Received 6 June 2024, accepted 15 August 2024, date of publication 21 August 2024, date of current version 30 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3447098

RESEARCH ARTICLE

Estimating Base Station Traffic and Throughput Using Machine Learning Based on Hourly Key Performance Indicator (KPI) Network Analysis

HAJIAR YULIANA[1], HENDRAWAN[1], (Member, IEEE), ISKANDAR[1], (Member, IEEE), AND YASUO MUSASHI[2], (Member, IEEE)

[1]School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Bandung 40132, Indonesia
[2]Research and Education Institute for Semiconductors and Informatics, Kumamoto University, Kumamoto 860-8555, Japan

Corresponding authors: Hajiar Yuliana (33222316@mahasiswa.itb.ac.id) and Hendrawan (hend@itb.ac.id)

ABSTRACT This research focuses on analyzing and predicting traffic and throughput at base stations in cellular networks using machine learning algorithms. The main research area is network performance optimization in telecommunication systems. With the increasing complexity of cellular networks and the need for resource optimization, modeling and predicting network performance has become very important. A model is developed to predict traffic and downlink throughput based on Key Performance Indicators (KPIs) captured hourly from network data. The model is trained using a comprehensive dataset that includes various KPIs. Data was collected from a cellular network site in Bandung, Indonesia, over a four-month period, providing high granularity for analysis. K-Nearest Neighbors (KNN), Random Forest, and XGBoost models were implemented to forecast network parameters. The XGBoost model demonstrated superior performance, with Mean Squared Error (MSE) and R-squared (R^2) values outperforming the other models. Specifically, the XGBoost model achieved an MSE of 0.485 and an R^2 of 0.976 for traffic prediction, and an MSE of 12.382 and an R^2 of 0.943 for downlink throughput prediction. Hyperparameter tuning further optimized model performance. The findings underscore the effectiveness of machine learning in network optimization, contributing to the advancement of 5G technologies. These results offer a promising approach for improving resource allocation and network efficiency.

INDEX TERMS Machine learning, traffic prediction, throughput estimation, base stations, KPI data.

I. INTRODUCTION

Modern telecommunications have become the core of global connectivity, enabling fast and efficient information exchange around the world. In this context, base stations play a crucial role in providing mobile network access to users. However, with the significant increase in the number of users and demand for data services, traffic and throughput management at the base station has become increasingly important to ensure optimal network performance. The rapid increase in mobile data usage has posed new challenges for telecom service providers in managing their networks efficiently. One of the key aspects of network management is the ability to predict and estimate traffic and throughput at base stations. Accurate predictions of traffic patterns and throughput can help service providers plan network resources, optimize performance, and improve user experience. However, predicting traffic and throughput at the base station is a complex task. Fluctuations in usage patterns, variations in network conditions, and other environmental factors can affect network performance in ways that are difficult to predict manually. In seeking solutions to these challenges, machine learning-based approaches offer exciting potential [1].

In telecommunication networks in general, throughput refers to the amount of data that can be transferred over a network in a given period of time. Throughput is usually measured in bits per second (bps), kilobits per second

The associate editor coordinating the review of this manuscript and approving it for publication was Yongming Li.

(kbps), or megabits per second (Mbps). Throughput is an important metric in assessing the performance and efficiency of mobile networks, as it has a direct impact on user experience in terms of download and upload speeds, web browsing, streaming, and other data-intensive activities [2]. On the other hand, communication traffic refers to the volume of data transmitted over a network in a given period of time [3]. Communication traffic covers various types of communication activities, including voice calls, text messages, internet browsing, file downloads, video streaming, and other data transfers. Communication traffic can be measured by the number of users, session duration, amount of data exchanged, or bandwidth consumed.

This research focuses on a key challenge in cellular network management, especially in predicting traffic and base station downlink throughput to improve network efficiency and performance. In increasingly complex modern cellular networks, the inability to accurately predict traffic changes can lead to inefficient use of resources and degraded service quality. Therefore, there is an urgent need to develop reliable prediction methods that can assist network operators in making better decisions. And the objective of this research is to develop a machine learning model capable of predicting traffic and throughput at base stations based on Key Performance Indicator (KPI) data collected on an hourly basis. By using machine learning algorithms, this research aims to provide accurate predictive solutions, which in turn can aid in network optimization and more efficient resource management.

The relationship between traffic and throughput is very close as they affect each other. When traffic increases, the pressure on the network also increases, which can affect throughput. If the throughput is insufficient to handle the high volume of traffic, this can lead to an increase in response time, a decrease in data transfer speed, or even an overall failure of data transmission. On the other hand, optimal throughput can allow the network to handle traffic more efficiently, ensuring fast and responsive data delivery to users. Factors such as base station capacity, signal quality, user density, and network resource management also play an important role in determining throughput quality [4]. Overall, predicting future throughput is critical to ensure the smooth operation, scalability, and competitiveness of mobile networks in meeting evolving user needs and supporting emerging applications and services. Meanwhile, predicting future communication traffic is critical to ensure the scalability, resource usage optimization, reliability, and sustainability of mobile networks in meeting the growing demand for data services and supporting the proliferation of connected devices, applications, and digital experiences in today's increasingly mobile-centric world.

Previous studies related to traffic and throughput prediction have not used machine learning. Prior to machine learning, prediction and estimation of traffic and throughput at base stations in cellular networks were generally done using traditional methods based on statistical analysis and mathematical models [4]. Several approaches are often used before machine learning, such as Simple Statistical Methods, Network Theory-Based Mathematical Models, and Capacity Analysis Methods [4]. While these approaches can provide reasonably good initial estimates in some cases, they are often limited in handling the complexities associated with traffic fluctuations, variations in network conditions, and dynamic usage patterns. This limits the accuracy and precision of their estimates in more complex or changing situations. Therefore, the use of machine learning has become an increasingly popular option due to its ability to handle complex data and discover patterns hidden in network KPI data.

In this study, the use of machine learning (ML) plays a key role in developing prediction models for communication throughput and traffic at base stations. The ML approach enables more complex and accurate analysis of network KPI data, which not only takes into account historical patterns but can also capture complex relationships between various network performance factors. By utilizing ML algorithms, this research explores various approaches in modeling the relationship between network KPIs and communication throughput/traffic. The strength of ML lies in its ability to handle complex and dynamic data, and its ability to learn from patterns in the data without the need for complex manual programming.

The study focuses on utilizing machine learning techniques to estimate base station traffic and throughput based on hourly Key Performance Indicator (KPI) data. This approach is crucial for optimizing network performance and resource allocation in wireless communication systems. By leveraging machine learning algorithms the model aims to predict traffic patterns and available throughput, enabling efficient data transfer and network management [5], [6].

In addition, the use of ML also allows real-time adaptation of the model to changes in the network environment. By monitoring KPI data hourly over the course of a month, the ML model can be continuously updated and adjusted to maintain a high level of predictability. This allows telecom service providers to take faster and more effective actions in optimizing their network performance. Thus, the use of machine learning not only improves the accuracy of communication throughput and traffic predictions, but also opens the door to the development of more adaptive and responsive solutions in mobile network management. The proposed methodology involves processing large datasets of KPIs to train the machine learning models, enabling them to learn complex patterns and relationships within the data. Through experiments on LTE networks, the effectiveness of the method in estimating available throughput during off-peak hours has been evaluated, showcasing its practical applicability in real-world scenarios [7]. Furthermore, the study aligns with previous research that highlights the significance of machine learning in traffic estimation and prediction.

Most of the current communication traffic and throughput prediction models rely on historical data, while the use of other network KPIs (Key Performance Indicators) for this purpose is still limited. KPIs are Key Performance Indicators, which in the context of mobile communications are specific metrics used to assess the performance, efficiency, and quality of a network or system [7], [8]. These metrics provide insights into various aspects of network operation and user experience, helping operators effectively monitor, manage and optimize their networks. KPI indicators are important for measuring network performance and detecting and resolving performance issues. KPI monitoring is critical to ensuring network efficiency and optimization, with the goal of maximizing performance while minimizing resource usage.

Several studies have highlighted the importance of using hourly KPI data to predict traffic and throughput in mobile networks. One study showed that proactively serving predictable user demand by storing data at base stations and user devices can reduce peak traffic demands [8]. In addition, another study showed that real-time traffic analysis with sub-second granularity enables near-real-time estimation of end-to-end performance [9]. Another, statistical modeling and multiple linear regression approaches have also been used for network selection based on KPIs such as Received Signal Code Power (RSCP) and Available Bandwidth (ABW) [10]. In the context of mobile traffic prediction, time series analysis methods such as exponential smoothing have been used for more efficient resource management and QoS improvement [11]. Thus, from various relevant studies, it can be concluded that the use of hourly KPI data has a significant role in predicting traffic and throughput on mobile networks, which can help in efficient and optimal traffic management. However, from these various studies, various predictions being carried out, especially regarding traffic and throughput predictions using KPI network parameters, have not utilized the use of machine learning in them. If there is any recent research related to the use of machine learning in traffic and throughput prediction, the research still uses historical data without utilizing the use of KPI network data.

By utilizing KPI (Key Performance Indicator) data collected from base stations on a regular basis, machine learning algorithms can be used to identify hidden patterns, trends, and relationships between traffic, throughput, and other network factors. Thus, this research aims to explore the potential of machine learning algorithms in helping telecommunication service providers to perform more accurate estimation and prediction of traffic and throughput at base stations in the context of mobile networks in general. It is hoped that this research can make a significant contribution to the development of more efficient and adaptive network management techniques.

In this study, the use of machine learning (ML) also plays a key role in developing prediction models for communication throughput and traffic at base stations. Machine learning approaches enable more complex and accurate analysis of network KPI data, which not only takes into account historical patterns but can also capture complex relationships between various network performance factors. This study shows the utilization of ML algorithm for prediction of throughput and communication traffic at a BTS based on real KPI data monitored hourly for one month from a BTS in Bandung city. In this model, 16 KPIs related to Accessibility, Retainability, Availability, Mobility, and Utilization KPIs are used. The ML prediction regression models used are three different types of ML models namely K-Nearest Neighbors (KNN), Random Forest and XGBoost. The integration of machine learning techniques for base station traffic and throughput estimation based on hourly KPI data presents a promising approach to enhance network performance, optimize resource utilization, and improve overall Quality of Experience (QoE) for users in wireless communication systems.

The main contributions of this paper can be summarized as follows:

1) This study introduces a novel approach to utilize real-time KPI data for communication throughput and traffic prediction at base stations. By monitoring KPI data hourly for one month, this paper demonstrates the ability to generate more accurate and dynamic estimates.
2) In the proposed model, this study uses 16 KPIs relating to key aspects of the network such as Accessibility, Retainability, Availability, Mobility, and Utilization. This reflects a holistic approach in analyzing network performance and throughput prediction.
3) To compare and evaluate the prediction performance, this paper uses three different types of Machine Learning models, namely K-Nearest Neighbors (KNN), Random Forest, and XGBoost. This shows a comprehensive effort to find the most suitable model for communication traffic and throughput prediction applications.
4) By conducting a case study on a Base Station in Bandung city, this paper makes a practical contribution in the context of cellular networks in Indonesia. The results of this case study can provide valuable insights for local telecommunication service providers in improving their network management.
5) The findings of this study have significant practical implications in mobile network management. By improving the prediction of communication throughput and traffic, service providers can increase resource efficiency, optimize network performance, and improve the overall user experience.

This paper consists of six (6) sections, beginning with Introduction, which provides the background and objectives of the research. Section II covers Related Work, discussing relevant previous research. Section III is newly added and titled Machine Learning, where the machine learning algorithms used for predicting traffic and downlink throughput in this study are discussed in detail. Section IV is the

Methodology section, explaining the approach and data collection process. Section V, Results and Discussion, presents the experimental results, model performance analysis, and discusses the implications of the findings. Finally, Section VI, the Conclusion, summarizes the main findings and offers recommendations for future research.

II. RELATED WORK

A. TRAFFIC PREDICTION USING MACHINE LEARNING

Predicting base station traffic is essential for optimizing resource allocation and enhancing terminal user experience. Reducing operating expenses, improving service quality, and controlling network load are all made possible by accurate prediction models. The difficulties of forecasting base station traffic have been investigated using a variety of machine learning and deep learning approaches.

As mentioned earlier, many studies have been conducted to predict and estimate traffic using conventional methods. In the past, traffic prediction still used conventional methods that were quite complex. For example, as done in this research [12] which using Holt-Winters (HW) for time series traffic prediction. And also explain in [13], that many traffic prediction which still uses statistical models in the form of time series models, probability estimation, and particle filtering. This statistical method is based on mathematical statistics and has strict requirements on the stability of the data. In traffic prediction, it is good that the data processed using this method is traffic flow data at a certain point and always follows a certain law. The data has strict periodicity and has high requirements on data and tedious calculations. However, one of the limitations of this mathematical model is that at the same time, due to its static characteristics, it cannot reflect the uncertainty and nonlinear characteristics of the traffic flow process [14].

In other studies, such as in research [15] that focuses on predicting short-term mobile communication traffic using a seasonal product model. This research presents a practical forecasting technology based on the autoregressive moving average model, emphasizing the product seasonal model for short-term prediction. This method determines the order of the product seasonal ARMA equation using the information criterion and estimates the equation parameters through maximum likelihood estimation. This research relies on the product seasonality model for prediction, which may not capture all the complex dynamics of mobile communication traffic, potentially leading to inaccuracies in forecasting under certain conditions. Also, the study in [16] introduced a traffic prediction model called LMA-Deepar, which is based on DeepAR and considers the nonlinearity and nonstationarity of base station cell network traffic. For non-linear and non-stationary network traffic, models such as LMA-DeepAR—which integrate local moving average features—display superior long-term prediction performance and stability. An artificial feature sequence calculation method based on local moving average (LMA) is proposed to capture the distribution characteristics of network traffic. This feature sequence is used as input to DeepAR to improve the prediction performance. The experimental results show that the LMA-Deepar prediction approach outperforms other methods in terms of long-term prediction performance and stability for multi-cell network traffic.

As time goes by, traffic prediction and estimation in mobile communication systems continues to evolve. In addition to utilizing conventional statistical-based methods, there have been many studies that utilize the use of machine learning for traffic prediction and estimation. Also, there are various studies that combine the use of statistical methods with machine learning algorithms. Of course, each method used will have its own advantages and disadvantages.

In recent years, the utilization of machine learning algorithms for traffic prediction in mobile communications has undergone significant development. Various studies and research have shown that machine learning can effectively handle the complexity and dynamics in mobile network traffic patterns. Algorithms such as K-Nearest Neighbors (KNN), Random Forest, XGBoost, and artificial neural networks have been applied with promising results in predicting and estimating traffic at base stations. In [17], using SVM (Support Vector Machine), Logistic, and Decision-Tree-Classifier models to generate the predictions. The outcomes of the experiment demonstrate that a range of machine learning algorithms may be used to more precisely forecast user demand for 5G traffic. Additionally, it can help operators execute more precise traffic bundle marketing for various users. Another study, based on experimental research in [18] that SVM, LR, and DT are the most accurate algorithms among the supervised learning approaches.

Among the many advantages of machine learning, it can predict traffic patterns and adjust how much power is used based on those predictions. This can save energy without compromising user experience. In the trial, this approach reduced energy consumption by 14% while still maintaining performance [19].

In addition to the use of various classical machine learning algorithms, the use of deep learning algorithms also plays an important role in traffic prediction in cellular networks. Deep learning offers an approach that can handle complexity and non-linearity in cellular network data, especially in time series traffic prediction [20]. For time series traffic prediction, the dynamics of cellular traffic are studied in [21], as well as the subtleties involved in forecasting future requests to enhance resource efficiency. In [20], when deep learning compared to conventional time series prediction models, the deep learning-based neural network model improves mobile communication traffic prediction accuracy by 21.6%, 33.4%, and 12.5%. The neural network model accurately predicts base station traffic. Also in [22], examines several approaches to time series forecasting for cellular traffic using convolutional and recurrent neural networks.

Long Short-Term Memory (LSTM) is a special type of Recurrent Neural Network (RNN) type deep learning algorithm that is designed to address long-term problems in data sequences. LSTMs are particularly effective in handling sequential data such as time series, which often arise in the context of mobile network traffic prediction [23]. Solutions for time series analysis using LSTMs are provided in [24], however the technical details are missing, the analysis is weak, and the collection of features used for the training and analysis is not mentioned. From another research [25], by accurately forecasting base station traffic, the STL-LSTM model outperforms conventional mobile communications algorithms in terms of performance. Based on that study, claim that base station traffic predictions made using Long Short-Term Memory (LSTM) networks are more accurate because they can effectively handle the volatility and periodicity of base station traffic data.

Other than the LSTM, types of neural network algorithms such as GRU and Bi-GRU are also highly considered and recommended in predicting mobile network traffic. Gated Recurrent Unit (GRU) and Bi-GRU models have shown higher performance and lower error rates compared to traditional time series models like AR and ARIMA [26]. In [26], this study focuses on predicting base station network traffic using the GRU (Gated Recurrent Unit) neural network model which is a type of deep learning algorithm. The study compared the GRU model with traditional models such as AR, ARIMA, and CNN models. From the results of the study, it shows that the use of the GRU algorithm results in superior prediction performance. The GRU model shows higher performance evaluation metrics and smaller experimental error in mobile communication traffic prediction, optimizing MAE values of 27.04%, 37.89%, and 9.12%. The findings of this study can help operators allocate network resources effectively, improve user experience and guide the development of future network technologies. In addition, the research presented in [27] also focuses on predicting mobile base station traffic using the GRU recurrent neural network model and its enhancements. The study used datasets from the China Universities Big Data Challenge 2021, analyzing key indicators such as the average number of subscribers, PDCP traffic, and activated subscribers. To improve prediction, a Bi-GRU-based model was developed, achieving a Mean Absolute Percentage Error (MAPE) of less than 0.1. And also, in this study [27] proposes a Bi-GRU-based mobile communication base station traffic prediction model, improving the performance of traffic prediction and enabling dormant energy savings strategies for base stations.

Beside using machine learning and deep learning algorithms independently, combining several algorithmic models in traffic prediction is also very helpful in developing better and more accurate prediction results. In [28], where the study involved pre-processing the dataset, feature engineering to mine temporal properties, and using XGBoost-LSTM for base station traffic prediction. The proposed XGBoost-LSTM model showed significant performance improvement compared to other algorithms. However, the combination of XGBoost and LSTM may introduce complexity in model implementation and interpretation.

However, although there have been many utilizations of machine learning in network traffic prediction, there are still limitations in data processing based on the analysis of KPI data that is monitored hourly. KPI (Key Performance Indicator) data measured hourly provides more granular and real-time information about network performance, allowing prediction models to capture changes and fluctuations in a shorter period of time. The use of hourly KPI data can help in identifying more detailed and specific traffic patterns, as well as providing a faster response to changes in network conditions. In spite of this, the majority of the research that has been done, focuses on aggregated historical data or data measured over longer time intervals, such as daily or weekly. This leaves a gap in utilizing the full potential of hourly KPI data to improve the prediction performance and responsiveness of prediction models.

There is one study that has actually also used KPI data to predict traffic, only the predictions made are also influenced by time series traffic data [29]. However, this research still has limitations, where there are still limitations on the variety of KPIs collected by cellular operators which can vary between operators. The KPI data used in this study is only limited to traffic data for the previous hour, user mobility, number of users in the cell, number of handover attempts. The limited amount of traffic data and KPI analysis only taken during this one-hour period is influenced by the dependence on the availability of data which is also limited. This will greatly affect the accurate and complete prediction results for proper analysis. Considering the limitations found in research [29], the current research is based on KPI data taken over a span of 4 months on an hourly basis. In addition, the types of KPIs used in this research were developed more and varied to be able to help produce more accurate traffic prediction results.

Therefore, research that integrates hourly KPI data analysis with machine learning algorithms has the potential to bring significant improvements in traffic and throughput prediction at base stations. With this approach, telecom service providers can gain more detailed and up-to-date insights into their network conditions, which will ultimately improve service quality and user experience.

B. THROUGHPUT PREDICTION USING MACHINE LEARNING

Review in [30] highlights the importance of throughput as contextual network information, and highlights the latest prediction algorithms for predicting network performance. Chen et al. [31] also state that throughput is the key statistic that has the biggest impact on end users' perceptions. Furthermore, when assessing the Quality of IoT-experience (QoIoT) in important industrial services, throughput has a direct impact on productivity [32]. For example, low throughput in remotely operated vehicles could result in the production being stopped down entirely because live footage cannot be broadcast [32]. The authors in [33] and [34] use historical throughput and lower-layer information to directly forecast the available throughput per User Equipment (UE).

The journey of throughput prediction and estimation shows an evolution from traditional methods to the increasingly dominant use of machine learning algorithms. The state of the art regarding throughput prediction in cellular networks using machine learning approaches has shown significant progress. Several studies have successfully developed ML models such as Random Forest, XGBoost, MLP, and SVR to predict throughput with a high degree of accuracy [34]. Real-life experiments under various conditions have been conducted, including in urban, rural, and underground mine environments. Important factors such as SINR have been identified as important determinants in throughput prediction. The use of new technologies such as network slicing, MIMO, and CA has also been the focus of research to improve network performance prediction. Although there are still challenges such as bias and overfitting in the training and test data, this research continues to strive to develop better and more accurate ML models in predicting throughput in LTE and 5G networks.

Initially, throughput prediction and estimation in mobile networks relied mainly on simple statistical approaches. These methods used statistical analysis and mathematical models to identify historical traffic patterns and predict future throughput. Even until 2020, by [35] the use of ARIMA models is still recommended and used for throughput prediction. In that study, the ARIMA model effectively predicted future network throughput with an average error rate of 2.84%, thus helping to improve network protocols and reduce latency. However, the complexity of the prediction process is one of the limitations that still needs to be improved and developed in future research.

However, over time, especially with the increasing complexity and dynamics in the cellular network environment, these statistical methods begin to show their limitations. Fluctuations in traffic patterns, variations in network conditions, and changes in usage patterns complicate accurate throughput estimation using traditional statistical approaches. In response to these challenges, the use of machine learning algorithms began to dominate in the literature related to throughput prediction and estimation. Several studies have proven to be able to handle higher data complexity and discover patterns hidden in network. For instance, Iranfar et al. [36] proposed a machine learning-based framework for throughput estimation in time-varying applications on multi-core servers, emphasizing the use of hardware events for prediction. Similarly, Mohammedali et al. [37] focused on traffic classification using a deep learning approach for end-to-end slice management in 5G/B5G networks, which included a machine learning model for throughput prediction.

Research related to this throughput prediction, one of which is presented in the paper [38]. From this paper, it is stated that this research conducts modeling and prediction of 4G LTE network throughput using various machine learning models. In another step, this research also explores the performance benefits of various machine learning models for output prediction, namely SVR, KNN for regression, ridge regression, and random forest regression. The results show that the random forest model provides the best performance in throughput prediction. The development of this research includes investigating throughput prediction techniques on data with higher granularity and collecting additional data from network operators to improve the performance of throughput prediction models. However, this research still has limitations including the limited data available and the need for further data access from network operators to improve prediction accuracy and performance.

In this study [39] used Machine Learning (ML) approach by selecting Decision Trees (DTs) models such as Random Forest and XGBoost to predict throughput in LTE and 5G networks. They performed hyperparameter optimization to improve model accuracy and performance. This research measures throughput on LTE and 5G networks under different environmental conditions by conducting real-life tests in various scenarios such as driving in urban, sub-urban, and rural areas, as well as in crowded areas such as shopping centers and live sports events. The measurements were carried out using tools that allow to generate network loads in controlled ranges, and the observed throughput values ranged from 1 Mbps to 150 Mbps in LTE and from 50 Mbps to 1.4 Gbps in 5G. The limitation of this study [39] is that it does not consider the use of geo-location as a feature due to the risk of bias towards a particular network setting in one location and the high cost of data collection.

Moreover, the work by Charan et al. [40] introduced a comprehensive machine learning framework for proactive decision-making in wireless networks based on visual data captured by base station cameras. Alkhateeb et al. [41] developed a solution where base stations learn to predict blockages in links using past observations of beamforming vectors, showcasing the predictive capabilities of machine learning in wireless systems. Furthermore, the study by Kassa et al. [42] addressed frame size optimization in WLAN downlink MU-MIMO channels using a machine learning-based adaptive approach, considering factors like traffic patterns and channel conditions. Additionally, Jia et al. [43] explored channel assignment in uplink wireless communication through a machine learning approach, demonstrating the potential of ensemble learning for network optimization.

In addition, there is also a trend that a combination of traditional statistical methods and machine learning algorithms can produce better prediction results. This approach utilizes the strengths of each method to overcome the weaknesses, thereby improving the accuracy and precision of throughput estimation at base stations in cellular networks. As presented in the research paper [44], where this research predicts throughput in time series on the downlink side in 4G networks. The combination of regression neural network and seasonal ARIMA model contributes to the accuracy of User Downlink Throughput prediction in LTE networks by utilizing historical data for feature prediction and optimizing the neural network to achieve minimal Root Mean Square

Error (RMSE). The seasonal ARIMA model predicts features such as DL TRAFFIC GB and DL PRB UTI, which are used as inputs for the neural network to predict UE DL throughput, resulting in more accurate predictions. This integration of machine learning and statistical methods helps improve the accuracy of the LTE network planning process and provides more reliable predictions for user experience. In several studies have also shown the use of hybrid approaches that combine heuristic models, machine learning, and deep learning has been shown to improve throughput prediction accuracy for video streaming and network management [45], [46]. Overall, these studies collectively highlight the diverse applications of machine learning in predicting base station throughput, ranging from workload estimation and traffic classification to proactive decision-making and channel optimization in wireless networks.

III. MACHINE LEARNING MODELS

A. KNN (K-NEAREST NEIGHBOUR)

K-Nearest Neighbors (KNN) is a versatile machine learning algorithm that can be used for classification or regression tasks. In regression, KNN predicts a continuous value for a new data sample based on the values of its nearest neighbors in the training data. In the training phase, the KNN algorithm stores all the training data for reference, i.e. the data set with features (independent variables i.e. KPIs in this example) and target variables (what it wants to predict, i.e. throughput or traffic in this example). When performing prediction for a new data sample with an unknown target value, KNN calculates the distance between this point and all points in the training data using a distance metric such as Euclidean distance. Then the KNN algorithm selects the K closest data points (neighbors) to the new point based on this distance. KNN predicts the target value for the new data point by averaging the target values of its K nearest neighbors. By averaging the nearest neighbors' values, KNN predicts the target value of the new data point that reflects its similarity to the surrounding data. This process is illustrated in Figure 1. Figure 1 is adapted from the architecture and concept of KNN in [47].

B. RANDOM FOREST

Random Forest (RF) algorithm is a supervised machine learning algorithm widely used for classification and regression tasks. RF is based on the concept of ensemble learning, which involves combining the predictions of multiple models to improve overall performance. The algorithm is particularly useful for handling noisy datasets and reducing variance. The algorithm starts by randomly selecting a subset of the training data (with replacement), to create a new dataset. This process is known as bootstrap aggregation or bagging.

For each random subset, a decision tree is built. A decision tree is a type of model that makes predictions by recursively dividing data into smaller subsets based on input feature values. To reduce the correlation between decision trees, a random subset of features is selected for each tree, which is called feature bagging. After all decision trees are trained, the algorithm predicts the regression value for a new input by averaging the predictions of all decision trees. This is done to reduce variance and improve prediction accuracy.

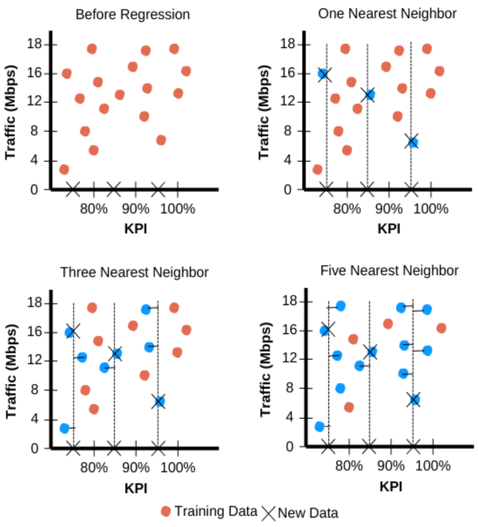


FIGURE 1. KNN algorithm for regression.

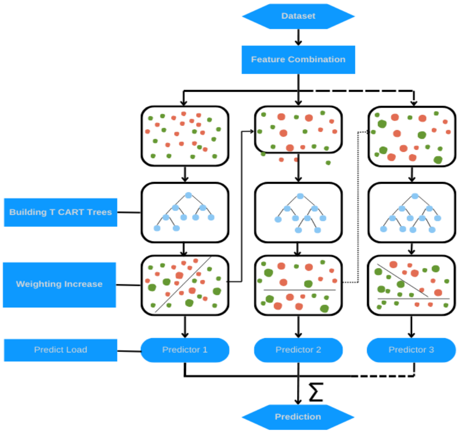


FIGURE 2. Architecture of random forest.

The Random Forest algorithm has several advantages, such as its ability to handle regression and classification tasks, its robustness against outliers and noise. However, it may be computationally expensive and may require more resources compared to other algorithms. It is also less intuitive when dealing with a large number of decision trees [48].

C. XGBoost (EXTREME GRADIENT BOOSTING)

XGBoost (Extreme Gradient Boosting) is a robust machine learning technique suitable for regression tasks. XGBoost extends the concept of gradient boosting by using multiple models (generally decision trees) to form a more robust ensemble. XGBoost is an optimized distributed gradient boosting library designed for efficient and scalable machine learning model training. It is an ensemble learning method that combines the predictions of multiple weak models to produce stronger predictions. XGBoost stands for ''Extreme Gradient Boosting'' and has become one of the most popular and widely used machine learning algorithms due to its ability to handle large data sets and its ability to achieve state-of-the-art performance in many machine learning tasks such as classification and regression.

In Kaggle's Higgs sub-signal detection competition, XGboost—a scalable tree boosting system—was introduced by Chen et al. [50] and has since gained widespread usage.Actually, XGboost is a method for lifting limit gradients [51]. It primarily makes use of the training set to forecast future changes and trends in the target variables. Building several CART trees is the fundamental component of this model. To determine the final prediction value, the model first predicts each tree independently before combining the predictions from each tree. Multiple weak learners are built using the decision tree as the base learner, and the model is then constantly trained along the gradient's falling direction. XGboost works by continuously adding and training new trees to fit the leftover mistakes from the previous iteration. Each instance is given a predicted value, which is calculated by summing the scores of all associated leaves. The model's details are as follows, and Figure 3 depicts the structure. XGBoost can be used in a variety of applications, including Kaggle competitions, recommendation systems, and click-through rate prediction, among others. It is also highly customizable and allows adjustment of various model parameters to optimize performance.

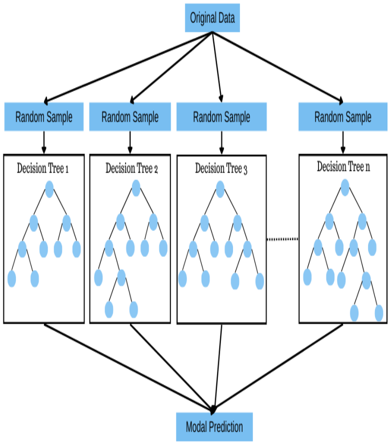


FIGURE 3. Representation diagram of the XGboost model.

The process starts with a base model, often a shallow decision tree, which provides an initial prediction for the target variable in the training data. Next, the error (deviation between predicted and actual values) is calculated for each data point and used as a ''gradient'' indicating the improvement needed in the prediction. A new decision tree is then built to address this error more effectively-with an intensive focus on the data points that the previous model was constrained. By sequentially adding the predictions of this new decision tree to the predictions of the original model, an increasingly accurate ensemble is formed. This iterative process involves continuously building new decision trees that concentrate on correcting previous errors and incorporating them into the collective knowledge base of the ensemble.

Through such iterative steps, XGBoost gradually improves the overall predictive performance. To prevent overfitting, XGBoost integrates regularization techniques during tree construction to encourage generalization of effort by penalizing overly complex trees and encouraging flexible pattern recognition based on data set features.

IV. METHODOLOGY

A. DATA COLLECTION AND DATASET PREPARATION

In this study, data was collected from Key Performance Indicators (KPI) at one of the cellular network sites in Bandung city, West Java, Indonesia. Key Performance Indicators (KPI) are often separated into two categories, there are radio network KPI and service KPI. This study employs three different types of KPIs in this research to track cell edge user throughput and how it relates to traffic load. This KPI data includes various relevant network performance metrics, such as accessibility, retention, availability, mobility, and utilization rates. In addition to this data, historical data on the total traffic and downlink throughput of the site was also obtained over a number of time intervals.

Data collection was conducted regularly on an hourly basis over a four-month period from 23 cell in one network sites. It has 16 KPIs of features and 45,243 rows. The four-month data collection aims to capture seasonal variations as well as usage patterns that may not be visible in a shorter timeframe. Monitoring KPI data on an hourly basis provides high granularity, enabling more detailed and accurate analysis of traffic and throughput dynamics at the base station.

Data pre-processing is essential to ensure the quality and usability of the data before it is fed into the machine learning model. The pre-processing stages include Data Cleaning, Normalization and Standardization, Feature Generation, Feature Selection, and Historical Data Integration. This data collection and pre-processing process is a critical step in ensuring that machine learning models can be trained with high-quality and relevant data. Thus, the model will be able to provide accurate and reliable predictions, which ultimately help in the management and optimization of mobile network performance in Bandung city.

B. FEATURE PARAMETERS AND NETWORK KPIs

Numerous KPIs, which differ from operator to operator, are typically gathered by mobile network operators. According to 3GPP, there are six main categories of KPIs that affect network performance, there are Accessibility, Retainability, Availability, Integrity, Mobility, and Utilization. Accessibility KPIs ensure users can connect to the network and make calls or use data. Accessibility KPIs include Call Setup Success

Rate, which is the percentage of successful call attempts connected, and Packet Delivery Ratio, which is the percentage of successful data packets sent. The Retainability KPI measures how well the network keeps connected users from being dropped for established calls or data sessions. Retainability KPIs include Call Drop Rate, which is the percentage of calls that are disconnected before the session is completed and Handover Success Rate, which is the percentage of successful handovers (switching between cells) during a call or data session. The Availability KPI measures the percentage of time during which a network cell is unavailable. This KPI is used to detect unacceptable performance features and ensure that the network meets the specified requirements. Integrity KPIs ensure error-free data transmission. Integrity KPIs include Bit Error Rate (BER) which is the percentage of data bits received with errors. Mobility KPIs are related to handover (HO) and measure intra-frequency, inter-frequency, and inter-RAT (Radio Access Technology) handover success rates. This KPI helps evaluate the network's ability to maintain continuous service for mobile users. The Utilization KPI measures the utilization of radio resources, such as physical channels and transmission power. This helps operators optimize resource allocation, avoid congestion, and ensure efficient use of available capacity. An illustration of the various KPI categories in a mobile cellular network is shown in Figure 4.

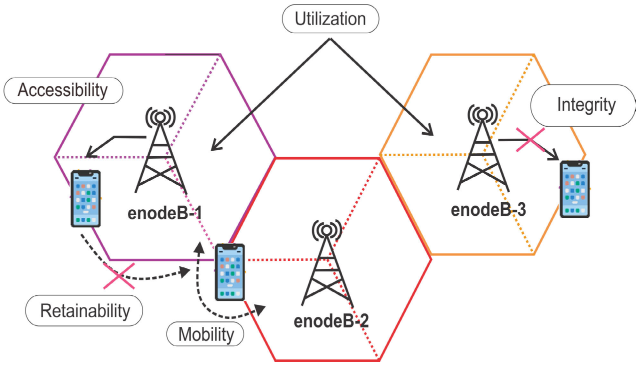


FIGURE 4. Illustration of KPI categories in cellular communication networks.

The types of KPIs used in this research are divided into 3 types of categories, namely Accessibility, Retaintability, and Mobility. Each of these categories has a type of KPI that influences each other on the resulting traffic prediction. The likelihood that a user will be able to access network services within the set tolerances under the given operational conditions is assessed using accessibility KPIs. Mobility KPIs, such as the Hand over Outgoing Success Rate (HOSR), are used to assess how well the network performs in terms of customer experience. Mobility KPIs, such as the Hand over Outgoing Success Rate (HOSR), are used to assess how well the network performs in terms of customer experience.

In this study, data was collected from Key Performance Indicators (KPI) at one of the cellular network sites in Bandung city, West Java, Indonesia. The KPI data is divided into three main categories: Accessibility, Availability, and Mobility. These categories include a total of 16 KPIs, which are critical for assessing and predicting the performance and throughput of the network. The detailed KPIs used in this study are shown in Table 1. These KPIs provide a comprehensive overview of the network's performance, covering aspects from user accessibility to network availability and mobility.

Key parameters for measurement included user traffic volume and downlink throughput. User traffic volume was measured in megabytes, while downlink throughput was measured in megabits per second. Temporal features such as the time of day and day of the week were incorporated to capture daily and weekly usage patterns. These parameters were critical for understanding the network's performance and predicting future traffic and throughput accurately. The models were evaluated using performance metrics such as Mean Squared Error (MSE) and R-squared (R^2) to determine their prediction accuracy.

C. MODELS TYPES AND METRICS

Regarding the types of machine learning models, K-Nearest Neighbors (KNN), Random Forest, and XGBoost algorithms are used for base station traffic and throughput prediction. When selecting model types, this study took into account a range of approaches that are typically recommended for tabular regression tasks. XGBoost builds successive trees that look for the cost function's minima while attempting to lower the error term.

Every algorithm model that use in this study, was chosen have a distinct set of hyperparameters that can be adjusted to increase accuracy and prevent overfitting or underfitting. The set of chosen hyper-parameters for each model is Table 2 and 3. To empirically determine the ideal values, this study employed a grid search with 10-fold cross-validation (CV). The Cross Validation is dividing the data on its own, giving it instructions to split it into 70% train, 10% validation, and 20% test data.

This research study employs R^2 and MSE, two popular statistical measures, for evaluation. The statistical analysis involved using Mean Squared Error (MSE) and R-squared (R^2) as the primary metrics for evaluating the performance of the prediction models. MSE measures the average of the squares of the errors, providing a sense of the average error magnitude.

R-Squared (R^2), or the coefficient of determination, measures the proportion of the variance in the dependent variable that is predictable from the independent variables. These metrics were calculated for both the training and testing datasets to assess model performance and ensure that the models generalize well to unseen data. Additionally, hyperparameter tuning was conducted using grid search and 10-fold cross-validation to optimize model parameters and enhance predictive accuracy.

TABLE 1. Explanation of the key performance indicator (KPIs) used in this Research.

KPI Category	KPIs Name	Explanation
Accessibility	Accessibility Success Ratio	The percentage of successful attempts to access the network compared to the total number of attempts.
Accessibility	RRC Connected User Number	The number of users connected to the Radio Resource Control (RRC) at any given time.
Accessibility	RRC Setup Success Ratio	The ratio of successful RRC setup attempts to the total number of attempts.
Accessibility	RRC Setup Success Rate (Signaling) (%)	The percentage of successful RRC setup signaling procedures.
Accessibility	E-RAB Drop Ratio	The ratio of dropped Evolved Radio Access Bearers (E-RAB) to the total number of E-RABs established.
Accessibility	Call Setup Success Rate (%)	The percentage of successful call setups compared to the total number of call setup attempts.
Accessibility	ERAB Setup Success Ratio	The ratio of successful E-RAB setup attempts to the total number of attempts.
Accessibility	CQI	An indicator of the channel quality experienced by the user, which is crucial for adjusting modulation and coding schemes.
Availability	S1 Signaling Success Ratio	The ratio of successful S1 signaling procedures to the total number of procedures attempted.
Availability	Cell Availability	The percentage of time the cell is available and operational compared to the total observation period.
Availability	CQI	This KPI is also included under availability to reflect the continuous assessment of channel quality.
Mobility	Intra Frequency Handover	The process and success rate of handing over a call or data session within the same frequency band.
Mobility	Intra eNB Handover Success Ratio	The success rate of handovers within the same eNodeB (base station). A high success ratio indicates effective handover processes within the base station.
Mobility	Inter eNB HandoverSuccess Ratio	The success rate of handovers between different eNodeBs. This is crucial for maintaining service continuity as users move across different base stations.
Mobility	CQI	An indicator of the channel quality during handovers, important for maintaining service quality and connection reliability.
Mobility	RSSI	A measure of the received signal strength, crucial for evaluating handover decisions and overall signal quality. This KPI is vital for determining the signal quality and strength during mobility events.

V. RESULT AND DISCUSSIONS

A. HYPERPARAMETER TUNING FOR MACHINE LEARNING MODELS

1) KNN (K-NEAREST NEIGHBOUR)

The value of K (number of neighbors) has a significant impact on model performance. A small K may lead to overfitting (focusing too much on close neighbors that may not be representative in general), while a large K may lead to underfitting (not capturing local patterns). To get the right K value to get the best performing KNN model, a hyperparameter optimization process is performed. Tuning the K value is done experimentally by defining the hyperparameter search space with a grid search procedure (trying every value in the search space) using a 10-fold cross-validation strategy. The K-Nearest Neighbors (KNN) model's hyperparameter tuning procedure is shown in Figure 5, specifically for (a) traffic prediction and (b) downlink throughput prediction. Plotting the mean squared error (MSE) versus various values of the hyperparameter k (neighborhood size) is done through graphs, where the optimal k value is 13 with MSE = 0.679 for traffic prediction and k value = 17 with MSE = 11.084 for Throughput prediction.

In Figure 5(a) for, as k rises from 1 to around 10, the MSE rapidly drops before stabilizing. It was discovered that the ideal value of k for traffic prediction is 13, at which point the MSE hits its minimum and starts to plateau. This suggests that the optimal compromise between bias and variance can be achieved by employing 13 neighbors, which leads to the most precise traffic forecasts. In the same way, Figure 5(b) illustrates the downlink throughput prediction adjusting procedure. From 1 to roughly 10, the MSE declines dramatically; it then declines more steadily until it reaches 17. It was found that 17 is the ideal value of k for downlink throughput prediction, as it minimizes the mean square error. Through its ability to capture the underlying patterns in the throughput data, this ideal k value aids the model in achieving optimal performance. These tuning findings highlight how crucial it is to use the right hyperparameters in order

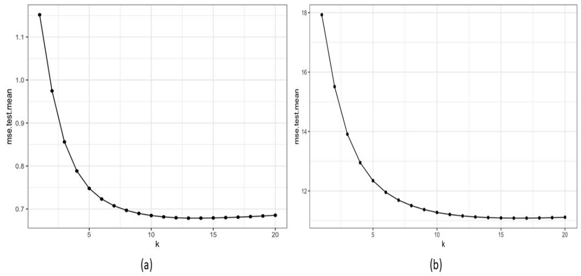


FIGURE 5. Hyperparameter tuning for KNN: (a) Traffic prediction and (b) Downlink throughput prediction.

to increase machine learning models' predictive accuracy. Through careful adjustment of the KNN algorithm's neighbor count, the model's capacity to more accurately forecast throughput and traffic was improved.

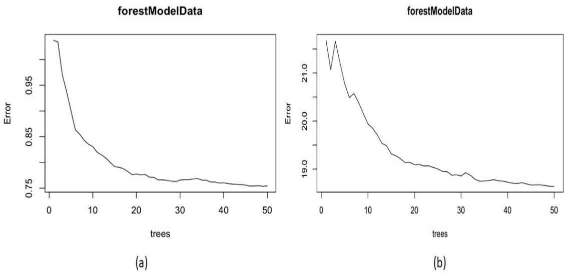
2) RANDOM FOREST

Random Forest has several hyperparameters that can be adjusted to optimize performance, such as the number of trees, maximum depth of trees, minimum samples per leaf and maximum features to be considered at each split. In the case of the traffic and throughput prediction example, the tuning of RF hyperparameters includes the number of individual trees to be trained (ntree), the number of predictor variables randomly sampled for each individual tree (mtry), the minimum number of cases allowed in a leaf node (nodesize) and determining the maximum number of nodes in each individual tree (maxnodes) [49].

Next, the hyperparameter search space is determined and each hyperparameter is defined as an integer number with reasonable lower and upper bounds. Next, a random search is performed with 100 iterations and a holdout cross-validation strategy. The results of hyperparameter tuning for traffic prediction were obtained, ntree = 50; mtry = 11; nodesize = 7; maxnodes = 30; MSE = 0.75. In the same way the results of hyperparameter tuning for downlink throughput prediction are obtained, ntree = 50; mtry = 11; nodesize = 9; maxnodes = 29; MSE = 18.45. The complete results are also presented as in Table 2 and 4.

To assess the predictive performance of Random Forests and to select appropriate values for the tuning parameters, an error estimation technique used in ensemble learning algorithms, namely Out-Of-Bag (OOB) error, is used. This OOB is calculated using observations that are not included in the bootstrap sample or subsample of the original data for each tree. These observations are referred to as out-of-bag (OOB) observations. A plot of the out-of-bag error is shown in Figure 6, where the y-axis shows the mean square error for all cases, predicted by the tree that did not include the case in the training set. The graph shows the variation in the number of trees in the ensemble, where a horizontal line indicates a sufficient number of individual trees in the RF.

The OOB error is a measure of the model's prediction error based on the data not used during the training process, providing an unbiased estimate of model accuracy. As the number of trees increases from 1 to approximately 10, Figure 6(a) shows a dramatic decline in the OOB error for traffic prediction, which subsequently levels off and stabilizes at about 50 trees. This suggests that while initially increasing the number of trees increases the accuracy of the model, the benefit of more trees eventually decreases. Achieving the lowest OOB error possible is roughly 0.75, which is indicative of strong model performance. A similar pattern can be seen in Figure 6(b), where the OOB error for downlink throughput forecast decreases noticeably from 1 to about 15 before stabilizing. The model does a good job of estimating downlink throughput, as evidenced by the lowest OOB error of about 19. But compared to traffic prediction, the error rate is greater, suggesting that downlink throughput prediction might be more complicated and that more data or fine-tuning might be needed to increase accuracy. These numbers emphasize how crucial it is to choose the ideal number of trees in order to strike a compromise between computational efficiency and model accuracy. OOB errors offer a trustworthy gauge of the Random Forest model's predictive power. The model performs well in both traffic and downlink throughput prediction. The knowledge gathered from these findings can direct more optimization and machine learning model application in network performance prediction.



(a)

(b)

FIGURE 6. Out-of-Bag error for random forest: (a) Traffic prediction and (b) Downlink throughput prediction.

3) XGBoost (EXTREME GRADIENT BOOSTING)

XGBoost, like other machine learning models, has several hyperparameters that can be tuned to optimize performance for regression tasks. The main tunable hyperparameters in XGBoost relate to Tree-Specific Hyperparameters, Learning Task-Specific Hyperparameters, and Regularization Hyperparameters [39]. In this study, the following hyperparatemer was tuned to optimize the performance of the XGBoost model:

- *eta* is known as the learning speed, has a value between 0 and 1, which is multiplied by the model weight of each tree to slow down the learning process to prevent overfitting.
- *gamma* is the minimum number of splits for which the node should improve the loss function (MSE in the case of regression).
- *max_depth* is the maximum amount of depth that each tree can grow to.
- *min_child_weight* is the minimum level of impurity required in a node before attempting to split.
- *subsample* is the proportion of cases to be randomly sampled (without replacement) for each tree.
- *colsample_bytree* is the proportion of predictor variables that are sampled for each tree. nrounds is the number of sequentially constructed trees in the model.

Hyperparameter optimization is performed by first determining the type as well as the upper and lower bounds of each hyperparameter to be searched. After the search space is determined, the tuning process is performed and gives the following results for traffic prediction: eta = 0.191;

gamma = 3.74; max_depth = 11; min_child_weight = 6.07; subsample = 0.92; colsample_bytree = 0.902; nrounds = 30, and mse.test.mean = 0.462.

In the same way, hyperparameter optimization was performed for downlink throughput prediction, and the following results were obtained: eta = 0.126; gamma = 4.06; max_depth = 11; min_child_weight = 8.4; subsample = 0.934; colsample_bytree = 0.813; nrounds = 30, and mse.test.mean = 10.597. Using this combination of hyperparameters gives a plot of the number of iterations (number of trees) against RMSE as shown in Figure 7.

In Figure 7(a), the RMSE for traffic prediction decreases sharply during the initial iterations and then gradually stabilizes as the number of iterations increases. This indicates that the model quickly learns the underlying patterns in the traffic data, significantly reducing error in the early stages of training. The RMSE plateaus after about 20 iterations, suggesting that additional iterations provide diminishing returns in terms of error reduction. Similar to this, Figure 7(b)'s RMSE for downlink throughput forecast first exhibits a sharp decrease before gradually stabilizing. Early in the training phase, the model quickly lowers the prediction error; the RMSE stabilizes after 20 iterations. This pattern shows how well the model learns from the data and emphasizes how well the boosting procedure reduces prediction error.

These figures underscore the importance of selecting an optimal number of iterations to balance model performance and computational efficiency. The XGBoost model demonstrates robust performance in both traffic and downlink throughput prediction, with the RMSE plots confirming its ability to effectively learn from the data and make accurate predictions. The insights gained from these results can guide further optimization of machine learning models in network performance prediction, contributing to improved resource allocation and network management.

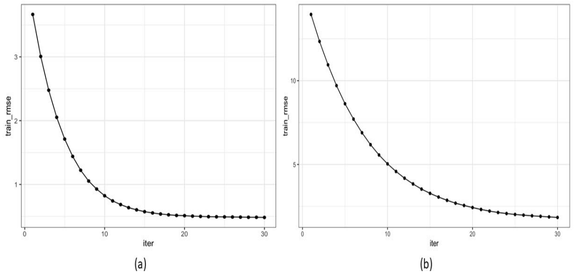


FIGURE 7. Plotting of RMSE against the number of iterations in the boosting process Prediction: (a) Traffic prediction and (b) Downlink throughput prediction.

In the development of machine learning models for traffic and throughput prediction at base stations, hyperparameter settings play a very important role [39]. Hyperparameters are parameters whose values are set before the model training process begins and are not updated during training, in contrast to model parameters that are optimized through the training process. These hyperparameters determine the overall structure and performance of the model. The validation set is used to carry out the hyper-parameter optimization, as recommended by Russell and Norvig [52]. The chosen hyper-parameters for every machine model used in this investigation are shown below in Table 2 for traffic prediction and Table 3 for downlink throughput prediction.

B. TRAFFIC AND DOWNLINK THROUGHPUT PREDICTION MODEL RESULT USING KNN

Figure 8 depicts the relationship between observed and predicted traffic values using the K-Nearest Neighbors (KNN) model and Figure 9 shows the relationship between observed and predicted downlink user throughput using the K-Nearest Neighbors (KNN) model. The scatter plot displays individual data points, with the observed traffic values on the x-axis and the predicted traffic values on the y-axis. The blue line represents the line of perfect prediction, where predicted values would match the observed values exactly.

In Figure 8, the clustering of points along the blue line indicates a strong correlation between observed and predicted traffic values, demonstrating the high accuracy of the KNN model in traffic prediction. The slight deviations from the line represent the prediction errors, which are relatively small, as indicated by the high R^2 value of 0.962. Also in Figure 9, the clustering of points along the blue line indicates a strong correlation between observed and predicted downlink throughput values, demonstrating the high accuracy of the KNN model in throughput prediction. The slight deviations from the line represent the prediction errors, which are relatively small, as evidenced by the R^2 value of 0.873. However, a few outliers can be seen, where the predicted throughput significantly deviates from the observed values, suggesting areas for further model improvement. Despite these deviations, the overall pattern confirms the KNN model's capability to provide reasonably accurate predictions for downlink user throughput.

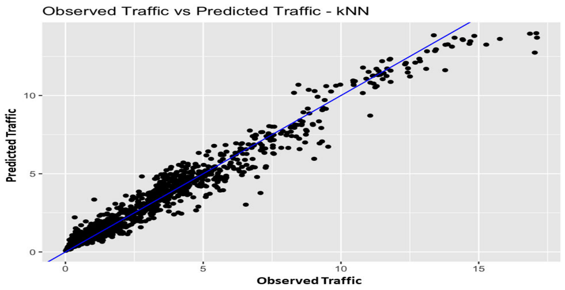


FIGURE 8. Actual vs. Predicted traffic using KNN model.

C. TRAFFIC AND DOWNLINK THROUGHPUT PREDICTION MODEL RESULT USING RANDOM FOREST

The relationship between observed and anticipated values for traffic and downlink throughput, respectively, using the Random Forest model is shown in Figures 10 and 11. The x-axis of Figure 10's scatter plot displays the observed traffic numbers, while the y-axis displays the expected traffic values. The line of perfect prediction, shown by the blue

TABLE 2. Hyperparameter tuning for traffic prediction.

Algorithm	Hyperparameter Tuning For Traffic Prediction
KNN	K = 13
Random Forest	ntree = 50; mtry = 11; nodesize = 7; maxnodes = 30
XGBoost	eta = 0.191; gamma = 3.74; max_depth = 11; min_child_weight = 6.07; subsample = 0.92; colsample_bytree = 0.902; nrounds = 30

TABLE 3. Hyperparameter tuning for downlink throughput prediction.

Algorithm	Hyperparameter Tuning For Downlink Throughput Prediction
KNN	K = 17
Random Forest	ntree = 50 ; mtry = 11; nodesize = 9; maxnodes = 29
XGBoost	eta = 0.126; gamma = 4.06; max_depth = 11; min_child_weight = 8.4; subsample = 0.934; colsample_bytree = 0.813; nrounds = 30

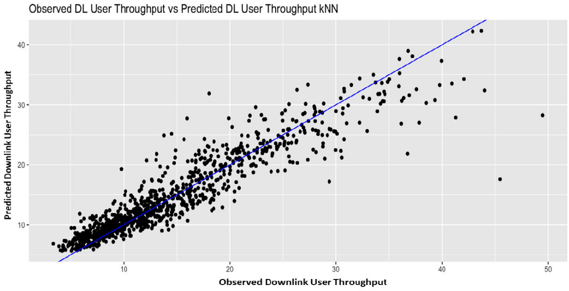


FIGURE 9. Actual vs. Predicted downlink throughput using KNN model.

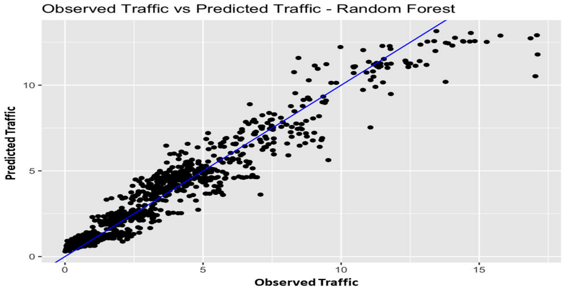


FIGURE 10. Actual vs. Predicted traffic using random forest model.

line, is where the expected values and the actual values coincide exactly. The clustering of points along the blue line illustrates the great accuracy of the Random Forest model in predicting traffic values by showing a good correlation between observed and anticipated traffic levels. Nevertheless, certain departures from the trend are noted, indicating potential domains in which the model's forecasts could be improved.

The observed downlink throughput values are shown on the x-axis of the scatter plot in Figure 11, while the anticipated downlink throughput values are displayed on the y-axis. There is a substantial correlation between the observed and expected downlink throughput numbers, as seen by the points' close clustering along the blue line. There are deviations from the line, especially at higher throughput values, which are similar to the traffic prediction results. These deviations reveal certain prediction inaccuracies that might be fixed with more model tuning.

All things considered, these numbers demonstrate how well the Random Forest model predicts base station traffic and downlink throughput. The near fit to the line of perfect prediction shows that the model has a significant level of predictive accuracy; yet, there is still room for improvement. The results highlight how machine learning models may be used to optimize resource allocation and network performance in cellular networks.

D. TRAFFIC AND DOWNLINK THROUGHPUT PREDICTION MODEL RESULT USING XGBoost

Figures 12 and 13 illustrates the relationship between observed and predicted of traffic and downlink throughput values using the XGBoost model. The scatter plot shows individual data points with observed traffic and downlink throughput values on the x-axis and predicted traffic and downlink throughput values on the y-axis. The blue line represents the line of perfect prediction, where predicted values would exactly match the observed values. The clustering of points along this line indicates a high correlation between observed and predicted values, demonstrating the XGBoost model's high accuracy in traffic prediction. The close alignment with the line of perfect prediction signifies that the XGBoost model effectively captures the underlying patterns in the traffic data. Also for downlink throughput prediction, the high degree of alignment suggests that the XGBoost model is highly effective in predicting downlink throughput, accurately reflecting the actual performance metrics.

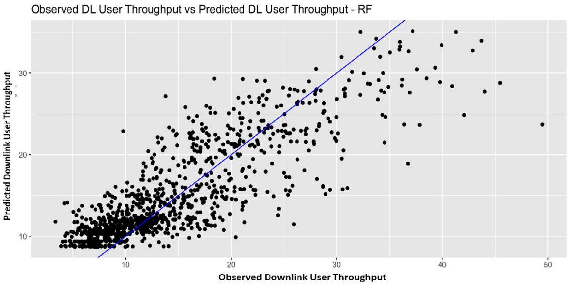


FIGURE 11. Actual vs. Predicted downlink throughput using random forest model.

These figures highlight the superior performance of the XGBoost model in predicting both traffic and downlink throughput at base stations. The high accuracy of the predictions, as indicated by the close fit to the line of perfect prediction, underscores the model's capability to enhance network performance predictions, contributing significantly to resource optimization and network management strategies.

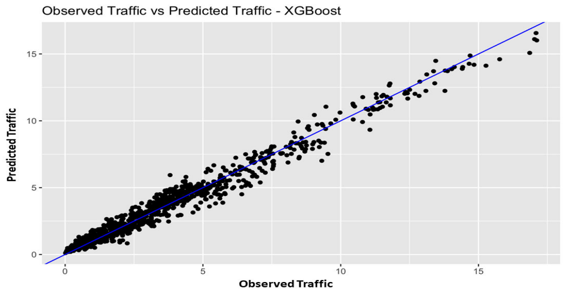


FIGURE 12. Actual vs. Predicted traffic using XGBoost model.

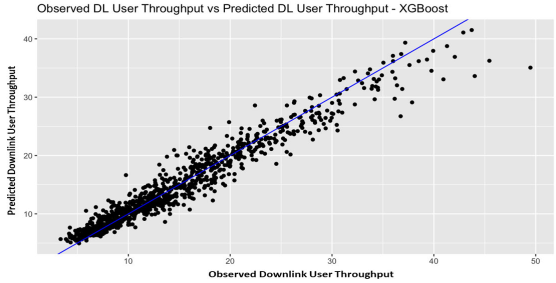


FIGURE 13. Actual vs. Predicted downlink throughput using XGBoost model.

E. COMPARISON OF THREE MACHINE LEARNING ALGORITHMS (KNN, RANDOM FOREST, AND XGBoost)

The performance metrics for traffic prediction and downlink throughput prediction using different algorithms are summarized in Table 4 and shown in Figure 14. For traffic prediction, the K-Nearest Neighbors (KNN) algorithm achieved a Mean Squared Error (MSE) of 0.7423673 and an R^2 value of 0.9618566, indicating high prediction accuracy. The Random Forest algorithm showed a slightly lower MSE of 0.722 but also a lower R^2 value of 0.931. The XGBoost algorithm outperformed the others with the lowest MSE of 0.485 and the highest R^2 value of 0.976, demonstrating superior performance in traffic prediction.

For downlink throughput prediction, KNN achieved an MSE of 12.321 and an R^2 value of 0.873, suggesting reasonable accuracy. Random Forest had a higher MSE of 19.197 and a lower R^2 value of 0.699, indicating less accurate predictions. XGBoost once again demonstrated superior performance with an MSE of 12.382 and an R^2 value of 0.944, showing the highest accuracy among the tested models for downlink throughput prediction.

These results show that XGBoost outperforms KNN and Random Forest in communication traffic prediction, as well as throughput prediction. The lower MSE and higher R^2 values indicate that XGBoost is more effective in capturing the underlying patterns of the data and making accurate predictions. In addition, the iterative nature of XGBoost, combined with its regularization technique, helps prevent overfitting and encourages generalization. This analysis shows that XGBoost is a good machine learning algorithm for the case of predicting communication traffic and downlink throughput. The use of an ensemble method such as XGBoost for communication traffic and downlink throughput prediction can result in a significant increase in accuracy compared to other algorithms such as KNN and Random Forest.

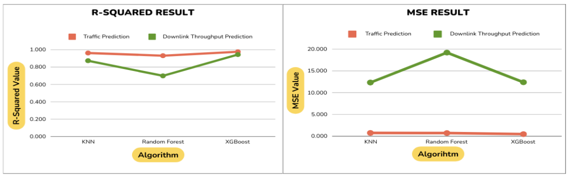


FIGURE 14. Graph of performance metrics for traffic prediction and downlink throughput prediction.

TABLE 4. Evaluation of traffic prediction and downlink throughput prediction.

Algorithm	Traffic Prediction Result Evaluation		Downlink Throughput Prediction Result Evaluation	
	MSE	R^2	MSE	R^2
KNN	0.742	0.962	12.321	0.873
Random Forest	0.722	0.931	19.197	0.699
XGBoost	0.485	0.976	12.382	0.944

The results are consistent with findings in recent studies. For instance, [3] utilized machine learning models for traffic prediction and reported moderate accuracy with traditional algorithms, but their performance was not as high as the results obtained with XGBoost in this study. Similarly, [34] discussed the application of machine learning for throughput prediction, but the accuracy was lower than that achieved by XGBoost model in this study. The use of advanced models like XGBoost, which effectively handles complex data patterns, aligns with the observations of [16], who also highlighted the benefits of using more sophisticated algorithms for network predictions.

The findings from this study have significant implications for network optimization. Accurate traffic and throughput predictions enable network operators to proactively manage resources, reduce congestion, and enhance user experience. By leveraging machine learning models, particularly XGBoost, network operators can achieve more efficient network planning and resource allocation, ultimately improving the overall performance and reliability of mobile networks. These insights support the objectives set forth by [1] regarding network densification and the transition to 5G technologies.

Figure 15 illustrates the comparison between observed traffic values and those predicted by the KNN, Random Forest (RF), and XGBoost (XGB) models over a series of hours. The red line represents the observed traffic,

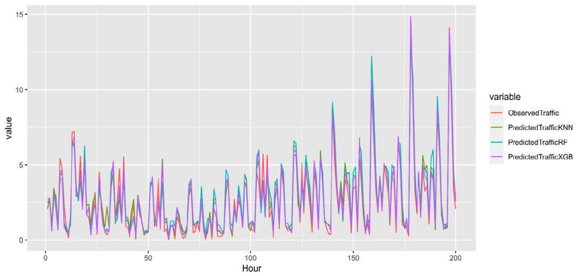


FIGURE 15. Observed vs. Predicted traffic using KNN, random forest, and XGBoost models.

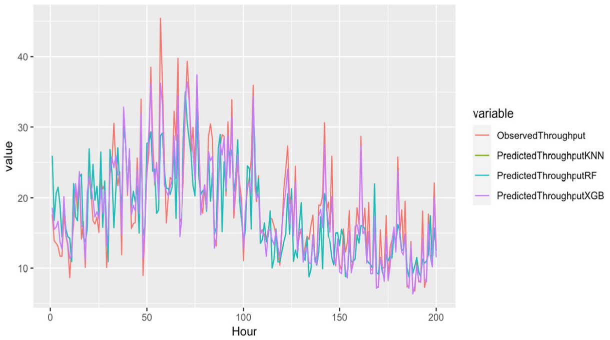


FIGURE 16. Observed vs. Predicted downlink throughput using KNN, random forest, and XGBoost models.

while the green, blue, and purple lines represent the traffic predicted by the KNN, Random Forest, and XGBoost models, respectively. The graph shows that all three models follow the general trend of the observed traffic, capturing the peaks and troughs corresponding to high and low traffic periods. However, the XGBoost model (purple line) consistently tracks the observed traffic more closely than the other models, indicating its superior predictive accuracy. This is in line with the performance metrics, where XGBoost demonstrated the highest R^2 value. The KNN model (green line) also performs well but shows slightly more deviation from the observed traffic, particularly during periods of rapid change. The Random Forest model (blue line) follows the observed traffic trends reasonably well but has a tendency to smooth out some of the fluctuations, which may result in less accurate predictions during periods of high variability.

Also, the comparison of the anticipated downlink throughput values over a period of hours by the K-Nearest Neighbors (KNN), Random Forest (RF), and XGBoost (XGB) models is shown in Figure 16. The throughput predicted by the KNN, Random Forest, and XGBoost models is represented by the green, blue, and purple lines, respectively, while the observed throughput is represented by the red line. As can be seen from the graph, all three models represent the highs and lows associated with times of high and low throughput, generally following the pattern of the observed throughput. The XGBoost model (purple line), as compared to the other models, constantly tracks the observed throughput more closely, demonstrating its higher predictive accuracy. This is consistent with the performance measures, where the highest R^2 value was shown by XGBoost. While it also performs well, the KNN model (green line) deviates slightly more from the observed throughput, particularly when there is a sudden variation. Although the Random Forest model (blue line) helps to smooth out some of the variances, it still tracks the observed throughput patterns. During times of high variability, this could lead to less accurate predictions.

Overall, this Figure 15 and 16 highlights the effectiveness of machine learning models in predicting network traffic and downlink throughput, with XGBoost emerging as the most accurate model among those tested. The close alignment of the predicted traffic and downlink throughput lines with the observed traffic and downlink throughput line reinforces the reliability of these models for traffic forecasting in mobile networks.

In addition, this research can be further explored by implementing one or more of these methods to validate the results obtained in more depth. This is due to the importance of considering these advanced techniques as a further step to strengthen the reliability and validity of the developed predictive model. One method worth considering is Combinatorial Augmentation, which involves enriching the data through the combination of different features to strengthen the generalization ability of the model. This method can result in a more representative and varied dataset, which in turn can improve prediction accuracy. In another development, Combinatorial Interaction Testing and Design of Experiments (DoE) approaches are also active research areas that offer more complex and effective sampling strategies. DoE, for example, allows the identification of interaction effects between variables in an experiment, which can be used to optimize the performance of machine learning models.

VI. CONCLUSION

This study investigates the application of machine learning algorithms to predict traffic and downlink throughput at base stations using hourly Key Performance Indicator (KPI) data collected from a cellular network site in Bandung, Indonesia. This study introduces a novel approach to utilizing real-time Key Performance Indicator (KPI) data for predicting communication throughput and traffic at base stations. The proposed model employs 16 KPIs that encompass critical aspects of the network such as Accessibility, Retainability, Availability, Mobility, and Utilization. This holistic approach enhances the understanding and prediction of network performance. By monitoring KPI data hourly over a four-month period, this research demonstrates the ability to generate more accurate and dynamic estimates. The models implemented included K-Nearest Neighbors (KNN), Random Forest, and XGBoost. The findings indicate that the XGBoost model consistently provided the most accurate predictions for both traffic and downlink throughput, as evidenced by its superior performance metrics and closer alignment with observed data trends.

The hyperparameter tuning process revealed optimal configurations for each model, further enhancing their prediction performance evalution. The results demonstrated significant improvements over traditional prediction methods, highlighting the potential of machine learning techniques in optimizing network performance and resource allocation. These insights contribute to the ongoing efforts in network densification and the transition to 5G technologies, providing a robust framework for enhancing the efficiency of cellular networks. Future research may focus on integrating additional data sources and exploring advanced machine learning models to further improve prediction accuracy.

In the future, this research can be extended by developing new predictive models that combine various machine learning algorithms to improve the accuracy and effectiveness of predictions. The development of this model will enable a more comprehensive analysis, as well as a more in-depth comparison with existing algorithms such as KNN, Random Forest, and XGBoost. This approach is expected to contribute more significant novelty in mobile network optimization.

REFERENCES

[1] N. Bhushan, J. Li, D. Malladi, R. Gilmore, D. Brenner, A. Damnjanovic, R. T. Sukhavasi, C. Patel, and S. Geirhofer, "Network densification: The dominant theme for wireless evolution into 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 82–89, Feb. 2014, doi: 10.1109/MCOM.2014.6736747.

[2] D. Raca, D. Leahy, C. J. Sreenan, and J. J. Quinlan, "Beyond throughput, the next generation: A 5G dataset with channel and context metrics," in *Proc. 11th ACM Multimedia Syst. Conf.*, May 2020, pp. 303–308, doi: 10.1145/3339825.3394938.

[3] X. Chen, J. Wang, H. Li, Y. T. Xu, D. Wu, X. Liu, G. Dudek, T. Lee, and I. Park, "One for all: Traffic prediction at heterogeneous 5G edge with data-efficient transfer learning," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2021, pp. 1–6, doi: 10.1109/GLOBECOM46510.2021.9685204.

[4] R. Borralho, A. Mohamed, A. U. Quddus, P. Vieira, and R. Tafazolli, "A survey on coverage enhancement in cellular networks: Challenges and solutions for future deployments," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 2, pp. 1302–1341, 2nd Quart., 2021, doi: 10.1109/COMST.2021.3053464.

[5] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of machine learning in wireless networks: Key techniques and open issues," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3072–3108, 4th Quart., 2019, doi: 10.1109/COMST.2019.2924243.

[6] S. M. Khan, S. Islam, M. Z. Khan, K. Dey, M. Chowdhury, N. Huynh, and M. Torkjazi, "Development of statewide annual average daily traffic estimation model from short-term counts: A comparative study for South Carolina," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2672, no. 43, pp. 55–64, Dec. 2018, doi: 10.1177/0361198118798979.

[7] K. Satoda, E. Takahashi, T. Onishi, T. Suzuki, D. Ohta, K. Kobayashi, and T. Murase, "Passive method for estimating available throughput for autonomous off-peak data transfer," *Wireless Commun. Mobile Comput.*, vol. 2020, pp. 1–12, Feb. 2020, doi: 10.1155/2020/3502394.

[8] E. Bastug, M. Bennis, and M. Debbah, "Living on the edge: The role of proactive caching in 5G wireless networks," *IEEE Commun. Mag.*, vol. 52, no. 8, pp. 82–89, Aug. 2014, doi: 10.1109/MCOM.2014.6871674.

[9] A. Bernal, M. Richart, M. Ruiz, A. Castro, and L. Velasco, "Near real-time estimation of end-to-end performance in converged fixed-mobile networks," *Comput. Commun.*, vol. 150, pp. 393–404, Jan. 2020, doi: 10.1016/j.comcom.2019.11.052.

[10] A. Rahil, N. Mbarek, M. Atieh, O. Togni, and A. Fouladkar, "Statistical learning and multiple linear regression model for network selection using MIH," in *Proc. 3rd Int. Conf. e-Technol. Netw. Develop.*, Apr. 2014, pp. 189–194, doi: 10.1109/icend.2014.6991378.

[11] Q. T. Tran, L. Hao, and Q. K. Trinh, "A comprehensive research on exponential smoothing methods in modeling and forecasting cellular traffic," *Concurrency Comput., Pract. Exper.*, vol. 32, no. 23, Dec. 2020, doi: 10.1002/cpe.5602.

[12] P. R. Winters, "Forecasting sales by exponentially weighted moving averages," *Manage. Sci.*, vol. 6, no. 3, pp. 324–342, Apr. 1960.

[13] G. M. L. George E. P. Box, G. C. Reinsel, and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*, 5th ed., Hoboken, NJ, USA: Wiley, 2015.

[14] W. Jiang and L. Zhang, "Geospatial data to images: A deep-learning framework for traffic forecasting," *Tsinghua Sci. Technol.*, vol. 24, no. 1, pp. 52–64, Feb. 2019, doi: 10.26599/TST.2018.9010033.

[15] L.-N. Wang, C.-R. Zang, and Y.-Y. Cheng, "The short-term prediction of the mobile communication traffic based on the product seasonal model," *Social Netw. Appl. Sci.*, vol. 2, no. 3, pp. 1–9, Mar. 2020, doi: 10.1007/s42452-020-2158-9.

[16] J. Zhang, X. Zuo, M. Xu, J. Han, and B. Zhang, "Base station network traffic prediction approach based on LMA -DeepAR," in *Proc. IEEE 6th Int. Conf. Comput. Commun. Syst. (ICCCS)*, Apr. 2021, pp. 473–479, doi: 10.1109/ICCCS52626.2021.9449212.

[17] S. Caiyu, W. Jinri, D. Jie, and W. Shanyun, "Prediction of 5th generation mobile users traffic based on multiple machine learning models," in *Proc. IEEE Conf. Telecommun., Opt. Comput. Sci. (TOCS)*, Dec. 2022, pp. 1174–1179, doi: 10.1109/TOCS56154.2022.10016092.

[18] B. Mahdy, H. Abbas, H. Hassanein, A. Noureldin, and H. Abou-Zeid, "A clustering-driven approach to predict the traffic load of mobile networks for the analysis of base stations deployment," *J. Sensor Actuator Netw.*, vol. 9, no. 4, p. 53, Nov. 2020, doi: 10.3390/jsan9040053.

[19] O. Persson and E. Performance, "Breaking the energy curve of mobile networks the ICT enablement effect," Tech. Rep., 2021.

[20] S. Li, E. Magli, G. Francini, and G. Ghinamo, "Deep learning based prediction of traffic peaks in mobile networks," *Comput. Netw.*, vol. 240, Feb. 2024, Art. no. 110167, doi: 10.1016/j.comnet.2023.110167.

[21] A. Azari, P. Papapetrou, S. Denic, and G. Peters, "Cellular traffic prediction and classification: A comparative evaluation of LSTM and ARIMA," in *Proc. 22nd Int. Conf. Discovery Sci. (DS)*, Split, Croatia. Cham, Switzerland: Springer, Oct. 2019, pp. 129–144, doi: 10.1007/978-3-030-33778-0_11.

[22] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling Long- and short-term temporal patterns with deep neural networks," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 95–104, doi: 10.1145/3209978.3210006.

[23] A. A. Kashyap, S. Raviraj, A. Devarakonda, K. S. R. Nayak, and S. J. Bhat, "Traffic flow prediction models—A review of deep learning techniques," *Cogent Eng.*, vol. 9, no. 1, Dec. 2022, Art. no. 2010510, doi: 10.1080/23311916.2021.2010510.

[24] J. Wang, J. Tang, Z. Xu, Y. Wang, G. Xue, X. Zhang, and D. Yang, "Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, May 2017, pp. 1–9, doi: 10.1109/INFOCOM.2017.8057090.

[25] Q. Duan, X. Wei, Y. Gao, and F. Zhou, "Base station traffic prediction based on STL-LSTM networks," in *Proc. 24th Asia–Pacific Conf. Commun. (APCC)*, Nov. 2018, pp. 407–412, doi: 10.1109/APCC.2018.8633565.

[26] T. Xu and Y. Yan, "Cell base station traffic prediction based on GRU," *Comput. Perform. Commun. Syst.*, vol. 7, no. 1, pp. 66–72, 2023, doi: 10.23977/cpcs.2023.070108.

[27] M. Jiang and J. Wang, "Mobile base station traffic prediction based on traffic data analysis," *Acad. J. Comput. Inf. Sci.*, vol. 5, no. 4, pp. 22–28, 2022, doi: 10.25236/ajcis.2022.050404.

[28] Q. Du, F. Yin, and Z. Li, "Base station traffic prediction using XGBoost-LSTM with feature enhancement," *IET Netw.*, vol. 9, no. 1, pp. 29–37, Jan. 2020, doi: 10.1049/iet-net.2019.0103.

[29] L.-V. Le, D. Sinh, L.-P. Tung, and B. P. Lin, "A practical model for traffic forecasting based on big data, machine-learning, and network KPIs," in *Proc. 15th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2018, pp. 1–4, doi: 10.1109/CCNC.2018.8319255.

[30] N. Bui, M. Cesana, S. A. Hosseini, Q. Liao, I. Malanchini, and J. Widmer, "A survey of anticipatory mobile networking: Context-based classification, prediction methodologies, and optimization techniques," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1790–1821, 3rd Quart., 2017, doi: 10.1109/COMST.2017.2694140.

[31] Y. Chen, K. Wu, and Q. Zhang, "From QoS to QoE: A tutorial on video quality assessment," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 2, pp. 1126–1165, 2nd Quart., 2015, doi: 10.1109/COMST.2014.2363139.

[32] D. Minovski, C. Åhlund, and K. Mitra, "Modeling quality of IoT experience in autonomous vehicles," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 3833–3849, May 2020, doi: 10.1109/JIOT.2020.2975418.

[33] C. Yue, R. Jin, K. Suh, Y. Qin, B. Wang, and W. Wei, "Link-Forecast: Cellular link bandwidth prediction in LTE networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 7, pp. 1582–1594, Jul. 2018, doi: 10.1109/TMC.2017.2756937.

[34] D. Raca, A. H. Zahran, C. J. Sreenan, R. K. Sinha, E. Halepovic, R. Jana, and V. Gopalakrishnan, "On leveraging machine and deep learning for throughput prediction in cellular networks: Design, performance, and challenges," *IEEE Commun. Mag.*, vol. 58, no. 3, pp. 11–17, Mar. 2020, doi: 10.1109/MCOM.001.1900394.

[35] D. Lee, D. Lee, M. Choi, and J. Lee, "Prediction of network throughput using ARIMA," in *Proc. Int. Conf. Artif. Intell. Inf. Commun. (ICAIIC)*, Feb. 2020, pp. 1–5.

[36] A. Iranfar, W. S. D. Souza, M. Zapater, K. Olcoz, S. X. d. Souza, and D. Atienza, "A machine learning-based framework for throughput estimation of time-varying applications in multi-core servers," in *Proc. IFIP/IEEE 27th Int. Conf. Very Large Scale Integr. (VLSI-SoC)*, Oct. 2019, pp. 211–216, doi: 10.1109/VLSI-SoC.2019.8920309.

[37] N. A. Mohammedali, T. Kanakis, A. Al-Sherbaz, and M. O. Agyeman, "Traffic classification using deep learning approach for end-to-end slice management in 5G/B5G," in *Proc. 13th Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2022, pp. 357–362, doi: 10.1109/ICTC55196.2022.9952446.

[38] H. Elsherbiny, H. M. Abbas, H. Abou-Zeid, H. S. Hassanein, and A. Noureldin, "4G LTE network throughput modelling and prediction," in *Proc. GLOBECOM IEEE Global Commun. Conf.*, Dec. 2020, pp. 1–6, doi: 10.1109/GLOBECOM42002.2020.9322410.

[39] D. Minovski, N. Ögren, K. Mitra, and C. Åhlund, "Throughput prediction using machine learning in LTE and 5G networks," *IEEE Trans. Mobile Comput.*, vol. 22, no. 3, pp. 1825–1840, Mar. 2023, doi: 10.1109/TMC.2021.3099397.

[40] G. Charan, M. Alrabeiah, and A. Alkhateeb, "Vision-aided 6G wireless communications: Blockage prediction and proactive handoff," 2021, *arXiv:2102.09527*.

[41] A. Alkhateeb, I. Beltagy, and S. Alex, "Machine learning for reliable mmWave systems: Blockage prediction and proactive handoff," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Nov. 2018, pp. 1055–1059, doi: 10.1109/GlobalSIP.2018.8646438.

[42] L. Kassa, J. Deng, M. Davis, and J. Cai, "Frame size optimization using a machine learning approach in WLAN downlink MU-MIMO channel," in *Proc. Data Sci. Mach. Learn.*, Sep. 2022, doi: 10.5121/csit.2022.121521.

[43] G. Jia, Z. Yang, H.-K. Lam, J. Shi, and M. Shikh-Bahaei, "Channel assignment in uplink wireless communication using machine learning approach," *IEEE Commun. Lett.*, vol. 24, no. 4, pp. 787–791, Apr. 2020, doi: 10.1109/LCOMM.2020.2968902.

[44] A. Mostafa, M. A. Elattar, and T. Ismail, "Downlink throughput prediction in LTE cellular networks using time series forecasting," in *Proc. Int. Conf. Broadband Commun. Next Gener. Netw. Multimedia Appl. (CoBCom)*, Jul. 2022, pp. 1–4, doi: 10.1109/CoBCom55489.2022.9880654.

[45] Z. Li, M. A. Kaafar, Xie, and Gaogang, "Session throughput prediction for internet videos," *IEEE Commun. Mag.*, vol. 54, no. 12, pp. 152–157, Dec. 2016.

[46] D. Phanekham, S. Nair, N. Rao, and M. Truty, "Predicting throughput of cloud network infrastructure using neural networks," in *Proc. IEEE INFOCOM Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, May 2021, pp. 1–6.

[47] H. I. Rhys, *Machine Learning With R, the Tidyverse, and MLR*, 2020.

[48] O. Sagi and L. Rokach, "Ensemble learning: A survey," *WIREs Data Mining Knowl. Discovery*, vol. 8, no. 4, pp. 1–18, Jul. 2018, doi: 10.1002/widm.1249.

[49] J. Won, J. Shin, J.-H. Kim, and J.-W. Lee, "A survey on hyperparameter optimization in machine learning," *J. Korean Inst. Commun. Inf. Sci.*, vol. 48, no. 6, pp. 733–747, Jun. 2023, doi: 10.7840/kics.2023.48.6.733.

[50] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[51] Y. Bi, D. Xiang, Z. Ge, F. Li, C. Jia, and J. Song, "An interpretable prediction model for identifying N7-methylguanosine sites based on XGBoost and SHAP," *Mol. Therapy Nucleic Acids*, vol. 22, pp. 362–372, Dec. 2020, doi: 10.1016/j.omtn.2020.08.022.

[52] S. P. Hosea, V. Harikrishnan, and K. Rajkumar, "Artificial intelligence," in *Proc. 3rd Int. Conf. Electron. Comput. Technol.*, vol. 4, Apr. 2011, pp. 124–129, doi: 10.1109/ICECTECH.2011.5941871.

HAJIAR YULIANA received the B.S. degree in electrical engineering and in telecommunication engineering expertise from Universitas Jenderal Achmad Yani, Indonesia, in 2013, and the master's degree in telecommunication engineering from Institut Teknologi Bandung, in 2017, where she is currently pursuing the Ph.D. degree with the School of Electrical Engineering and Informatics. She is also a Lecturer in electrical engineering with Universitas Jenderal Achmad Yani. Her research interests include wireless communication, cellular technology, data science, and applied machine learning algorithms for mobile networks.

HENDRAWAN (Member, IEEE) received the B.S. degree in electrical engineering from Institut Teknologi Bandung, in 1985, and the master's degree in telecommunication engineering and the Ph.D. degree in electronics systems engineering from the University of Essex, U.K., in 1990 and 1995, respectively. His research interests include queuing theory, data communication, multimedia communication, telecommunication networks, network management, big data and machine learning, and artificial intelligence.

ISKANDAR (Member, IEEE) received the B.S. degree in electrical engineering and the master's degree in electrical and telecommunication engineering from Institut Teknologi Bandung (ITB), Indonesia, in 1995 and 2000, respectively, and the Ph.D. degree from Waseda University, Japan, in 2007. From August 1995 to March 1997, he was a Satellite Engineer with Pasifik Satelite Nusantara (PSN), a private telecommunication company. Since April 1997, he has been an Assistant Professor with ITB, where he is currently an Associate Professor with the School of Electrical Engineering and Informatics. His main research interests include wireless and mobile communications, satellite communications, high-altitude platform communications, and non-terrestrial network communications. He has contributed to numerous high-impact journals and conferences in the field of channel propagation and MIMO systems, signal processing, and multiple access schemes.

YASUO MUSASHI (Member, IEEE) has been with the Information Processing Center, Kumamoto University, as a Cooperative Researcher and an Assistant Professor, since 1997. Since 2002, he has also been an Associate Professor with the Center for Multimedia and Information Technologies. He was a Guest Scientist with the Johann Wolfgang Goethe Universität Frankfurt am Main, from January 2005 to July 2005. Since May 2014, he has been with the Center for Management and Information Technologies (CMIT), Kumamoto University. He has been a Full Professor with CMIT, since 2015, and the Research Education Institute for Semiconductors and Informatics (REISI), since 2023. His research interests include computer network security and developing security incident detection and prevention systems.