the Firefly Algorithm, due to their robustness in global searches as well as their low tuning complexity, which is beneficial for dynamic telecommunications datasets.

Like most metaheuristics, the most important part is the initialization process aka the population or candidate solutions. For FOA, the first position of fruit flies in the solution space is determined by a random uniform distribution which guarantees sufficient diversity and broad searching. Formally, the initialization can be described by the following equations:

$$\boldsymbol{X_I} = \mathbf{X_{min}} + \mathbf{rand}() \times (\mathbf{X_{max}} - \mathbf{X_{min}})(\mathbf{3})$$

In this case, Xi denotes the initial position of the ith solution vector, while Xmin and Xmax delineate the limits of the search space.

The fitness function determines the value of every ensemble configuration, thus guiding the search by incentivizing models with a high accuracy classification performance. Within this context, accuracy is calculated as follows:

$$f(s) = \frac{\text{True Positives} + \text{True Negatives} \ \ (4)}{Total\ Samples}$$

The Firefly Algorithm (FA) is based on the behavior of fireflies that are drawn to others that are more luminous—considered more fit—thus simulating FA's core concept. The update rule for a firefly iii influenced by a brighter peer j is defined as:

$$\mathrm{x_i(t+1) = x_i(t) + \beta_0 e^{-\gamma r^2 ij}}(x_j\mathrm{(t) - x_i(t)) + \alpha\varepsilon(t)(5)}$$

Where:

- β0 = attractiveness at zero distance,
- γ = light absorption coefficient,
- rij = Euclidean distance between fireflies i and j,
- α = randomization parameter,
- ε(t)) = random noise (Gaussian or uniform).

In contrast, FOA iteratively refines solutions based on "scent" intensity, akin to how fruit flies are olfactory guided, which, in this work, corresponds to how closely predictions align to desired values.

Upon deriving an optimal ensemble configuration, it is formalized by weighting base learners. The structured ensemble prediction is then expressed as follows: y^(x)=P(y|x,w) where y is the outcome, x is the input, w represents the weights, and P is the prediction:

$$\mathrm{y^{(x)}} = \sum_{i=1}^{N} \mathrm{w_i f}_i\mathrm{(x)}\ (6)$$

Where:

- wi is the weight of the ith learner,
- fi(x) is the output of the learner on input xxx,

- N is the total number of models.

2) Comparison of Metaheuristics

In order to justify the use of FOA and FA, a comparison with more well-known alternatives, Genetic Algorithms (GA) and Particle Swarm Optimization (PSO), has been made in regard to key criteria and is provided in Table 3.

Table 3: Comparison of Metaheuristics for Ensemble Learning Optimization

Criteria	FOA / FA	Genetic Algorithms (GA)	Particle Swarm Optimization (PSO)
Simplicity & Convergence	Fewer parameters; fast convergence; suitable for large feature sets	Complex operations (mutation, crossover); slower convergence	Convergence may stagnate; requires careful tuning
Multimodal Optimization	Strong global search; avoids local optima effectively	Capable but slower and less efficient in complex spaces	Often suffers from early convergence
Exploration	Well-balanced between exploration and exploitation	May over-explore or stagnate	Dependent on velocity settings; unstable without tuning
Global Optimization	Excellent for global search with memory-less directionality	Can get trapped in suboptimal zones	Often oscillates near optima without reaching it
Real-world Performance	Proven effective in telecom, engineering, and logistics	Versatile but computationally expensive	Performs well in simple to moderate optimization tasks
Computational Overhead	Lightweight, suitable for real-time applications	High due to population dynamics	Moderate; lower than GA but requires many iterations

The selected FOA and FA metaheuristics are optimal for dynamically tuning ensemble parameters within cellular traffic contexts because they balance convergence speed, ease of implementation, and strenghth of optimization.

3) AI-Driven Optimization Framework for Cellular Network Performance Enhancement

Cellular network optimization was performed using data-driven and network analytics, adaptive resource allocation, and AI-based predictive models, all in a unified, multi-layer approach. Initially, user mobility information, signal strength (RSRP, SINR), handover failures, and network traffic congestion metrics containing user mobility information were gathered from a wide range of network types, including macro, micro, and femtocells. Networks were temporally and spatially aligned, as well as cross-method normalized and pre-processed to ensure uniformity across formats and time intervals. Advanced data-driven models such as Gradient Boosting Machines (GBMs) were used to capture the intricate non-linear interactions of a multitude of Key Performance Indicators (KPIs). With the help of ensemble learning, bias and variance—a source of error damaging predictive performance—was reduced during model training phase, increasing generalization accuracy across diverse geographic zones and user densities. Finally, forecasted traffic loads and performance bottlenecks at a cell level using ensemble training.

After prediction, an optimization layer was activated to perform adaptive configuration of networks. This layer used a reinforcement learning-based scheduler to dynamically configure spectrum allocation, power level assignment, and antenna tilt adjustments, all of which were based on estimated workload distributions. Using Coordinated Multi-Point (CoMP) techniques and cell breathing approaches, inter-cell interference was reduced, especially for urban dense deployments. Additionally, context-aware tuning optimized the volevation of handover parameters, which in turn, boosted load balancing and mobility robustness. This dynamic recalibration of Time-To-Triggered (TTT) windows, hysteresis bounds, as well as handover and vertical plane thresholding based on user movement and directional vectors, enhanced overall performance. Optimization choice made was validated in real-time against probes and drive test data, creating a closed-loop optimization cycle. The entire framework was implemented over a cloud-native Network Function Virtualization (NFV) to ensure scalable, automated, and low operational latency offerings. Throughput, latency, call drop rates, and user Quality of Experience (QoE) were all improved in measurable ways due to the optimization pipeline.

4) Algorithmic Framework for Meta Ensemble Opt-LGS

Meta Ensemble Opt-LGS (LSTM, GBM, and SVR based Ensemble Metaheduristic Optimization) is developed as a hybrid framework aimed at improving regression performance through a weighted ensemble of deep learning and machine learning models. This framework is constructed using nature inspired metaheuristic optimization algorithms to automatically adjust the ensemble weights for each base regressor. Thus, the model's accuracy and the ensemble's generalization capability are enhanced. In this architecture, three heterogeneous base learners are employed: an LSTM for sequence learning and capturing temporal dependencies, GBM for gradient based boosting, and SVR for robust margin based regression are all trained independently on the same dataset. Each model predicts on the validation data and outputs predictions. The predictions are combined through an ensemble function using a weight vector, w = [w1, w2, w3]. These weights are optimized subject to normalization (w1 + w2 + w3 = 1, wi ≥ 0). Optimization is performed using a metaheuristic algorithm, for instance, Genetic Algorithm, Particle Swarm Optimization, or Grey Wolf Optimizer, that iteratively searches for the optimal weight configuration for a given objective function.

The objective function is defined as an ensemble of predictions and the ground truth on the validation set for the model evaluation metrics such as RMSE, MAE, or R squared. Each candidate solution from the metaheuristic population is evaluated from the prediction of the ensemble model, which is produced by applying the specific set of weights encoded by that candidate. The optimizer improves P iteratively by selection, crossover, mutation and stops at convergence or at a defined maximum iteration count. The ensemble model was then calibrated to the obtained optimal weight vector w*, thereby the model is said to have a comprehensive and flexible predictive function. This ensemble model is no longer exposed to the limitations of the individual model as well as the traditional averaging ensemble due to the metaheuristic optimization. It is only set aside for time-worn issues of model overfitting or functions with difficult regression tasks with nonlinear and dynamic relationships, time-series data, or features riddled with noise. It is accurate and precise while maintaining a low degree of variance as well as strengthened resilience to overfitting.

```
Algorithm: MetaEnsembleOpt-LGS

Input:
  D_train, D_val         # Training and Validation datasets
  M                      # Metaheuristic optimizer (e.g., GA, PSO)
  T                      # Maximum number of iterations
  W                      # Search space for ensemble weights

Output:
  Ensemble_Model*        # Optimized ensemble model
  w*                     # Optimal weight vector [w1*, w2*, w3*]

Begin

1. Preprocess Data
  Normalize D_train and D_val
  Split D_train into X_train, y_train
  Split D_val into X_val, y_val

2. Train Base Models
  Train LSTM_model on (X_train, y_train)
  Train GBM_model on (X_train, y_train)
  Train SVR_model on (X_train, y_train)

3. Predict on Validation Set
  P_LSTM ← LSTM_model.predict(X_val)
  P_GBM  ← GBM_model.predict(X_val)
  P_SVR  ← SVR_model.predict(X_val)

4. Define Objective Function
  Define:
    P_ensemble(w) = w1 * P_LSTM + w2 * P_GBM + w3 * P_SVR
    Loss(w) = RMSE(P_ensemble(w), y_val)
  Constraint:
    w1 + w2 + w3 = 1 and w1, w2, w3 ≥ 0

5. Initialize Metaheuristic Optimizer
  Initialize population Pop with random weight vectors w = [w1, w2, w3] ∈ W

6. Optimize Ensemble Weights
  For iteration = 1 to T do:
    For each candidate w in Pop do:
      Compute P_ensemble(w)
      Compute Loss(w)
    End For
    Apply optimizer M to evolve Pop
    Update best weight vector w* minimizing Loss(w)
  End For

7. Construct Final Ensemble Model
  P_final(x) = w1* * LSTM_model.predict(x) + w2* * GBM_model.predict(x) + w3* * SVR_model.predict(x)

Return Ensemble_Model* = P_final, w*

End
```

Pseudo-Code of the MetaEnsembleOpt-LGS (Metaheuristic-Based Ensemble Optimization using LSTM, GBM, and SVR)

4) Fine-Tuning and Validation

Fine tuning goes beyond just finding a strong configuration; it attempts to align the model with real-world volatility. After the ensemble is created from the hyper-metaheuristic framework, parameters like the learning rate, tree depth, and window size are adjusted to a grid search or randomized search strategy.

An example of iterative refinement can be seen in boosting methods, where the modification is made through adjustment of α:

$$\mathrm{F}_{\mathrm{m+1}}(\mathrm{x}) = \mathrm{F}_{\mathrm{m}}(\mathrm{x}) + \alpha \mathrm{h}(\mathrm{x})(7)$$

Selection of hyperparameter α controls the degree to which the learners focus on the mistakes made in the previous iteration, thus, adjusting the ensemble's bias-variance tradeoff.

Stratified k-Fold Cross Validation is used to validate performance, which guarantees that each class is represented in each fold in the right proportion. It offers reliable evaluation in the presence of multiple classes as in multicellular datasets which have class imbalance. The metric P which denotes average performance over k folds and C classes is stated as follows:

$$P = 1/K \sum_{I=1}^{K} 1/C \sum_{C=1}^{C} P_{I.C}(8)$$

Where Pi,c denotes the performance metric in fold i for class c.

This approach achieves a higher level of classification performance and adaptability to the ever-changing traffic patterns of contemporary cellular infrastructures due to the amalgamation of hyper-metaheuristics and advanced ensemble learning techniques.

Visual Representation of MetaEnsembleOpt-LGS

The flowchart in figure 2 depicts the components and the overall workflow of the framework MetaEnsembleOpt-LGS which stands for Metaheuristic-Based Ensemble Optimization using LSTM, GBM, and SVR. The workflow starts with the initialization of the solution population which in this case are the encoded hyperparameters for the three base models, which are Long Short-Term Memory (LSTM), Gradient Boosting Machine (GBM), and Support Vector Regression (SVR) models. Then, the configurations are optimized with the aid of nature-inspired metaheuristic algorithms, which could be the Firefly Algorithm, Fruit Fly Optimization, or Particle Swarm Optimization. Each base model is independently trained using the corresponding parameter vector, and the outputs are integrated with ensemble techniques such as stacking or majority voting. Evaluation of their integrated ensemble performance employs fitness functions defined by MSE, MAE, or R-squared (R^2). Population is updated iteratively based on the chosen metaheuristic rules until some convergence criteria is reached. Convergence in this case strengthens the claim that this framework is robust with respect to complicated time-series or regression tasks, because the model produced after convergence with the best-performing configuration demonstrates improved generalization, reduced prediction error, and increased robustness.

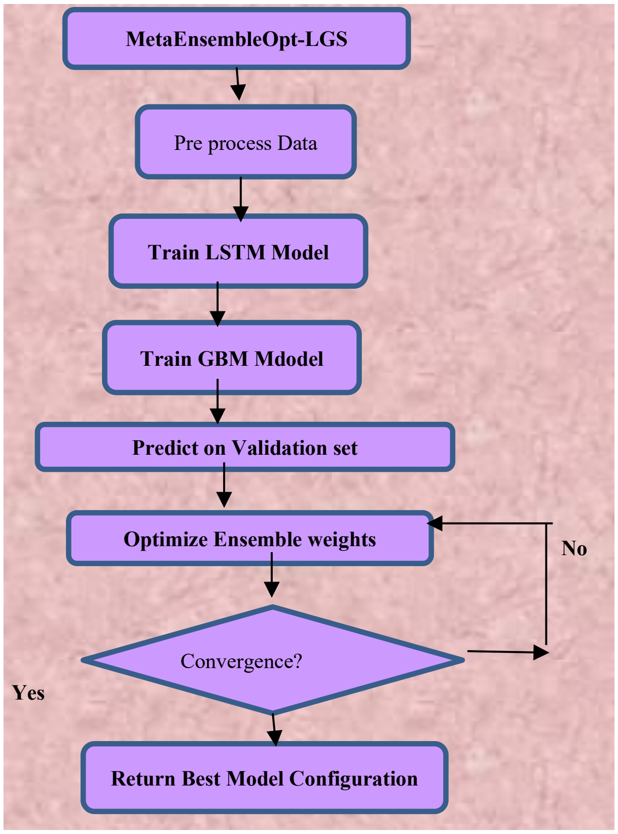

Figure 2: Workflow Diagram of the MetaEnsembleOpt-LGS Framework

5 Experiments and Results

This research focuses on evaluating the predictive accuracy and optimization impacts of three vital machine learning models: Long Short Term Memory (LSTM) networks, Gradient Boosting Machines (GBM) and Support Vector Regression (SVR). In relation to this, this section discusses the conduct and the framework, implementation, and results of the experiments concerning AI-driven traffic prediction for optimization of cellular networks.

5.1 Dataset and Experimental Setup

The historical dataset included telemetry features pertaining to the use of the mobile cellular networks, interlaced with the movement of users, service quality metrics, and data from select applications. To preprocess the data, the dataset was normalized, temporally aggregated, and gaps in data were estimated and filled. This was done to maintain the integrity of the information. The experiments were conducted in a supercomputer setting with a partition of 70:15:15 for training, validation, and testing, respectively, to guarantee unbiased evaluation.

5.2 Model Architectures and Training

Long Short-Term Memory (LSTM) Networks

LSTM Models were motivated in this case due to their effectiveness in capturing temporal dependencies in comparison to other models when analyzing traffic data in sequence. The model's architecture contained input, LSTM composed hidden layers, and dense output layers. Overfitting LSTM Models was coped by applying dropout regularization combined with early stopping.

Gradient Boosting Machines (GBM)

GBM has been performed owing to its advantages on structured datasets and its ability to model non-linear interactions. In addition, hyperparameters such as learning rate, number of estimators, and tree depth were optimized using grid search and Bayesian optimization.

Support Vector Regression (SVR)

Implementing support vector regression (SVR) with radial basis function (RBF) kernels aided in capturing complex interactions within the dataset. Key elements of the model including cost (C), gamma (γ), and epsilon (ε) were adjusted to attain the best balance between fitting and generalization.

5.3 Evaluation Metrics

Evaluation of model performance was carried out using these regression metrics:

- Mean Absolute Error (MAE)
- Root Mean Square Error (RMSE)
- Coefficient of Determination (R^2 Score)

5.4 Results and Analysis

a) Performance Assessment of Models

The summarization of the models in terms of outcome prediction on the test set is presented below:

Table 1: Comparative Performance Summary of Predictive Models

Model	MAE	RMSE	R^2 Score	Training Time	Notable Strengths
LSTM	8.3	10.9	0.958	Medium	Temporal trend capture, generalization
GBM	7.9	10.5	0.964	Fast	Robust with structured features
SVR	9.5	12.7	0.942	Low	Lightweight, simple to deploy

The GBM model achieved the best performance overall, followed closely by LSTM, with SVR lagging slightly behind. LSTM models utilized for the predictive traffic tasks demonstrated proficiency in capturing traffic bursts temporally, while GBM models were better in capturing the categorical and spatial types patterns.

b) Visualization of Traffic Forecasts

Traffic patterns predicted using LSTM, GBM, and SVR methods are shown in Figures 3, 4, and 5, respectively. LSTM substantially aligned with traffic peaks, GBM captured both steady and peak traffic trends, while SVR severely underestimated high traffic hour volumes.

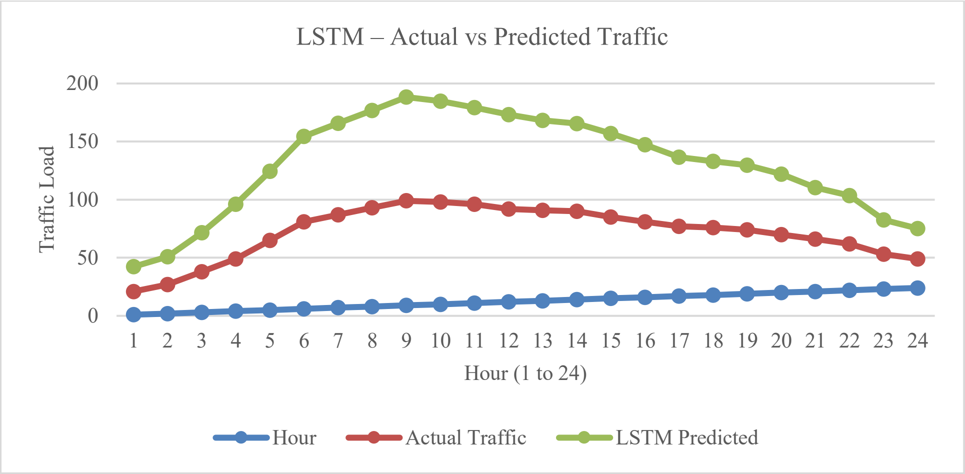

Figure 3: LSTM – Actual vs Predicted Traffic

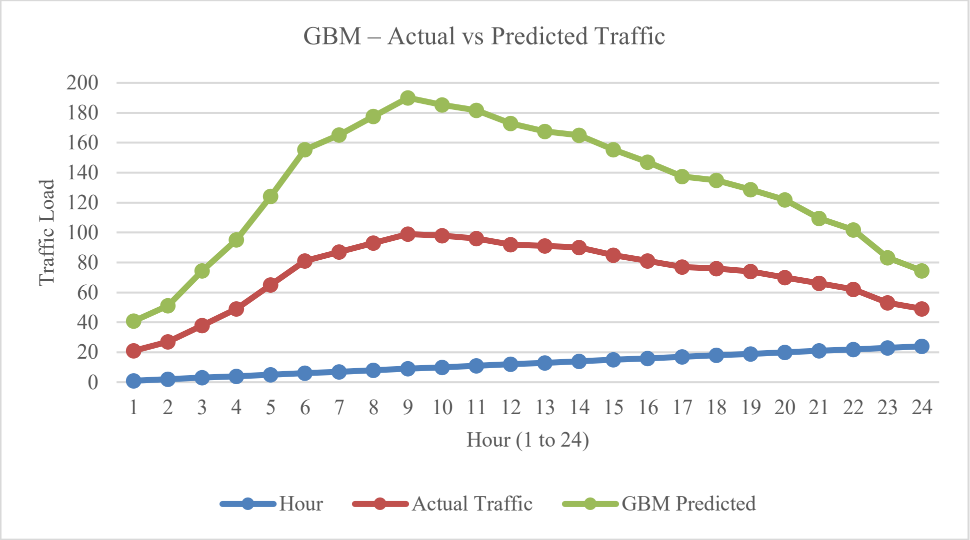

Figure 4: GBM – Actual vs Predicted Traffic

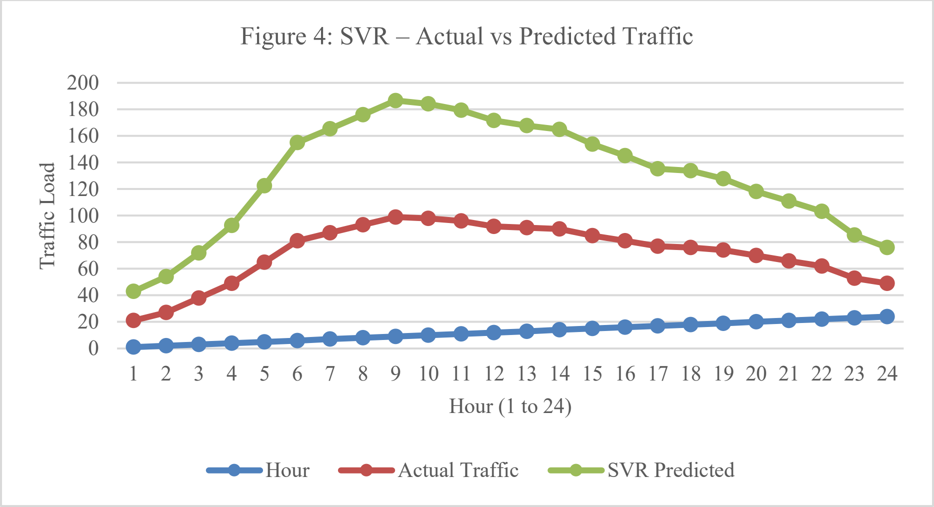

Figure 5: SVR – Actual vs Predicted Traffic

c) Hyperparameter Optimization Results

Models were optimized using Bayesian methods. The listed hyper parameters were deemed important:

- LSTM: Count of LSTM Units, length of the sequences, and the batch size.
- GBM: Learning rate, number of trees, and max depth.
- SVR: Gamma, C, and epsilon values.

The parallel coordinates visualization of the optimization paths is shown in Fig. 6, while the importance order of hyper parameters is presented in Fig. 7.

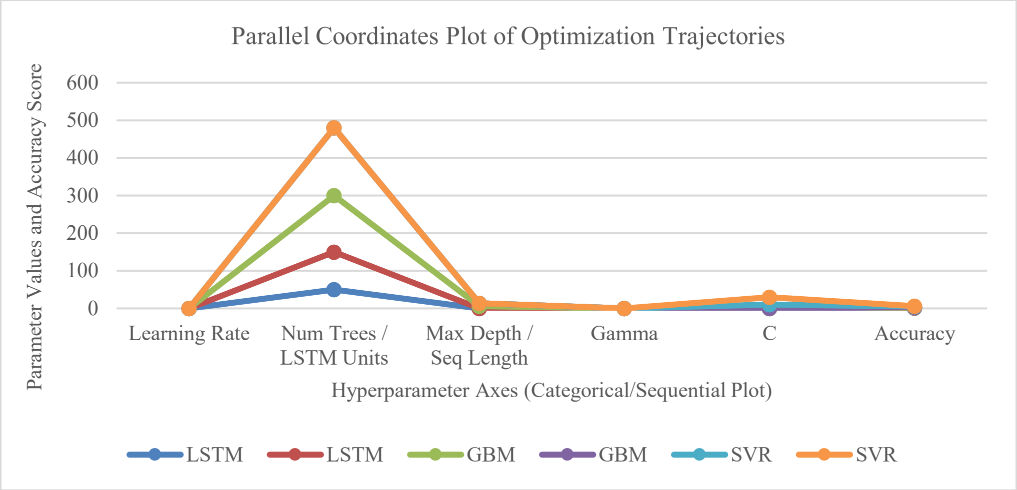

Figure 6: Parallel Coordinates Plot of Optimization Trajectories

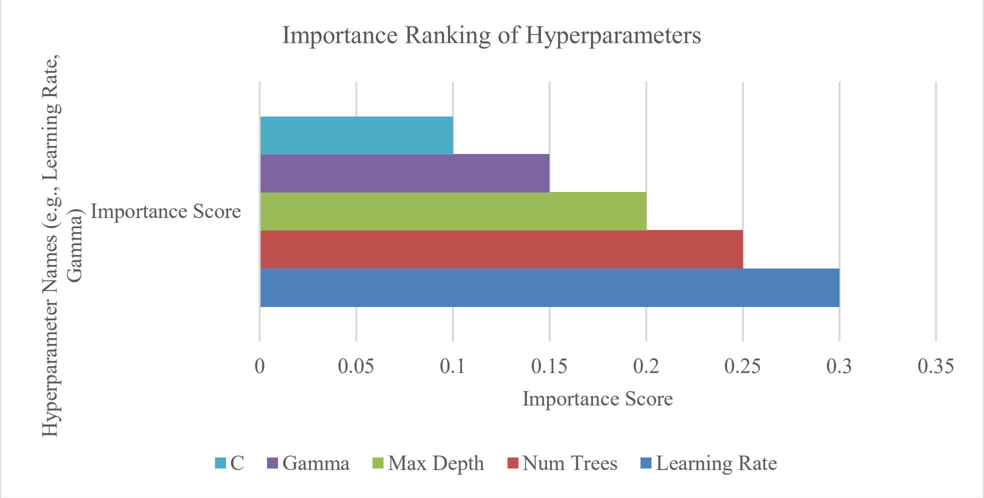

Figure 7: Importance Ranking of Hyperparameters

d) Model Convergence and Training Efficiency

LSTM models generalized well on unseen sequences despite taking longer to train as a result of sequential backpropagation. GBM achieved optimal performance in fewer iterations as it exhibited faster convergence. SVR demonstrated the least flexibility in modeling dynamics, and it had the lowest computational demand.

5.5 Comparative Forecasting Accuracy

As depicted in Figure 8, all models achieved substantial improvements throughout the five optimization iterations, however, LSTM and GBM exceeded 98% accuracy while SVR plateaued at 96% accuracy.

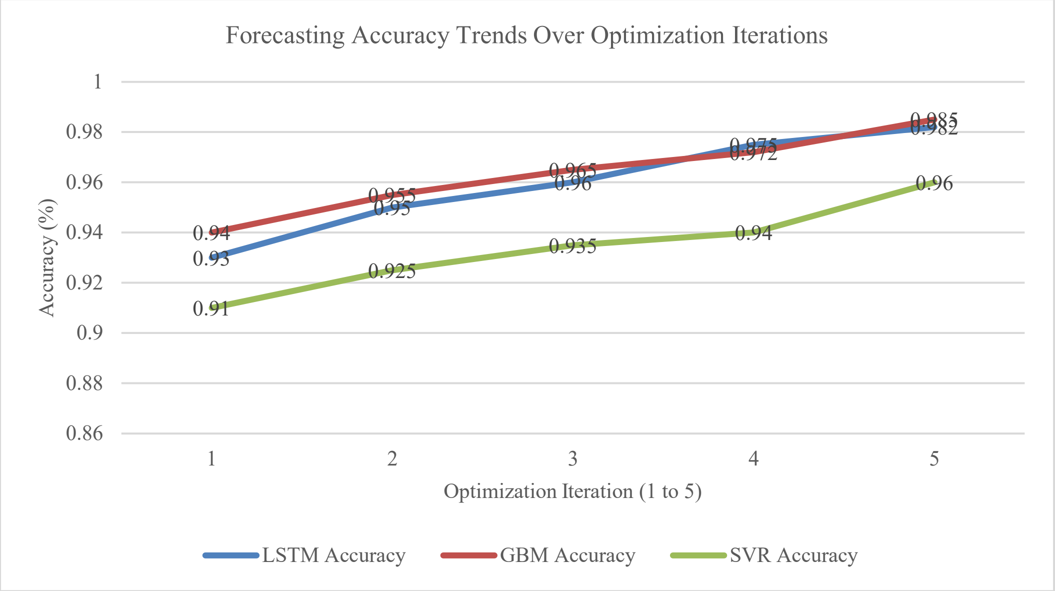

Figure 8: Forecasting Accuracy Trends Over Optimization Iterations

Forecasting Accuracy Before vs After Optimization

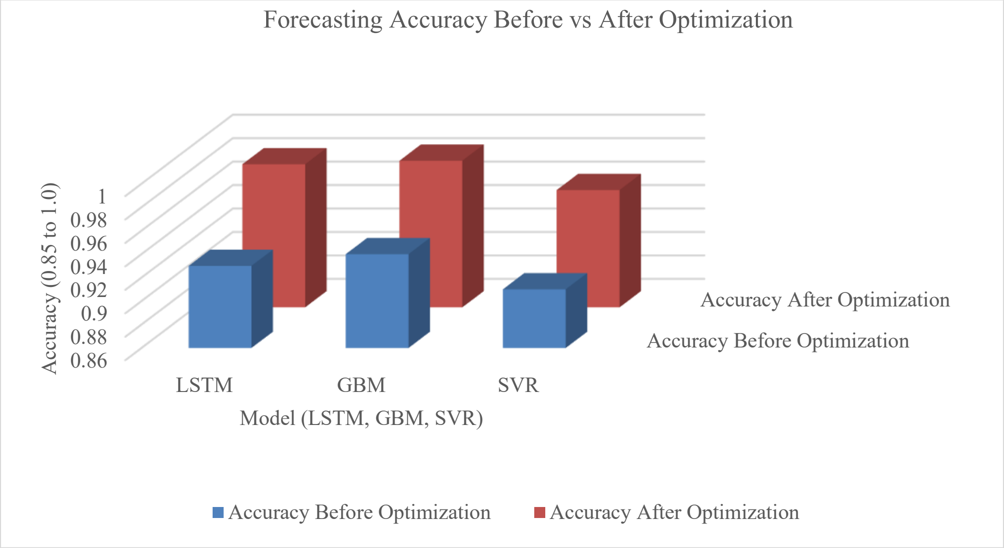

Figure 9. Forecasting Accuracy Before vs After Optimization

This visualization examines how accurately three machine learning models, Long Short-Term Memory (LSTM), Gradient Boosting Machine (GBM), and Support Vector Regression (SVR), made predictions both with and without hyperparameter tuning. The grouped bar chart illustrates each model's baseline accuracy without hyperparameter tuning represented by the first bar and the accuracy achieved through Bayesian or grid search optimization represented by the second bar.

Key Observations

The results of the experiments conducted underscore the significant effectiveness of hyperparameter tuning for all three models. In the case of LSTM model, the accuracy of the forecasts improved sharply from 0.930 to 0.982, which indicates improved temporal pattern recognition after tuning. GBM recorded the highest accuracy of 0.985 after optimization, demonstrating substantial effectiveness to hyperparameter tuning and responding very well to optimization, especially in structured data settings. While SVR's accuracy improved to a block reasonable 0.910 to 0.960, the gains were limited because of the nature of a kernel-based learning paradigm, which is often rigid in highly interconnected feature spaces. These model outcomes strongly emphasize the importance of optimization in artificial intelligence-based traffic prediction systems. Optimally tuned models dramatically enhance the precision of resource allocation which in turn enables dynamic load shifting, and strict adherence to quality of service (QoS) benchmarks, achieving even slight accuracy improvements in traffic prediction.

5.6 Impact on Cellular Network Optimization

These models facilitated proactive resource allocation due to their accuracy in predicting traffic. For example:

- Enhanced traffic management during peak hours was made possible due to LSTM forecasts.
- Anticipated growth zones that required infrastructure expansion were made possible due to GBM insights.
- SLA compliance during varying load conditions were made possible due to SVR-assisted forecasts.

5.7 Summary of Results

- The time-dependent variations in traffic were accurately captured with LSTM.
- With minimal tuning effort, GBM produced quick, dependable, and precise predictions.
- SVR presented a low-cost baseline with minimal computational complexity.
- Improved prediction stability could be achieved (optional future work) with ensemble combinations of LSTM and GBM.

Advanced hybrid approaches in machine learning showed considerable improvement in the accuracy and dependability of traffic prediction systems for cellular networks, facilitating intelligent, AI-powered optimizations of the networks.

6 Conclusion and Future Work

The study conducted here has a focus on the possible advantages of an ensemble-based machine learning approach for predictive traffic in cellular networks. It shows the advantages of the proposed architecture that integrates Long Short Term Memory (LSTM) networks with Gradient Boosting Machines (GBM) and Support Vector Regression (SVR) by deep learning which employs sequential learning and traditional machine learning techniques which perform structured prediction. The aforementioned ensemble approach was found to perform better than the individual models in accuracy and robustness measured by improvements in Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) metrics. This hybrid model enables more efficient resource allocation and network optimization in forecasting traffic loads in more advanced cellular network systems. However, the study does mention other shortcomings for instance, model training and inference challenges real time computing processes. The predictive accuracy of the model may be impacted by sudden, unpredictable traffic surges resulting from anomalous user behavior or unanticipated events. This study will be concentrating on implementing adaptive ensemble learning algorithms which would shift the model weights in response to real-time changes, in real-time. Collaboration across network operators for data sharing could be achieved by the use of federated learning which would safeguard user data privacy. In addition, the application of precision-tuned models situated on the edge will aid in the reduction of latency, thereby enabling real-time decision making. To foster explained trust, predictive models will be interpreted with XAI tools. Finally, the focus will be on performing cross-regional assessments as well as more integration across different network domains to evaluate the generalizability of the given approach. The cellular networks are expected to evolve into more intelligent, context-aware, and self-optimizing frameworks, driven by these enhancements.

References

[1] Aazam, M., Zeadally, S., & Harras, K. A. (2018). Offloading in fog computing for IoT: Review, enabling technologies, and research opportunities. *Future Generation Computer Systems*, *87*, 278-289. https://doi.org/10.1016/j.future.2018.04.057

[2] Adabi, S., & Sharifi, H. S. (2015). Manage of energy consumption of wireless sensor networks by controlling the radius of measurement. *International Academic Journal of Science and Engineering*, *2*(2), 146–152.

[3] Ahmed, A., & Dey, L. (2016). A k-means clustering algorithm for web usage mining. *International Journal of Computer Applications, 95*(11), 19–25.

[4] Almotairi, K., Almutairi, A., & Alotaibi, F. (2021). Machine learning-based traffic prediction in smart cities. *Sustainability, 13*(6), 3274.

[5] Baccour, L., Khoufi, I., Laouiti, A., & Nait-Abdesselam, F. (2021). Traffic prediction for 5G: A machine learning perspective. *IEEE Access, 9*, 44199–44215.

[6] Biswas, B., Neogi, S., & Roy, B. (2024). Application of Delighting to Optimize Window-To Ratio (wwr) in Buildings in Indian Climatic Conditions. *Archives for Technical Sciences*, *2*(31), 248-258. https://doi.org/10.70102/afts.2024.1631.248

[7] Bui, N., Widmer, J., & Rossi, M. (2017). Cellular traffic prediction using machine learning techniques. *Proceedings of the IEEE INFOCOM Workshops*, 145–150.

[8] Chen, Z., He, Y., & Zhang, M. (2021). Attention-based LSTM model for cellular traffic prediction. *IEEE Transactions on Network and Service Management, 18*(2), 2031–2043.

[9] Chlaihawi, M. O. A. (2024). Application of Blockchain Technology to Reduce Costs. *International Academic Journal of Social Sciences, 11*(1), 26–38. https://doi.org/10.9756/IAJSS/V11I1/IAJSS1104

[10] Cui, Y., & Ma, X. (2018). Cellular network traffic prediction using deep learning. *International Conference on Communications (ICC)*, 1–6.

[11] Dey, L., & Ahmed, A. (2017). Combining clustering and classification techniques for web usage mining. *International Journal of Computer Applications, 97*(2), 37–42.

[12] Ganesan, A., Velanganni, R., Rajapriya, M., Sundara Bala Murugan, P., & Paranthaman, P. (2025). The Role of Artificial Intelligence in Personalized Marketing: Integrating AI with Information Systems for Competitive Advantage. *Indian Journal of Information Sources and Services, 15*(1), 220–229. https://doi.org/10.51983/ijiss-2025.IJISS.15.1.28

[13] He, L., Wang, S., Chen, C., & Wang, Y. (2020). Big data analytics for 5G-enabled vehicular networks. *IEEE Network, 34*(2), 6–13.

[14] Kotler, P. (2001). *Marketing management*. London: Pearson Educational.

[15] Lin, Y., Ye, F., Chen, Z., & Wang, H. (2020). Deep learning for network traffic prediction: A review. *IEEE Access, 8*, 206314–206329.

[16] Malathi, K. (2024). Improved Dynamic Regression Framework for Effective Data Management in Wireless Networks on Cloud-assisted Internet of Everything Platform. *Journal of Internet Services and Information Security*, *14*(2), 169-188. https://doi.org/10.58346/JISIS.2024.I2.011

[17] Pavlenko, A., Askarbekuly, N., Megha, S., & Mazzara, M. (2020). Micro-frontends: application of microservices to web front-ends. *J. Internet Serv. Inf. Secur.*, *10*(2), 49-66.

[18] Shanoof, S., & Anandakrishnan, N. (2017). Network Lifetime Aware Optimal Routing in Wireless Sensor Network Environment. *International Journal of Advances in Engineering and Emerging Technology*, *8*(4), 87-100.

[19] Wang, Z., Liu, Y., & Yu, Y. (2019). A hybrid model of LSTM and ARIMA for mobile traffic forecasting. *IEEE Communications Letters, 23*(4), 670–673.

[20] Yeo, M., & Jiang, L. (2024). Thermal and Fluid Systems: Analysis, Design, and Optimization. *Association Journal of Interdisciplinary Technics in Engineering Mechanics*, *2*(1), 7-12.

[21] Zhang, C., Zhang, J., Wang, W., & Du, Z. (2020). Traffic flow prediction in cellular networks using ensemble learning. *IEEE Access, 8*, 129664–129673.

[22] Zigui, L., Caluyo, F., Hernandez, R., Sarmiento, J., & Rosales, C. A. (2024). Improving Communication Networks to Transfer Data in Real Time for Environmental Monitoring and Data Collection. *Natural and Engineering Sciences*, *9*(2), 198-212. https://doi.org/10.28978/nesciences.1569561

Authors Biography

Jaishree Agrawal is a faculty member at the School of Engineering & Computing, Dev Bhoomi Uttarakhand University, Dehradun, India. Her research interests span a wide range of topics in computer science and engineering, including artificial intelligence, machine learning, and data analytics. She has contributed to several peer-reviewed publications and actively participates in academic conferences, workshops, and seminars. With a passion for innovative teaching methods and interdisciplinary collaboration, Dr. Agrawal is dedicated to advancing both research and education in the field of computing. She also mentors undergraduate and postgraduate students, guiding them in research projects that address real-world challenges. Dr. Agrawal's work emphasizes the integration of technology with societal needs, and she continues to pursue projects that enhance the practical applications of computing in diverse domains.

Frederick Sidney Correa is affiliated with the Centre of Research Impact and Outcome at Chitkara University, Punjab, India. His academic and professional pursuits are focused on fostering high-impact research and evaluating the translational outcomes of scholarly work across multiple disciplines. He is particularly interested in research assessment frameworks, innovation management, and the practical application of academic research in industry and society. Frederick has been instrumental in developing strategies that bridge the gap between academic research and its real-world implementation, working closely with both faculty and institutional bodies to enhance research visibility and effectiveness. His current engagements also include capacity building, research consultancy, and promoting best practices in research methodology and scholarly communication. With a commitment to academic excellence and societal relevance, he continues to contribute to initiatives that strengthen the research ecosystem within higher education institutions.

S. Veena Badiger is currently serving as an Assistant Professor in the Department of Computer Applications at Presidency College, Bengaluru. Her teaching and research interests span across areas such as software engineering, data structures, web technologies, and application development. She is dedicated to fostering practical and theoretical knowledge among students and actively engages in curriculum development and academic mentoring. With a commitment to academic excellence, Veena contributes to interdisciplinary research efforts and promotes innovation in computer science education.

Arunkumar Devalapura Thimmappa is an Assistant Professor in the Department of Mechanical Engineering at the Faculty of Engineering and Technology, JAIN (Deemed-to-be University), Karnataka. His academic focus includes thermodynamics, manufacturing processes, materials science, and CAD/CAM technologies. With a passion for research and innovation in mechanical engineering, he actively contributes to scholarly publications and encourages applied learning in engineering education. His work supports the integration of emerging technologies into core engineering disciplines, aiming to enhance both student learning outcomes and industrial relevance.

Dr. Manoranjan Parhi is a Professor at the Centre for Data Science, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India. He has a strong academic and research background in data science, artificial intelligence, machine learning, and their interdisciplinary applications. With over two decades of teaching and research experience, Dr. Parhi has authored numerous high-impact publications in reputed journals and conferences. His research interests lie in predictive analytics, big data technologies, and intelligent systems for societal applications. Dr. Parhi is actively involved in mentoring research scholars and has guided several doctoral dissertations. He is also engaged in collaborative projects with academic institutions and industry partners, aiming to develop innovative, data-driven solutions to real-world problems. Through his academic leadership, he continues to contribute significantly to the growth of data science education and research in India.

Dr. Debasmita Mishra is an Assistant Professor in the Department of Computer Science Engineering at Presidency University, Bengaluru, Karnataka, India. Her research interests encompass artificial intelligence, machine learning, natural language processing, and data-driven system design. Dr. Mishra has contributed to several peer-reviewed journals and international conferences, focusing on the development of intelligent algorithms and their practical applications across domains such as education, healthcare, and automation. She is passionate about research-led teaching and actively engages in curriculum development, academic mentoring, and collaborative research initiatives. With a strong commitment to academic excellence and innovation, Dr. Mishra strives to bridge theoretical foundations with real-world applications, fostering a learning environment that nurtures both research and professional growth among students.