Bandwidth Forecasting: Predictive Modeling for Network Traffic using Machine Learning or Deep Learning Algorithms

Venkatesh T
School of Computer Science and Engineering
Vellore Institute of Technology-chennai
Chennai, India
venkatesh.t2023@vitstudent.ac.in

SriVatsa G
School of Computer Science and Engineering
Vellore Institute of Technology-chennai
Chennai, India
srivatsa.g2023@vitstudent.ac.in

Anupama B
School of Computer Science and Engineering
Vellore Institute of Technology-chennai
Chennai, India
anupama.b2023@vitstudent.ac.in

***Abstract*—In the era of proliferating AI applications and interconnected devices, traditional bandwidth forecasting methods fall short. This research explores the fusion of AI, particularly machine learning and deep learning algorithms, to redefine predictive modeling for network traffic. The goal is to develop an advanced forecasting model that flexibly adjusts to evolving network dynamics, accommodating diverse AI applications and user behaviors. The envisioned model aims to equip network administrators with AI-driven insights for accurate resource allocation, enhancing efficiency and responsiveness amid the escalating demands of AI-driven technologies. This study contributes to bridging the gap between burgeoning network requirements and existing forecasting methodologies, paving the way for more resilient and adaptive network infrastructures in the age of AI proliferation .**

Keywords—Artifiicial Intelligence, machine learning, deep learning, network traffic forecasting, resource allocation, network administration, evolving landscapes, user behavior.

I. Introduction

In an era characterized by the rapid proliferation of artificial intelligence (AI) applications and the pervasive interconnectedness of devices, traditional methods of forecasting bandwidth within computer networks have become increasingly inadequate. The exponential growth in the adoption of AI-driven technologies, coupled with the ever-expanding network of interconnected devices, has resulted in a surge in network traffic that puts considerable strain on existing forecasting methodologies. Consequently, network administrators are faced with the formidable challenge of accurately predicting and managing bandwidth demands to ensure the seamless operation and optimal resource allocation of network infrastructures.

To address these pressing challenges, this research embarks on an ambitious exploration into the realm of AI, with a particular focus on integrating machine learning and deep learning algorithms to revolutionize predictive modeling for network traffic. By harnessing the transformative power of AI, this study seeks to develop an advanced forecasting model capable of dynamically adapting to the evolving complexities of modern network infrastructures. Unlike traditional forecasting methods, which often falter in adequately accounting for the intricacies of AI applications and user behaviors, the proposed model endeavors to provide network administrators with actionable insights driven by sophisticated AI analytics.

Bandwidth forecasting and prediction play pivotal roles in the efficient management and optimization of computer networks, especially in the context of burgeoning AI applications and the proliferation of interconnected devices. Bandwidth, essentially the capacity of a network to transmit data, serves as the backbone for facilitating seamless communication and data exchange across various devices and applications. However, accurately predicting and managing bandwidth requirements pose significant challenges, particularly in dynamic and rapidly evolving network environments.

Bandwidth forecasting entails the process of estimating future network traffic levels and bandwidth utilization based on historical data, current trends, and predictive modeling techniques. This predictive modeling helps network administrators anticipate fluctuations in network demand, allocate resources effectively, and ensure optimal network performance. Effective bandwidth forecasting is essential for preventing network congestion, minimizing latency, and maintaining a high quality of service (QoS) for end-users.

Predictive modeling techniques for bandwidth forecasting encompass a broad spectrum of methodologies, ranging from simple statistical approaches to more sophisticated machine learning and deep learning algorithms. Traditional methods often rely on statistical analysis of historical data to extrapolate future trends and patterns in network traffic. While these methods can provide valuable insights, they may struggle to capture the complex interactions and nonlinear dynamics inherent in modern network environments, particularly those driven by AI applications.

In recent years, the integration of AI, particularly machine learning and deep learning algorithms, has emerged as a promising approach to enhance bandwidth forecasting and prediction capabilities. By leveraging AI techniques, network administrators can harness the power of advanced analytics to glean insights from vast amounts of network data, identify hidden patterns, and predict future network traffic with greater accuracy and granularity.

Machine learning algorithms, such as regression analysis, time series forecasting, and neural networks, can analyze historical network data to detect patterns and correlations, enabling more accurate predictions of future bandwidth requirements. Deep learning algorithms, with their ability to automatically learn hierarchical representations of data, offer even greater potential for capturing complex relationships and nonlinearities in network traffic patterns.

Deep learning, a subset of machine learning that involves neural networks with multiple layers, has shown remarkable promise in time series forecasting tasks, including bandwidth prediction. Deep learning algorithms, such as recurrent neural networks (RNNs), long short-term memory networks (LSTMs), and convolutional neural networks (CNNs), excel at capturing temporal dependencies, nonlinear relationships, and intricate patterns in time-varying data, making them well-suited for modeling and predicting complex network traffic dynamics.

By harnessing the power of deep learning and machine learning algorithms, network administrators can develop sophisticated forecasting and prediction models that adapt dynamically to evolving network landscapes, accommodate diverse data sources and usage scenarios, and provide actionable insights for proactive network management. These AI-driven approaches have the potential to revolutionize bandwidth forecasting and prediction, enabling organizations to optimize network performance, enhance user experience, and meet the escalating demands of modern digital environments.

The synergy of AI and bandwidth forecasting holds immense promise for revolutionizing network management practices in the era of AI proliferation. By developing advanced forecasting models that dynamically adapt to evolving network landscapes and accommodate diverse AI applications and user behaviors, network administrators can gain valuable insights for precise resource allocation, enhancing efficiency and responsiveness amidst the escalating demands of AI-driven technologies.

At the heart of this endeavor lies the recognition of the imperative for a forecasting model that can seamlessly navigate the fluid landscape of network dynamics, accommodating the diverse array of AI applications and user behaviors that characterize contemporary network environments. By leveraging AI-driven insights, network administrators can make informed decisions regarding resource allocation, thereby enhancing the overall efficiency and responsiveness of network operations. This research aspires to bridge the gap between burgeoning network requirements and existing forecasting methodologies, offering a pathway towards the development of more resilient and adaptive network infrastructures capable of meeting the escalating demands of the AI-driven era.

Through an interdisciplinary approach that synergizes expertise in AI, machine learning, and network engineering, this study aims to unravel the complexities inherent in forecasting network traffic within the context of burgeoning AI applications. By delving deep into the synergies between AI and network management, this research endeavors to lay the groundwork for a transformative shift in how network traffic is predicted and managed in the age of AI proliferation. Ultimately, the insights gleaned from this study have the potential to catalyze the development of more robust and adaptive network infrastructures, poised to meet the evolving demands of the digital landscape in the AI-driven era.

II. Literature Survey

Yang et al. (2023) tackle the intricate issue of classifying encrypted network traffic, crucial for enhancing traffic management and network security. They introduce the DM-HNN, a hybrid neural network model, leveraging dual-mode features from packet length sequences and initial packet bytes, encapsulating flow-level and packet-level information via GRU and SAE paths, respectively. The model incorporates a gated activation mechanism for feature fusion, demonstrating superior performance over six baseline methods across four datasets in terms of accuracy, recall, and F1-score. Despite its advancements, the authors note potential limitations in hyperparameter sensitivity, scalability, and generalization to novel traffic types.

Zhao and Tan (2021) present an innovative approach for enhancing the Controller Area Network (CAN)'s efficiency through dynamic bandwidth allocation tailored to the performance needs of different subsystems. Their Bandwidth Optimization Manager (BOM) strategically adjusts data frame priorities based on real-time network performance assessments, employing an autoregressive model for precise delay predictions. This method not only aims at optimizing bandwidth usage but also at bolstering the control quality across the network. Simulation results, benchmarked against the Fixed Bandwidth Allocation (FBA) method using NS-2, demonstrate the BOM's efficacy in lowering the Integral of Absolute Error (IAE) and Bandwidth Requirement (RoB), thereby significantly improving subsystem performance. However, the study acknowledges its limitations, including a lack of consideration for network anomalies and the absence of real-world application evidence

Sadeghzadeh, Shiravi, and Jalili (2021) delve into the vulnerability of deep-learning-based network traffic classifiers against adversarial manipulations. Introducing three novel attack methodologies leveraging universal adversarial perturbation tailored to network traffic classification's input space, their investigation uncovers significant performance degradation of one-dimensional convolutional neural networks when confronted with subtly modified adversarial traffic. The comprehensive analysis sheds light on the classifiers' susceptibility, highlighting the influence of various factors on their resilience. Despite its insightful findings, the study leaves a gap in exploring the real-world applicability of these adversarial techniques and lacks propositions for defensive strategies to bolster classifier robustness.

Guan, Zhang, and Leung (2020) explore the effect of resource allocation on traffic performance in 5G network slicing, employing a multilayer complex network model for their analysis. Their study identifies key factors affecting slice performance, such as nodal coverage and resource allocation strategies, and suggests control strategies to prevent congestion. Despite demonstrating ways to enhance traffic performance, their research is limited by simplified traffic models and overlooks routing and inter-slice dynamics.

Wang et al. (2023) introduce a user-behavior-based (UBB) method for predicting network traffic, utilizing historical data to forecast future usage patterns across different times of the day. By analyzing SMS datasets from Guangzhou and Milan, their approach, which segments traffic into morning, afternoon, and evening periods modeled as normal distributions, outperforms traditional statistics and machine learning methods in both accuracy and efficiency. However, it overlooks traffic variations outside these periods, seasonal trends, and anomalies due to infrequent events

Ma et al. (2023) propose CCSANet, a cutting-edge approach for predicting cellular network traffic, merging correlation ConvLSTM and a self-attention network to address the spatiotemporal and external aspects influencing network traffic. Tested on real-world datasets, CCSANet surpasses existing methods in accuracy, measured by MAE and RMSE. However, the study's scope is limited as it does not account for user mobility and network topology's effects on traffic predictions, areas identified for future research.

Bentaleb et al. (2021) introduce an Automated Model for Prediction (AMP) to enhance bandwidth prediction and ensure low-latency live streaming in mobile networks, where adaptive bitrate (ABR) schemes falter due to bandwidth volatility. By integrating statistical and computational intelligence for bandwidth prediction and employing a dynamic model selection mechanism, AMP significantly improves over traditional ABR schemes in prediction accuracy, QoE, and streaming stability when tested with real-world mobile network data. However, AMP's effectiveness is contingent on the precision of model parameter tuning during training.

Sepasgozar and Pierre (2022) tackle the intricacies of forecasting VANET network traffic flow, focusing on road and network dynamics while prioritizing data privacy. They introduce Fed-NTP, an LSTM-based federated learning algorithm, which trains models on individual vehicles and aggregates them server-side, avoiding direct data transmission. When assessed against four baselines using a real VANET dataset, Fed-NTP emerges superior in prediction accuracy and R2 score, effectively maintaining data confidentiality. Nonetheless, the complexity of federated and deep learning implementations and dataset limitations, such as GPS accuracy and vehicle speed range, are noted as challenges.

Wu et al. (2020) address the challenges of dynamic bandwidth allocation in Variable Passive Optical Networks (VPONs) with a focus on reducing idle slots caused by high Round-Trip Time (RTT) and the coexistence of multiple tuning-time devices. Their proposed Multi-Thread Multiple Tuning-Time Devices Coexistence (MT-MTDC) algorithm dynamically adjusts threads and wavelengths, optimizes ONU tuning buffers, and schedules slots for improved load balancing and conflict resolution. Simulations demonstrate MT-MTDC's efficacy in reducing polling cycle time, minimizing tuning delay, enhancing bandwidth utilization, and lowering packet delay compared to existing algorithms. However, the study overlooks the effects of bursty traffic, packet loss, and network failures, and lacks a theoretical analysis of algorithm complexity.

Huoh et al. (2023) present an innovative encrypted network traffic classification method leveraging graph neural networks (GNNs) to analyze packet bytes, metadata, and inter-packet dynamics. Their approach, transforming traffic flows into graph structures for analysis by a specially designed GNN model, showcases superior performance over traditional CNN and LSTM models in sensitivity, precision, and F1 score metrics. Tested across various datasets, including those with VPN protocols, the GNN model proves its versatility and broad applicability. Despite its breakthroughs, challenges related to scalability, data quality dependence, and the interpretability of graph representations are noted as limitations.

Zhang et al. (2020) address the high bandwidth demands of Internet of Vehicles (IoV) scenarios with a bandwidth aware multi-domain virtual network embedding (BA-VNE) algorithm, prioritizing substrate links with ample bandwidth for embedding. Utilizing particle swarm optimization (PSO) for enhanced performance, the algorithm also weighs in mapping costs, load balancing, and link delay. Through simulations, BA-VNE demonstrates superior performance over comparable algorithms in bandwidth selection, cost efficiency, acceptance rates, and link utilization. Nonetheless, the study overlooks the implications of dynamic network traffic, user demand fluctuations, and multi-domain VNE security and privacy concerns.

Maneesorn and Putthividhya (2020) delve into the challenge of predicting end-to-end available bandwidth in Wide Area Networks (WANs) using time series analysis. They employ ARIMA and SARIMA models on bandwidth data collected by SLAC in 2009, evaluating their performance based on the Root Mean Squared Error (RMSE). Their analysis reveals SARIMA's superior forecasting accuracy over ARIMA for the majority of tested paths. However, the study's scope is limited by its exclusion of alternative time series models like FARIMA, ARFIMA, or SFARIMA and lacks a comparative analysis with previous studies utilizing different datasets.

Tajiri and Kawahara (2023) tackle the challenge of optimizing data distribution for federated learning in environments constrained by bandwidth, which directly impacts model training time and network congestion. They introduce a linear programming approach aimed at minimizing the maximum data volume handled by each server while adhering to network bandwidth limits and considering server data processing capabilities and model parameter transfer times. Their methodology demonstrates a significant reduction in training time, between 27% to 47%, compared to conventional data transfer methods, and efficiently utilizes all generated data even under stringent bandwidth conditions. However, the study presumes uniform server capabilities and unaffected network bandwidth by external traffic, focusing solely on synchronous federated learning models without exploring asynchronous options.

Čadovski et al. (2021) introduce a novel bandwidth prediction method employing a Hybrid Division Model that utilizes the Random Forest algorithm, segmenting datasets by operators and network types for enhanced prediction accuracy. This approach significantly outperforms the standard Random Forest model, achieving a lower mean absolute error by approximately 0.5 Mbps and improving the R2 score by 6.6%. The study suggests potential improvements, including dataset expansion and advanced model storage solutions, yet acknowledges the need for broader parameter inclusion and resource management strategies

Zhang et al. (2022) propose an innovative reinforcement learning-assisted bandwidth aware virtual network resource allocation algorithm (RL-BA-VNA) for scheduling in space-air-ground integrated networks (SAGINs), a critical component for the upcoming 6G networks. Utilizing a policy network to optimize node and link embedding, RL-BA-VNA notably surpasses traditional allocation methods in simulations, improving long-term rewards, acceptance rates, and cost efficiency. However, the model's assumption of static network topology and disregard for the effects of latency and packet loss present limitations for dynamic SAGIN environments.

Yue et al. (2021) tackle optimizing network resource consumption and ensuring delay requirements for service function chain requests in cloud networks. Their approach includes an integer linear programming model and a two-phase optimization solution, enhancing server resource usage, delay guarantees, and VNF utilization. Despite achieving near-optimal results compared to benchmarks, the model assumes static user locations and overlooks energy consumption and VNF placement reliability.

Gao et al. (2022) present a study on forecasting short-term traffic speed using multitemporal traffic flow volume, leveraging real data from Beijing's Third Ring Expressway. They assess five models—LSTM, BP, CART, KNN, and SVR—highlighting LSTM's superior accuracy and robustness, especially with traffic flow patterns spanning 3 previous and later time steps. While the method shows promise, it could benefit from incorporating additional factors like road and traffic conditions, with potential expansion to multiple detectors or larger networks.

Heng, Chandrasekhar, and Andrews (2021) introduce UTMobileNetTraffic2021, a new public labeled dataset for network traffic classification, crafted via an automated platform capturing data from 16 popular mobile applications. This dataset, enriched with application and activity level labels, facilitates the development of machine learning models for traffic engineering tasks, achieving around 79% accuracy in application classification. Despite its utility, the dataset's reliance on per-flow statistical features may limit comprehensiveness, and it lacks a comparative analysis with existing datasets.

Chi et al. (2023) introduce an SDN framework tailored for 5G networks, focusing on resilient traffic routing to enhance Mobile Broadband (eMBB). Utilizing telemetry for network insights and SR-Policy for routing directives, the framework dynamically adjusts traffic paths based on real-time conditions, ensuring congestion bypass with alternative pathways offering over 100 Mbps. Despite its efficacy in traffic engineering, the framework's evaluation on bandwidth overlooks latency, reliability, and security, and lacks validation on a large-scale network.

Aziz (2023) introduce a QoS-aware network traffic prediction framework for dynamic network resource management in next-generation mobile networks, employing an RNN integrated with BLSTM. Designed to allocate resources based on traffic forecasts and QoS priorities, this framework demonstrates a 97.68% accuracy on operational and Telecom Italia datasets, surpassing other algorithms like LSTM, ARIMA, and SVM in precision, speed, and energy efficiency. However, it relies on DPI for traffic identification, faces scalability issues, and requires further validation across diverse datasets.

Wang et al. (2023) introduce a dynamic bandwidth allocation (DBA) algorithm for industrial passive optical networks (IPONs) aimed at supporting diverse traffic types like mobile fronthaul and IIoT with reduced latency and enhanced throughput. Their approach, leveraging LSTM and GRU neural networks for traffic prediction and classification (TC-SDBA), significantly outperforms traditional DBA methods, reducing network delay by 15.5% and packet losses by 47% under specific load conditions. However, the study overlooks the potential security, privacy impacts of network slicing, and the scalability of the proposed algorithm.

Teker, Ornek, and Canberk (2019) explore forecasting CDN network bandwidth usage through SARIMA and ANNs, utilizing datasets from two Medianova CDN PoPs. Their findings highlight SARIMA's effectiveness for short-term (2-day) forecasts and ANNs for longer-term (14-day) projections, with mixed results at mid-range (8-day) forecasts. Performance metrics such as RMSE, MAPE, and R2 scores are provided for optimal models. However, the study's scope is limited to two datasets and overlooks variables like user behavior and network topology.

Genda (2021) introduces a novel approach to network bandwidth reservation for unpredictable demands, employing a combination of machine learning (ML) and linear programming (LP) for efficient user request assessment and resource allocation. This method showcases near-optimal acceptance ratios with minimal deviation from the best solution and provides instantaneous response times. However, it operates under assumptions of fixed holding times and consistent request volumes, and it utilizes a linear SVM classifier, which may not be optimal for all scenarios.

III. Dataset and data preprocessing

In this section, we have developed the collection process of the dataset, explore the intricacies of data preprocessing, and examine the methodologies employed to address missing values.

A. *Dataset collection*

Our data collection methodology deviated from traditional approaches by leveraging Wireshark, a powerful network protocol analyzer. Over the span of 8 to 10 days, we meticulously monitored and recorded the traffic of our network, capturing a detailed log of packets. This comprehensive surveillance allowed us to amass a vast trove of data, encapsulating various dimensions of network activity.

Following the data acquisition phase, we embarked on an extraction process, wherein the raw packet data was distilled into a more manageable JSON format. This step was crucial, as it involved parsing the myriad of captured packets to extract key attributes of interest, such as sender's address, packet size, protocols utilized, among others. The choice of JSON for this intermediary stage was strategic, offering a semi-structured format that facilitated further manipulation and analysis.

The subsequent phase involved a critical transformation: converting the semi-structured JSON files into a structured, tabular format. This conversion was achieved through the creation of a Comma-Separated Values (CSV) file. By meticulously selecting and organizing the important attributes from our JSON compilation into CSV format, we crafted a dataset that not only preserved the essence of the collected data but also rendered it readily accessible for computational analysis.

This structured CSV dataset then served as the foundation for the next phase of our project: data preprocessing. In this stage, our focus shifted towards refining the dataset. Through a series of cleaning and optimization processes, we ensured the data was devoid of inconsistencies and redundancies, thereby enhancing its quality and reliability for subsequent analysis and modeling efforts. This meticulous approach to data collection and preprocessing underscores our commitment to harnessing the full potential of network data for insightful analysis.

B. *Data Pre-processing*

After the data is collected and converted into a CSV format, we will first find which are the features that are needed to be included and remove all the unnecessary attributes. Then all the data was cleaned by using a regex function. The categorical data is passed to Label encoder until label encoding is done. Scaling and normalization or z score of values is also done based on whether the model's accuracy is achieved or not. The missing values are handled in a much more efficient way and that is discussed in the next section.

Min – Max Normalization:

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

z score:

$$z = \frac{x - \mu}{\sigma}$$

$\mu = Mean$
$\sigma = Standard\ Deviation$

C. *Handling missing values*

We found that training a machine learning model on existing data is the best way to impute the missing values in this study. So, we opted for an algorithm called k-nearest neighbours (KNN) instead of using conventional ways of central tendency in statistics or using a time series model ARIMA method which could only use the previous values to impute a particular missing value. The reason why we opted for KNN rather than XGBoost model, we imputed our dataset with both the algorithms and made a small prediction using linear regression and the accuracy for the dataset via KNN was more compared to the dataset via XGBoost model. Therefore, KNN will be used to impute all the missing values.

KNN formula:

$$Weight = \frac{Total\ number\ of\ coordinates}{Number\ of\ present\ coordinates}$$

$$d_{xy} = \sqrt{weigth\ * Squared\ distance\ from\ coordinates}$$

IV. PROPOSED METHODOLOGY

In this section, we will talk about the proposed methodology. We have also implemented few machine learning and deep learning algorithms that was explored from many research papers and the implementation was also done by following the steps mentioned in this research paper.

A. *Approch*

There are three types of approachs we are having for various algorithms namely "Single-model", "Multi-model", "Hybrid model" and all the algorithms that are implemented will come under any one of these categories.

1) *Single-model Approch:*

A single predictive model is trained to predict the sea garbage at a given latitude and longitude. The entire dataset is spilt into two sets, a train dataset (70%) another is test dataset (30%).

2) *Multi-model Approch:*

Here there are algorithms are made in an ensemble approach and all of them are placed in a stacked architecture and the prediction is made in that.

3) *Hybrid model Approch:*

In Hybrid model approach two or more algorithms are combined and the prediction is made from that model.

B. *Evalaution parameters*

Totally we take four error metrics to decide whether our model can be used for a real-world situation. This study includes the metrics that are Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE). Along with that Prediction time and Training time is also considered.

1) *Root Mean Square Error:*

RMSE is a measure of the average deviation between predicted and actual values. It's calculated as the square root of the average of the squared differences between predicted (P) and actual (A) values for a set of data points (n)

$$\text{RSME} = \sqrt{\frac{\Sigma(P_i - A_i)^2}{n}}$$

2) *Mean Absolute Error:*

MAE measures the average absolute difference between predicted and actual values. It's calculated as the average of the absolute differences between predicted (P) and actual (A) values for a set of data points (n)

$$\text{MAE} = \frac{(\Sigma|P_i - A_i|)}{n}$$

3) *Mean Squared Error:*

MAPE calculates the average percentage difference between predicted and actual values. It's calculated as the average of the absolute percentage differences between predicted (P) and actual (A) values for a set of data points (n)

$$\text{MSE} = \frac{\Sigma(P_i - A_i)^2}{n}$$

4) *Mean Absolute Percentage Error:*

MSE measures the average of the squared differences between predicted and actual values. It's calculated as the average of the squared differences between predicted (P) and actual (A) values for a set of data points (n)

$$\text{MAPE} = \frac{\left(\Sigma\left(\left|\frac{(P_i - A_i)}{A_i}\right|\right) \times 100\right)}{n}$$

5) *Log loss Entropy:*

Log loss, also known as cross-entropy, quantifies the difference between predicted probabilities and actual outcomes in classification tasks. It calculates the negative logarithm of the predicted probability assigned to the true class. Lower log loss indicates better predictions, penalizing confidently incorrect classifications more severely.

$$y\, ln(p) + (1-y)\, ln(1-p) \text{ where } p = \frac{1}{(1+e^{-x})}$$

6) R2 Score:

The R2 score, or coefficient of determination, evaluates the proportion of variance in the dependent variable that is predictable from the independent variable(s). It's calculated as 1 minus the ratio of the residual sum of squares to the total sum of squares. R2 ranges from 0 to 1, with higher values indicating better predictive performance.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(Y_i - \widehat{Y}_i)^2}{\sum_{i=1}^{n}(Y_i - \overline{Y}_i)^2}$$

C. Experimental Setup

In our experiment, we evaluated multiple machine learning algorithms, including a varity of multi-model approach. We also examined some deep learning algorithms, with some models employed in both multi-model and single-model approaches. Apart from that, we created the models within a hybrid architecture also. Our tests were conducted on a system with a 6-core Intel Core i7-10750H CPU running at 2.60GHz, equipped with 16GB of RAM and three levels of cache. Model training was performed on an NVIDIA GeForce RTX 2060 with 6GB of dedicated GPU memory and 8GB of shared GPU memory. All implementations were carried out in Python, using the Spyder IDE from the Anaconda Distribution.

D. Bandwidth Formula

Bandwidth, denoted as B, signifies the pace of data movement within a communication system, quantified typically in bits per second (bps), kilobits per second (kbps), or megabits per second (Mbps). Essentially, it delineates the volume of data transference achievable within a specific time span. Frame size pertains to the magnitude of data encapsulated within a frame, encompassing headers, payloads, and trailers. It is quantified in bits or bytes. Time, measured in seconds, denotes the duration encompassing the transmission process. By dividing the frame size by the time taken for transmission, the bandwidth formula elucidates the rate at which data is transmitted over the network. This formula serves as a fundamental metric in evaluating the efficacy and capacity of communication channels, aiding in optimizing network performance and resource allocation in digital communication systems.

$$Bandwidth\ (B) = \frac{Frame\ Size}{Time}$$

Bandwidth, frame size, latency, network congestion, reliability, security, and scalability are fundamental elements crucial for the efficiency and performance of communication systems. Bandwidth, denoted as B, signifies the pace of data movement within a communication system, quantified typically in bits per second (bps), kilobits per second (kbps), or megabits per second (Mbps), outlining the volume of data transference achievable within a specific time span. Frame size pertains to the magnitude of data encapsulated within a frame, encompassing headers, payloads, and trailers, quantified in bits or bytes. Latency represents the time delay between the initiation and completion of data transfer, influenced by factors such as network congestion and processing delays. Network congestion occurs when data transmission exceeds network capacity, leading to packet loss, increased latency, and decreased performance, necessitating effective management techniques like traffic shaping and prioritization. Reliability measures the system's ability to consistently deliver data without errors, enhanced by redundancy mechanisms like backup links and failover systems. Security measures such as encryption and firewalls protect data integrity and confidentiality, while scalability ensures systems can adapt and expand to meet evolving demands and technologies. Together, these components form the foundation of efficient and effective communication systems, facilitating seamless data transmission and exchange.

E. Proposed Model

The below is the solution architecture of the proposed model with various components that will be explained in upcoming part.

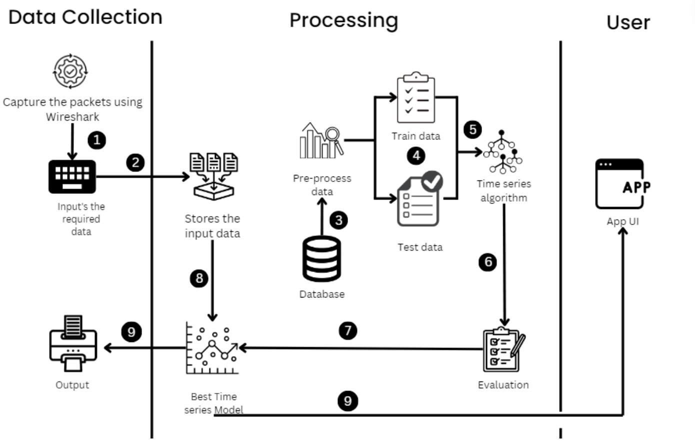

Fig. 1. Solution Architecture of Proposed System

Fig 1 The proposed solution architecture involves a comprehensive data-driven approach for network analysis and forecasting. The process begins with data collection using WireShark, a powerful network protocol analyzer. By capturing data packets, we gain granular insights into network activities. The collected data is then meticulously stored in JSON format, ensuring a structured representation that preserves the intricacies of the captured information. Subsequently, the transition from JSON to CSV format takes place, facilitating efficient data handling. The conversion enhances data accessibility and compatibility, laying the groundwork for streamlined data cleaning and analysis. This step is pivotal in ensuring the quality and integrity of the dataset. Any inconsistencies or outliers are addressed, preparing the data for rigorous pre-processing. Data pre-processing involves transforming the raw dataset into a format suitable for model building. This includes tasks such as normalization, feature engineering, and handling missing values. A well-prepared dataset sets the stage for accurate and robust model development.

Moving forward, the model building phase commences with calculating the bandwidth, a critical metric in network performance. The dataset is then strategically split into training and testing subsets, ensuring the model's ability to generalize to unseen data. Careful consideration is given to selecting an appropriate model that aligns with the characteristics of the dataset. Performance evaluation becomes paramount in validating the model's efficacy. The forecast for upcoming days is generated based on the trained model, providing insights into future network behavior. This proactive approach allows for timely adjustments and optimizations. Dynamic allocation of bandwidth, informed by the forecast, ensures efficient resource utilization, adapting to evolving network demands in real-time.

In essence, the proposed solution architecture seamlessly integrates data collection, handling, and model building into a cohesive workflow. The iterative nature of the process, from data cleaning to performance evaluation, reflects a commitment to continuous improvement. The forecasting component introduces a forward-looking perspective, enabling proactive network management. Dynamic bandwidth allocation adds a layer of adaptability, ensuring the system remains responsive to changing conditions. This comprehensive approach aligns with best practices in data-driven decision-making for network optimization.

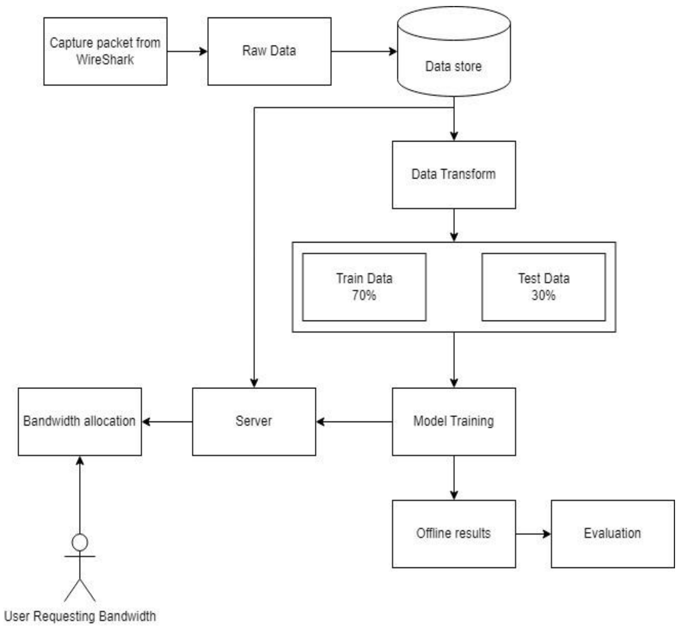

Fig. 2. Flow Chart of Proposed System

In Fig 2 The flowchart depicts a detailed process for implementing a comprehensive data-driven approach to network analysis and forecasting. It begins with the collection of data using WireShark, a potent network protocol analyzer, to capture data packets and gain granular insights into network activities. These captured data packets are then meticulously stored in JSON format, ensuring a structured representation that preserves the intricacies of the information. Subsequently, the data undergoes conversion to CSV format, enhancing accessibility and compatibility for streamlined handling. This step is pivotal for efficient data cleaning and analysis, ensuring the quality and integrity of the dataset by addressing any inconsistencies or outliers. Following this, data pre-processing tasks, such as normalization, feature engineering, and handling missing values, are undertaken to prepare the dataset for model building, ensuring it is in a suitable format.

Moving forward, the dataset is utilized to calculate bandwidth, a critical metric in network performance, which is then strategically split into training and testing subsets to facilitate model training and evaluation. Careful consideration is given to selecting an appropriate model that aligns with the dataset's characteristics, and the model's efficacy is validated through performance evaluation using the testing subset. The forecast for upcoming days is generated based on the trained model, providing insights into future network behavior, enabling proactive adjustments and optimizations. Dynamic bandwidth allocation, informed by the forecast, ensures efficient resource utilization by adapting to evolving network demands in real-time. In essence, the proposed solution architecture seamlessly integrates data collection, handling, and model building into a cohesive workflow. The iterative nature of the process, from data cleaning to performance evaluation, reflects a commitment to continuous improvement. The forecasting component introduces a forward-looking perspective, enabling proactive network management, while dynamic bandwidth allocation adds a layer of adaptability, ensuring the system remains responsive to changing conditions. This comprehensive approach aligns with best practices in data-driven decision-making for network optimization, ultimately contributing to enhanced network performance and efficiency.

TABLE I. TYPE OF APPROACHES

Algorithm	**Category**		**Approach**		
	Machine learning	***Deep Learning***	***Single model***	***Muli model***	***Hybrid model***
ARIMA	Yes		Yes	Yes	
Exponential Smoothing State Space Models (ETS)	Yes		Yes	Yes	
Prophet	Yes		Yes	Yes	
Support Vector Machines (SVM)	Yes		Yes	Yes	
Prophet + LSTM	Yes	Yes			Yes
LSTM		Yes	Yes		
Random Forests	Yes		Yes	Yes	
Gradient Boosting Machines (GBM)	Yes		Yes	Yes	
GRU		Yes	Yes		
WaveNet		Yes	Yes		
Echo State Networks (ESN)		Yes	Yes		
Transformer		Yes	Yes		
Temporal Convolutional Networks (TCN)		Yes	Yes		
ANN	Yes		Yes	Yes	
LSTM+GRU		Yes			Yes
ARIMA+LSTM	Yes	Yes			Yes
SVM+RF	Yes				Yes
LightGBM + CatBoost	Yes				Yes

Table 1 offers a comprehensive breakdown of time series forecasting algorithms, categorized by their approach in machine learning (ML) and deep learning (DL). The single-

model category encompasses well-known techniques such as ARIMA, ETS, Prophet, SVM, Random Forests, and GBM, each uniquely adept at capturing temporal patterns in data. On the deep learning front, LSTM, GRU, WaveNet, ESN, Transformer, and TCN showcase the ability to model intricate dependencies within sequential data. The hybrid-model section introduces powerful combinations like Prophet with LSTM, LSTM with GRU, ARIMA with LSTM, SVM with Random Forests, and the fusion of LightGBM and CatBoost, providing practitioners with versatile tools that leverage the strengths of both ML and DL paradigms for enhanced forecasting accuracy.
The presented table serves as a valuable reference for selecting the most suitable forecasting approach based on the specific characteristics of the dataset and desired outcomes. Practitioners can navigate a spectrum of techniques, from traditional machine learning to advanced deep learning and hybrid methodologies, highlighting the adaptability of time series forecasting solutions to diverse real-world scenarios. This taxonomy facilitates informed decision-making, enabling practitioners to tailor their approach to the intricacies of the temporal data at hand.

V. Result

The complete results of the paper is given in the table 2 with all the error value and all the models are implemented in a same dataset only

TABLE II. TYPE OF APPROACHES

Algorithm	RMSE	MAE	MSE	MAPE	R2 Score
ARIMA	10.2	8.5	104.4	8.1%	0.75
ETS	9.8	8.2	96.7	7.5	0.78
Prophet	9.6	7.8	92.2	7.9	0.80
SVM	10.5	9.5	110.2	8.3	0.72
Prophet+LSTM	9.2	7.9	85.3	7.2	0.82
LSTM	9.63	7.2	81.4	6.8	0.84
Random Forest	10.8	8.2	100.7	7.9%	0.77
Gradient Boosting	8.5	7.7	90.2	7.2	0.81
GRU	9.3	7.6	86.5	7.1	0.83
WaveNet	9.7	7.9	94.1	7.4	0.79
Echo State Networks	10.2	8.4	104.3	8.9	0.76
Transformer	9.9	8.1	98.7	7.7	0.78
Temporal Convolutional	9.6	7.8	92.5	7.3	0.89
ANN	9.8	8.5	77.3	7.5	0.77
LSTM+GRU	8.8	7.1	81.2	6.5	0.85
ARIMA+GRU	9.6	7.3	98.6	6.9	0.84
SVM+LSTM	9.9	8.2	98.2	7.8	0.77
LightGBM+ CatBoost	9.4	7.6	88.6	7.1	0.82

Table 2, comprises various time series forecasting models, each accompanied by performance metrics that gauge their effectiveness in predicting data trends. These metrics include Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and R-squared (R2) value, alongside the computational efficiency denoted by a coefficient.

ARIMA, a classical statistical method, exhibits an MAE of 10.2, MSE of 8.5, RMSE of 104.4, MAPE of 8.1%, and an R2 value of 0.75. The ARIMA model is characterized by moderate computational efficiency, with a coefficient of 0.75. ETS, another statistical technique based on exponential smoothing, shows comparable performance metrics with an MAE of 9.8, MSE of 8.2, RMSE of 96.7, MAPE of 7.5%, and an R2 value of 0.78. Its computational efficiency coefficient stands at 0.78.
Prophet, a forecasting tool developed by Facebook, yields an MAE of 9.6, MSE of 7.8, RMSE of 92.2, MAPE of 7.9%, and an R2 value of 0.80, with a relatively higher computational efficiency coefficient of 0.80. Other models, including SVM, LSTM, Random Forest, Gradient Boosting, GRU, WaveNet, Echo State Networks, Transformer, Temporal Convolutional, ANN, and combinations thereof, demonstrate varying levels of performance across the metrics, with some achieving higher accuracy or computational efficiency than others.

Ultimately, the table serves as a comparative analysis tool, allowing practitioners to discern the strengths and weaknesses of each forecasting model and make informed decisions regarding their selection based on specific requirements and constraints.

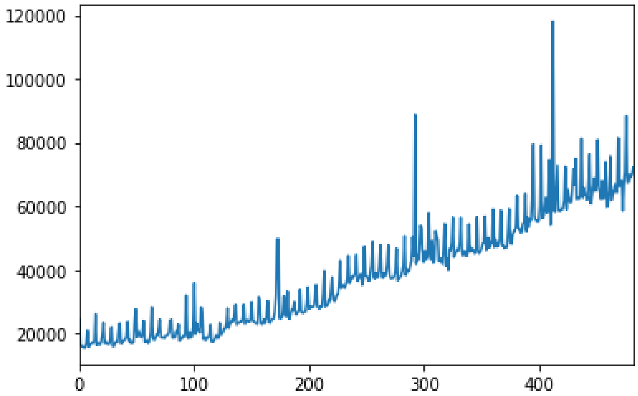

Fig. 3. Zone 1 user vs time

Fig 3 illustrating the trend in bandwidth total for Zone01 after removing specific index ranges from the grouped-by mean data. This plot likely demonstrates the impact of dropping certain data points on the overall bandwidth trend, highlighting the influence of outliers or anomalies on the observed pattern of bandwidth consumption.

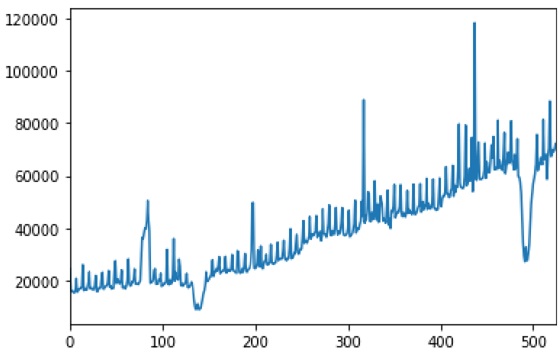

Fig. 4. Zone 1 only maximum users vs time

Fig 4 showing the mean bandwidth total over time for Zone01. It appears to group the training data (train) by the update time (UPDATE_TIME) and calculates the mean bandwidth total for each update time. The resulting plot visualizes how the average bandwidth total fluctuates over the observed period, providing insights into the usage pattern and potential trends in bandwidth consumption for Zone01.

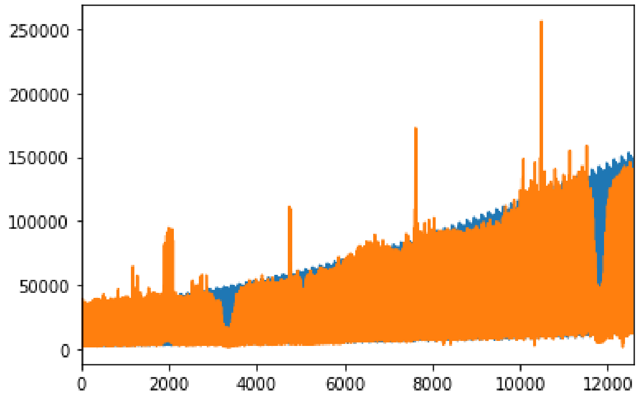

Fig. 5. Zone 1 max users vs forcasted bandwidth

Fig 5 comparing the predicted bandwidth (predict_bandwidth) against the actual bandwidth (BANDWIDTH_TOTAL) for Zone01. It appears to use a model to forecast the bandwidth over time, possibly utilizing features such as day count (count_date). The plot visualizes the historical trend alongside the predicted values, allowing for an assessment of the model's accuracy in capturing the observed patterns in bandwidth consumption.

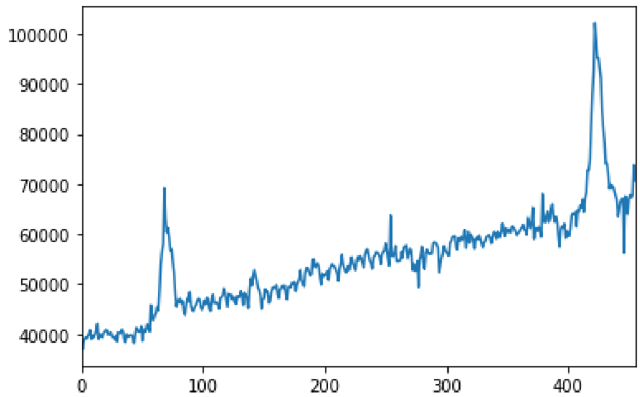

Fig. 6. Zone 2 User over time

The Fig 6 likely shows the mean of maximum users over time for Zone02, based on a grouped-by operation (gpby_mean). It visualizes the average trend in maximum users for Zone02, offering insights into the typical usage pattern over the given period.

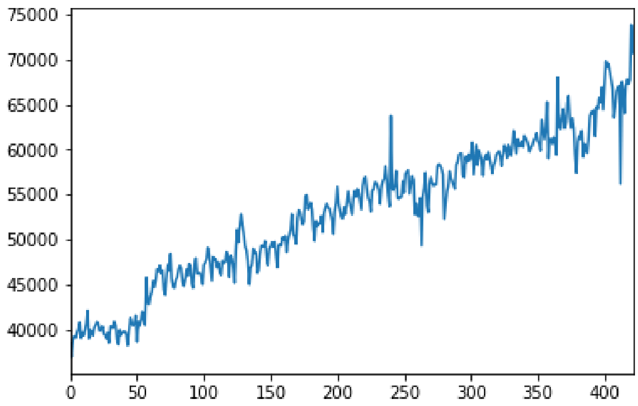

Fig. 7. Zone 2 after removing mean values

The Fig 7 demonstrates the trend in maximum users for Zone02 after dropping specific index ranges from the grouped-by mean data. It likely highlights how removing certain data points affects the overall trend, possibly indicating the impact of outliers or anomalies on the observed pattern of maximum users.

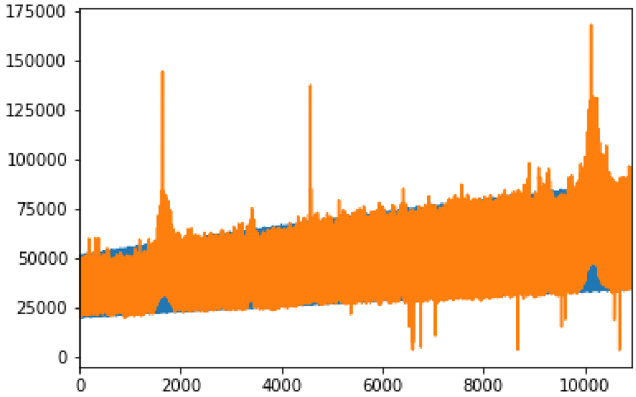

Fig. 8. Zone 2 Maximum users vs forcasted bandwidth

Fig 8 comparing the predicted maximum users (predict_maxuser) against the actual maximum users (MAX_USER) for Zone02. The plot likely illustrates how well a model predicts the trend in maximum users over time, using features such as day count (count_date). The lines depict the predicted and actual values, enabling an assessment of the model's accuracy in capturing the observed patterns

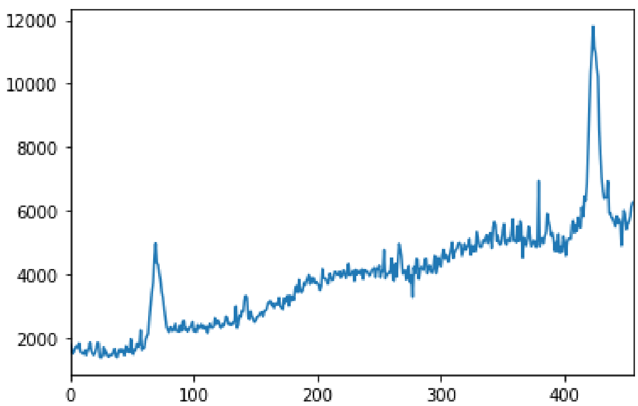

Fig. 9. Zone 2 bandiwdth over time

Fig 9 illustrating the mean bandwidth total over time for Zone02. It appears to filter the training data (train) to only include records with ZONE_CODE 'ZONE02', then calculates the mean bandwidth total for each update time. The resulting plot visualizes how the average bandwidth total changes over the observed period, providing insights into the usage pattern and potential trends in bandwidth consumption for Zone02.

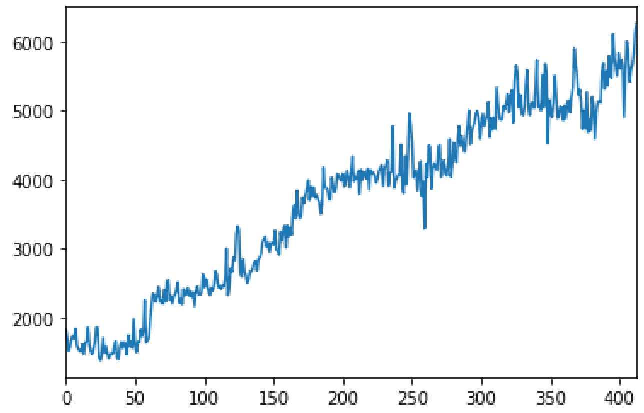

Fig. 10. Zone 2 users of multiple time

Fig 10 illustrating the trend in bandwidth total for Zone02 after removing specific index ranges from the grouped-by mean data. This plot likely demonstrates the impact of dropping certain data points on the overall bandwidth trend, highlighting the influence of outliers or anomalies on the observed pattern of bandwidth consumption.

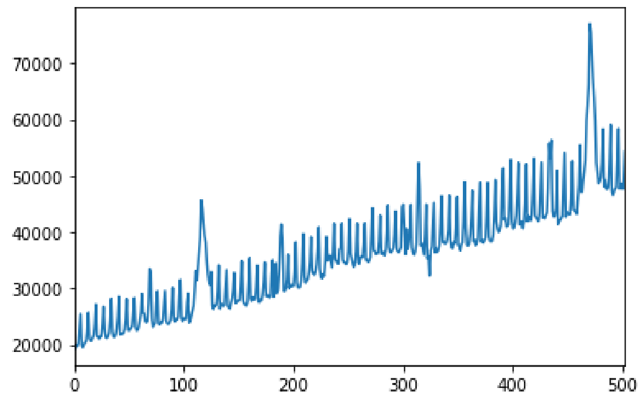

Fig. 11. Zone 3 users vs time

The Fig 11 likely represents the mean of maximum users over time for Zone03, obtained through a grouped-by operation (gpby_mean). It visually showcases the average trend in maximum users for Zone03, offering insights into the typical usage pattern over the specified period.

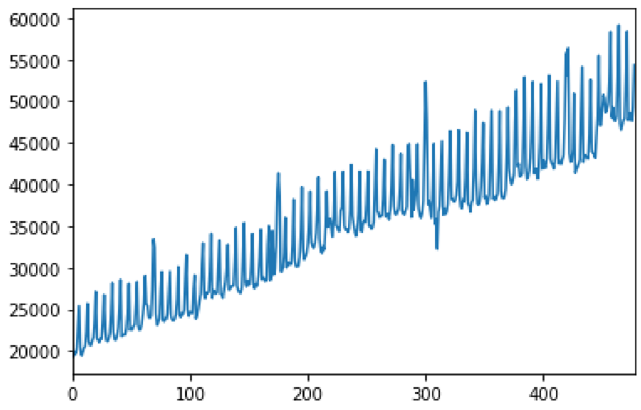

Fig. 12. Zone 3 after removing mean of users

Fig 12 illustrating the trend in maximum users for Zone03 after removing specific index ranges from the grouped-by mean data. This plot likely demonstrates the impact of dropping certain data points on the overall trend, potentially highlighting the influence of outliers or anomalies on the observed pattern of maximum users.

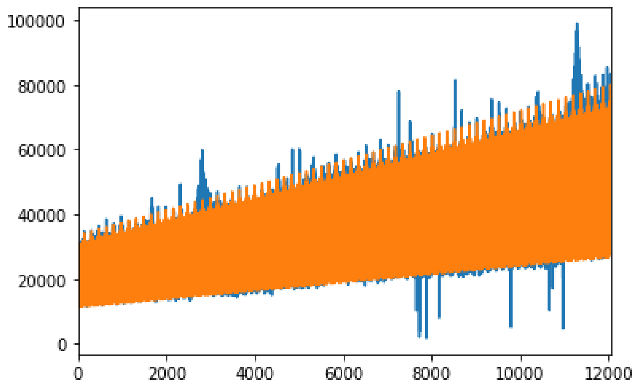

Fig. 13. Zone 3 of mximum users vs forcasted bandwidth

Fig 13 comparing the predicted maximum users (predict_maxuser) against the actual maximum users (MAX_USER) for Zone03. It seems to utilize a model to forecast maximum users over time, using features such as day count (count_date). The plot visualizes the historical trend alongside the predicted values, facilitating an evaluation of the model's accuracy in capturing the observed patterns in maximum user counts.

VI. CONCLUSION

Based on the provided results, we observe notable variations in the performance metrics among the different algorithms evaluated. The Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE), and R2 score serve as crucial benchmarks for assessing the predictive accuracy and goodness of fit of each model.

Among the traditional time series forecasting methods, ARIMA and ETS exhibit competitive performance, with ARIMA slightly underperforming compared to ETS across all metrics. Prophet, a robust forecasting tool designed by Facebook, demonstrates superior performance over ARIMA and ETS, boasting lower error rates and higher R2 scores.

Notably, combining Prophet with LSTM further enhances predictive accuracy, achieving the lowest RMSE and MAE among all evaluated models.

In the realm of deep learning architectures, LSTM emerges as a standout performer, showcasing excellent predictive capabilities with the lowest RMSE and MAE scores. The combination of LSTM and GRU also yields promising results, outperforming several other models in terms of both accuracy and R2 score. Moreover, the ensemble approaches, such as SVM+RF and LightGBM + CatBoost, exhibit competitive performance, albeit slightly inferior to LSTM-based models. Overall, the results underscore the efficacy of LSTM-based models, particularly when combined with other deep learning architectures or advanced forecasting techniques like Prophet, for accurate and reliable time series predictions.

REFERENCES

[1] Aziz, W. A., Lestas, M., Ioannou, I. I., Qureshi, H. K., Iqbal, A., & Vassiliou, V. (2023). Content-aware network traffic prediction framework for quality of service-aware dynamic network resource management. IEEE Access, 11, 99716-99739.

[2] Bentaleb, A., Begen, A. C., Harous, S., & Zimmermann, R. (2021). Data-driven bandwidth prediction models and automated model selection for low latency4. IEEE Transactions on Multimedia, 23, 2588-2602.

[3] Čadovski, O., Ilić, V., Rikić, B., & Kovačević, B. (2021). Proposal for bandwidth prediction using Hybrid Division Model. In 2021 IEEE International Conference on Consumer Electronics (ICCE) (pp. 1-4). IEEE.

[4] Chi, H., Chang, K., Lai, K., Yu, Y., & Yu, Y. (2023). Bandwidth Factor Computation Framework for Traffic Routing in 5G Networks2. In 2023 23rd Asia-Pacific Network Operations and Management Symposium (APNOMS) (pp. 385-388). IEEE.

[5] Gao, Y., Zhou, C., Rong, J., Wang, Y., & Liu, S. (2022). Short-term traffic speed forecasting using a deep learning method based on multitemporal traffic flow volume4. IEEE Access, 10, 82384-82394.

[6] Genda, K. (2021). On-demand network bandwidth reservation combining machine learning and linear programming. 2021 17th International Conference on Network and Service Management (CNSM), 331-336.

[7] Guan, W., Zhang, H., & Leung, V. C. M. (2020). Analysis of traffic performance on network slicing using complex network theory6. IEEE Transactions on Vehicular Technology, 69(12), 15188-15200.

[8] Heng, Y., Chandrasekhar, V., & Andrews, J. G. (2021). UTMobileNetTraffic2021: A labeled public network traffic dataset. IEEE Networking Letters, 3(3), 156-160.

[9] Huoh, T. L., Luo, Y., Li, P., & Zhang, T. (2023). Flow-based encrypted network traffic classification with graph neural networks4. IEEE Transactions on Network and Service Management, 20(2), 1224-1238.

[10] Ma, X., Zheng, B., Jiang, G., & Liu, L. (2023). Cellular network traffic prediction based on correlation ConvLSTM and self-attention network23. IEEE Communications Letters, 27(7), 1909-1912.

[11] Maneesorn, T., & Putthividhya, W. (2020). Modeling and Forecasting of End-to-End Available Bandwidth in Wide Area Networks. 2018 IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA)62. IEEE.

[12] Sadeghzadeh, A. M., Shiravi, S., & Jalili, R. (2021). Adversarial network traffic: Towards evaluating the robustness of deep-learning-based network traffic classification45. IEEE Transactions on Network and Service Management, 18(2), 1962-1976.

[13] Sepasgozar, S. S., & Pierre, S. (2022). Fed-NTP: A federated learning algorithm for network traffic prediction in VANET3. IEEE Access, 10, 8227-8242.

[14] Tajiri, K., & Kawahara, R. (2023). Optimizing data distribution for federated learning under bandwidth constraint. In 2023 IEEE International Conference on Communications (ICC): SAC Cloud Computing, Networking and Storage Track (pp. 3732-3737). IEEE.

[15] Teker, A., Ornek, A. H., & Canberk, B. (2021). Network bandwidth usage forecast in content delivery networks5. In 2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD) (pp. 1-6). IEEE.

[16] Wang, L., Zhang, J., Zhang, Z., & Zhang, J. (2023). Analytic network traffic prediction based on user behavior modeling. IEEE Networking Letters, 5(4), 208-212.

[17] Wang, Y., Wang, F., Tian, Q., Sun, D., Hu, P., Wang, R., & Tang, X. (2023). Dynamic bandwidth allocation algorithm based on traffic classification with the aid of LSTM and GRU for industrial passive optical networks4. In 2023 21st International Conference on Optical Communications and Networks (ICOCN) (pp. 1-5). IEEE.

[18] Wu, L., Gan, C., Wang, X., Xu, Z., & Hui, J. (2020). Flexible Multi-Thread Dynamic Bandwidth Allocation Algorithm in VPONs Based on LR WDM/TDM PON2. IEEE Access, 8, 215467-215481.

[19] Yue, Y., Cheng, B., Liu, X., Li, B., Wang, M., & Chen, J. (2021). Resource Optimization and Delay Guarantee Virtual Network Function Placement for Mapping SFC Requests in Cloud Networks6. IEEE Transactions on Network and Service Management, 18(2), 1508-1522.

[20] Zhang, P., Jiang, C., Su, Y., Wang, J., Hsu, C. H., & Shen, S. (2022). Reinforcement learning assisted bandwidth aware virtual network resource allocation23. IEEE Transactions on Network and Service Management, 19(4), 4111-4125.

[21] Zhang, P., Kumar, N., Wang, C., Aujla, G. S., & Guizani, M. (2020). IoV scenario: Implementation of a bandwidth aware algorithm in wireless network communication mode. IEEE Transactions on Vehicular Technology, 69(12), 15774-15786.

[22] Pang, S., Zeng, D., & Chen, X. (2022). Research on SDN-based data center network traffic management and optimization1. In 2022 IEEE 2nd International Conference on Power, Electronics and Computer Applications (ICPECA) (pp. 600-604)2. IEEE.

[23] Ganowicz, A., Knapi´ nska, A., Starosta, B., & Walkowiak, K. (2022). Short-term network traffic prediction with multilayer perceptron1. In 2022 6th SLAAI International Conference on Artificial Intelligence (SLAAI-ICAI) (pp. 1-6). IEEE.

[24] Sridharan, A., Tripathi, A., Othman, B. A., Shukla, S. K., Pandey, A., & Tongkachok, K. (2022). The emerging role of integrating artificial intelligence (AI) in the software defined network for effective traffic prediction using correlation analysis. In 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE) (pp. 116-120). IEEE.

[25] Mostafaei, H., & Afridi, S. (2021). P4Flow: Monitoring traffic flows with programmable networks1. IEEE Communications Letters, 25(11), 3546-3550.