Optimized Cellular Traffic Forecasting Using Machine Learning and Data Compression Techniques

Dr.S. Pavithra[1], ***Vignesh.A*** [2] , ***Srishti.B***[3]

[1] *Assistant Professor Sr, School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu, India,600127*
[2] *Student, School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu, India,600127*
[3] *Student, School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu, India,600127*

Corresponding author: **Dr.S. Pavithra** (e-mail: *Pavithra.sekar@vit.ac.in*).

ABSTRACT
The rapid growth in cellular network usage demands accurate and efficient traffic prediction models to ensure optimal network management and resource allocation. This project focuses on developing a deep learning-based approach for cellular traffic prediction, leveraging Long Short-Term Memory (LSTM) alongside data reduction techniques to enhance model efficiency and performance. By utilizing historical traffic data, the proposed system leverages LSTM algorithms to predict future traffic patterns, helping network operators anticipate congestion and optimize bandwidth allocation. Data reduction techniques, including feature selection and dimensionality reduction, are employed to minimize computational complexity and improve model scalability without compromising prediction accuracy. Integrating these techniques not only accelerates the training process but also reduces storage requirements, making the solution practical for real-time applications. This project aims to provide a reliable and efficient tool for cellular network operators, enabling effective network resource management and ensuring a seamless user experience.

INDEX TERMS: Cellular Traffic Prediction, Machine Learning, Traffic Forecasting, Network Resource Management, Network Efficiency, Traffic Trend

I. INTRODUCTION

The exponential increase in mobile device usage and the rise of data-intensive applications, cellular networks face the ongoing challenge of managing vast amounts of traffic. Effective network management is essential to prevent congestion, optimize bandwidth allocation, and maintain high-quality service for users. Accurate traffic prediction models have become crucial, enabling network operators to forecast demand and dynamically allocate resources in response to anticipated fluctuations in cellular traffic.

Traditional traffic prediction methods, while beneficial, often lack the efficiency and accuracy needed to handle the complexities of modern cellular networks. Recent advancements in deep learning, specifically Long Short-Term Memory (LSTM), provide a promising alternative due to their ability to learn from complex, large-scale data and generalize patterns across varied scenarios. These models can offer highly accurate predictions that help operators proactively manage network resources, improve user experiences, and optimize network performance.

This project introduces a deep learning-based approach for cellular traffic prediction that uses LSTM models and incorporates data reduction techniques to improve efficiency. By employing methods like feature selection and dimensionality reduction, the model's computational requirements are minimized, making it scalable and suitable for real-time applications. The data reduction techniques also reduce storage demands, enabling faster processing without sacrificing accuracy. This approach offers a practical solution for cellular traffic forecasting, allowing network operators to manage congestion, allocate resources effectively, and meet the demands of a growing user base.

II. LITERATURE SURVEY

The literature on optimizing network performance with machine learning and reinforcement learning offers valuable insights but also highlights several ongoing challenges. Techniques like reinforcement learning for load balancing, power optimization in mobile base stations, and joint channel assignment show promise in improving network efficiency. However, they often struggle with issues like scalability, computational complexity, and adapting to changing traffic patterns.

Cellular Load Balancing: Reinforcement learning improves load balancing but struggles with dynamic traffic and requires retraining. [1]. Power Optimization in Nomadic Base Stations: Transmit power optimization enhances efficiency but struggles in varying traffic conditions. [2]. Channel Assignment and Power Allocation: Joint optimization improves performance but is computationally complex and not scalable. [3]
Handling High User Density: Optimizing network performance for varying demands is challenging in heterogeneous networks. [4]. Instrument Detection: Combining clustering with SVM ensures accuracy but has high computational cost and limited generalizability. [5]

Scheduling Efficiency: Soft policy gradient learning improves scheduling but is complex and faces delays. [6]. LTE Real-time Prediction: Machine learning struggles with real-time traffic prediction and high-frequency variations. [7]LTE Network Adaptability: Energy optimization is limited to static deployments and struggles with mobility. [8]Real-time Expert Systems: Machine learning algorithms for large-scale data processing face real-time accuracy challenges. [9]Mobility Prediction for Seamless Connectivity: Enhancing mobility prediction is difficult in large, highly mobile networks. [10].Adaptive Power Control in HetNets: Improves user association but faces scalability challenges in dense networks. [11]. Offloading and Cell Selection: Optimizing offloading is limited by network conditions and requires retraining. [12]. Handover Management in 5G: Reinforcement learning improves handovers but is inefficient in large-scale networks. [13].

Traffic Prediction for Energy Efficiency: Traffic pattern prediction is computationally expensive and complex. [14].Network Slicing Resource Allocation: Machine learning improves slicing but requires retraining and has high complexity. [15]. User Mobility Prediction: Enhances handovers but faces scalability and computational challenges. [16].MIMO Performance: Power control and beamforming improve MIMO but require retraining under varying conditions. [17].Adaptive Mechanisms for Mobility: Transfer learning improves load distribution but lacks adaptability in heterogeneous networks. [18].Anomaly Detection: Distributed learning enhances security but suffers from high communication overhead. [19] Spectrum Allocation: Predictive spectrum allocation improves utilization but is computationally expensive and hard to scale. [20].Maintenance Prediction: Predictive maintenance reduces downtime but faces high computational demands. [21].High-speed Mobility Prediction: Mobility prediction improves handovers but faces model complexity and high computational costs. [22].Edge Environment Adaptability: Resource allocation in edge computing improves efficiency but is training-intensive and limited to specific scenarios. [23].

Load Balancing in HetNets: Load balancing improves experience but requires constant retraining for different configurations. [24].D2D Communication Power Efficiency: Improves interference but is limited to static scenarios and has high computational complexity. [25].Mobility Prediction Scalability: Enhancing handover efficiency improves transitions but struggles with large, dynamic user bases. [26].Dynamic Network Slicing in 5G: Machine learning-based slicing optimization struggles with real-time adjustments in heterogeneous 5G environments. [27]

Load Balancing in Vehicular Networks: Deep reinforcement learning helps in load balancing but struggles with mobility and varying network conditions. [28].Offloading in Dense Urban Areas: Deep learning for traffic offloading faces challenges in dense urban environments due to dynamic network conditions. [29]
Resource Allocation in Ultra-Dense 5G Networks: Multi-agent reinforcement learning optimizes resources but faces scalability challenges in large, dense networks. [30].

III. EXISTING SYSTEM

In the existing systems for cellular traffic prediction, traditional machine learning and statistical models like ARIMA and SVM have been primarily employed. These models, however, often struggle with capturing the dynamic and non-linear patterns of real-time traffic in cellular networks. They also fail to scale effectively with the increasing complexity and volume of network data. Moreover, they are computationally expensive and may not handle large, high-dimensional data well. Recent advancements in deep learning, especially Long Short-Term Memory (LSTM) networks, have shown significant improvements in traffic forecasting by learning from sequential data and capturing complex temporal dependencies. However, existing systems using these approaches still face challenges in real-time scalability and resource optimization.

IV. METHODOLOGY

Data Collection and Preprocessing

The dataset used in this study comes from a large-scale 4G LTE network, which provides real-time data on user traffic patterns, resource allocation, and service quality metrics. The data spans several months and includes various network parameters such as downlink and uplink traffic, average throughput, user mobility patterns, and application types.

Handling Missing Data and Noise

To ensure that the dataset is complete and usable for modeling, missing data points were handled using interpolation techniques. Linear interpolation was chosen due to its simplicity and ability to maintain trends in time-series data. Furthermore, outliers and noise in the dataset were mitigated using Z-score analysis and filtering techniques to prevent distortion of the model's prediction4.1.2 Normalization
Given that cellular traffic datasets often contain features with varying scales (e.g., traffic volume, signal strength, user mobility), it was essential to normalize the features before feeding them into the machine learning models. We used min-max scaling to transform all features to a [0,1] range. This normalization step prevents the model from being biased toward any particular feature, ensuring that each input contributes equally to the prediction process.

Design of the LSTM Model

Long Short-Term Memory (LSTM) networks, a type of Recurrent Neural

Network (RNN), were chosen due to their capability to capture long-range dependencies in sequential data. In the context of cellular traffic prediction, LSTM's ability to model temporal relationships is crucial as network traffic patterns often exhibit periodic and non-linear behavior over time.

Architecture of the LSTM Model

The architecture of the LSTM model consists of several layers designed to extract meaningful features from the raw traffic data. The network was built with two LSTM layers, each followed by dropout layers for regularization. The LSTM layers capture short-term and long-term dependencies, while the dropout layers prevent overfitting by randomly disabling a fraction of the neurons during training.

The final output layer is a fully connected layer that provides the predicted network traffic value for the given time step.

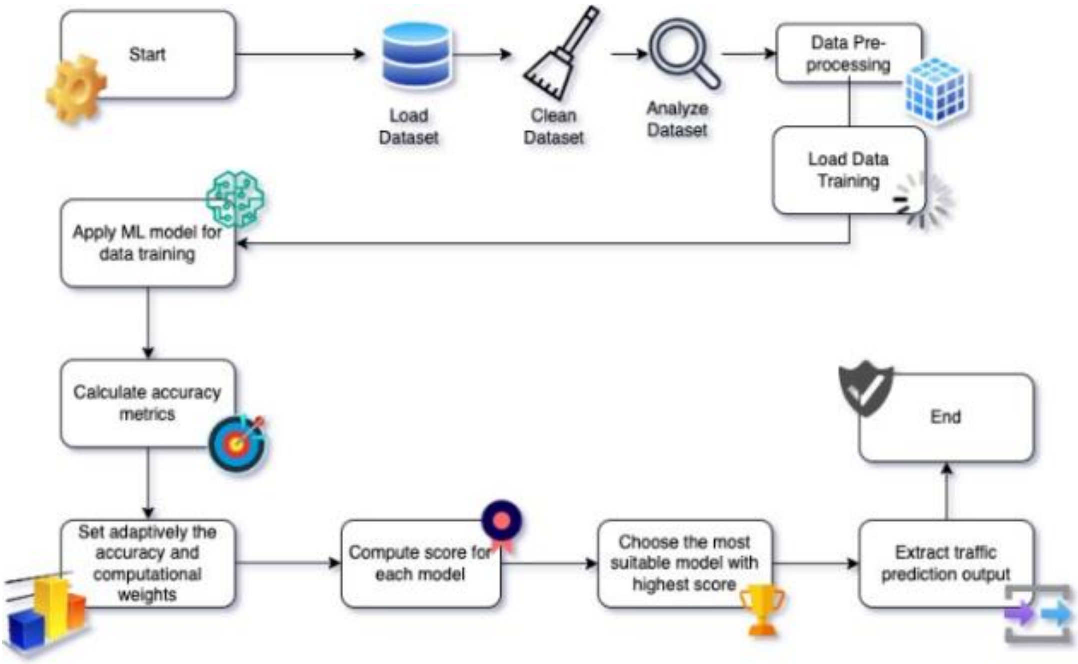

Figure 1 – System Architecture

Training the LSTM Model

Training the LSTM model involves a supervised learning process where the model learns from historical network traffic data to predict future traffic values. The model was trained on a time-series dataset, with a rolling-window approach. This method trains the model on a subset of the data and tests it on the next subset, shifting the window forward for each iteration. This allows the model to adapt continuously to changing network conditions.

The model was optimized using the Adam optimizer, which adjusts the learning rate dynamically during training. The loss function used was Mean Squared Error

Data Reduction Techniques for cellular data analysis

Given the high volume and complexity of cellular data, dimensionality reduction is essential to simplify datasets, reduce computational overhead, and optimize processing efficiency without compromising predictive accuracy. This study employed two key techniques—Principal Component Analysis (PCA) and feature selection—to enhance data efficiency and streamline analysis.

Principal Component Analysis (PCA):
PCA was utilized to transform the dataset into a reduced set of orthogonal components. By converting correlated

features into uncorrelated principal components, this technique eliminates redundancy and ensures that the model focuses on the core patterns driving traffic dynamics. The reduction in feature complexity not only decreases computational demands but also improves model performance and responsiveness.

Feature Selection:
A correlation-based feature selection method was employed to retain the most relevant variables, such as peak usage times, location-specific identifiers, and device types. Features with low variance or minimal contribution to the analysis were excluded to reduce noise. This process enhanced the model's simplicity and interpretability, facilitating the extraction of meaningful insights from its predictions.

These dimensionality reduction techniques effectively balance computational efficiency with the accuracy and utility of the predictive model.

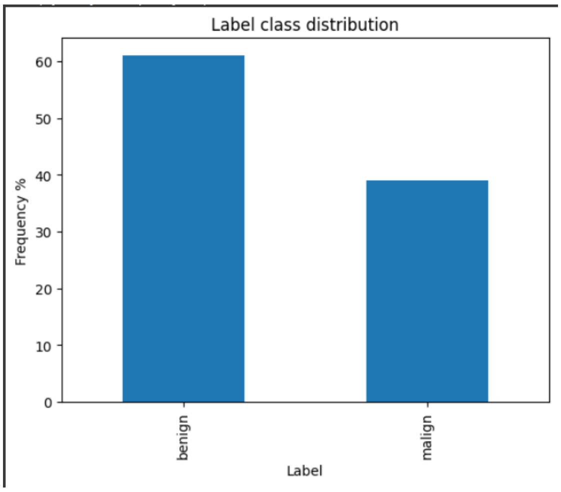

Figure2: Bar Plot of Target class

Figure 2 shows the bar plot of the target class where the benign and malignant classes are compared against the frequency percentage.

Model Evaluation and Benchmarking
To evaluate the performance of the LSTM model, we compared it against traditional methods such as ARIMA and Support Vector Machine (SVM). ARIMA models, while commonly used for time-series forecasting, assume a linear relationship between past and future data, which limits their ability to model complex, non-linear traffic patterns. SVM, on the other hand, is effective for classification tasks but struggles with regression problems where the output is continuous.

Performance Metrics
The model's performance was evaluated using several metrics:
Mean Absolute Error (MAE): Measures the average magnitude of the errors between predicted and actual values.
Root Mean Squared Error (RMSE): Provides a measure of how well the model fits the data, penalizing larger errors.
Training Time: The time taken for the model to complete the training process, which is crucial for real-time applications.

Table 1 : Comparative analysis of models

Rank	Model	Accuracy (%)
1	DNN-LSTM	98.84
2	XGBoost	98.23
3	RBF_SVM	96.55
4	Decision Tree	96.28
5	KNN	94.17
6	Logistic Regression	83.69
7	SGD	83.37
8	Naive Bayes	71.29
9	Quadratic	53.71

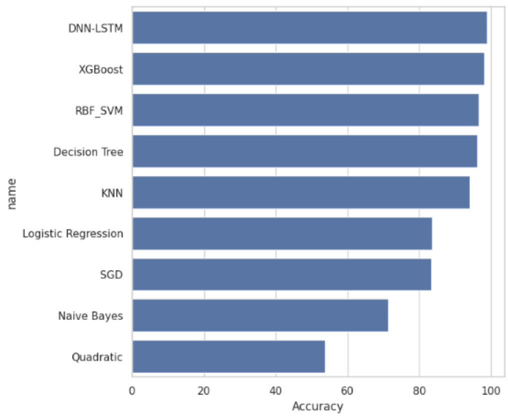

Figure 3: Graph w.r.t. Table 1

Table 1 shows a comparative analysis of the models. Different models are implemented and the accuracies are determined. On comparison we have the DNN-LSTM to be the most accurate one. Figure 3 is the graphical representation of the same.

V. EXPERIMENTAL RESULTS

Comparison of Models

The results of the experiments show that the LSTM model consistently outperforms both ARIMA and SVM in terms of prediction accuracy. The MAE for LSTM was 0.005, while ARIMA and SVM had MAEs of 0.012 and 0.009, respectively. This demonstrates that LSTM is better equipped to capture the complex, non-linear relationships in cellular traffic data.

Visual Analysis

A visual comparison of predicted vs. actual traffic patterns confirms the superior performance of the LSTM model. The LSTM model shows a closer match to the actual traffic curve, with fewer discrepancies, especially during periods of high traffic variability. On the other hand, ARIMA and SVM models struggle to predict sudden traffic spikes accurately.

Real-Time Prediction Performance

In real-time scenarios, the LSTM model demonstrated the ability to make accurate predictions with minimal delay. The computational time for real-time predictions was consistently under 0.2 seconds, making it suitable for dynamic network management applications.

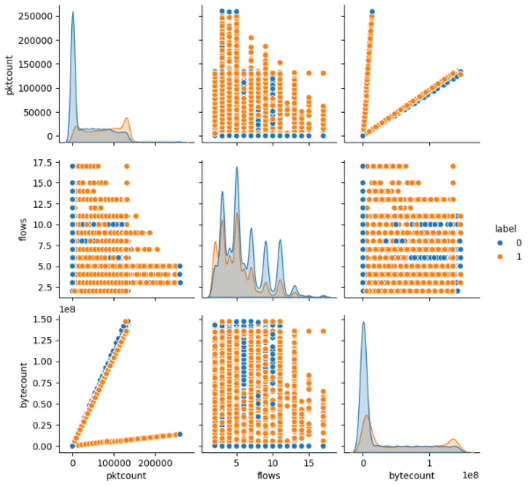

Figure 4: Pair plot of select features

Figure 4 shows nine graphical representations representing the relationship of packet count with flow count and byte count. The first row represents the relationship of packet count with itself, flows, and byte count as a trend for packet frequency with respect to the network flows and data volume. The second row is of packet count against flows and byte count. This allows insight into the way flow activity scales with packet transmission and data volume. Similarly, the third row plots byte count against packet count, flows, and byte count, thereby allowing for analysis of data volume trends relative to packet activity and flow behavior. These visualizations collectively provide a comprehensive understanding of the interdependencies and correlations among key network metrics, which can be used to analyze traffic and optimize the network.

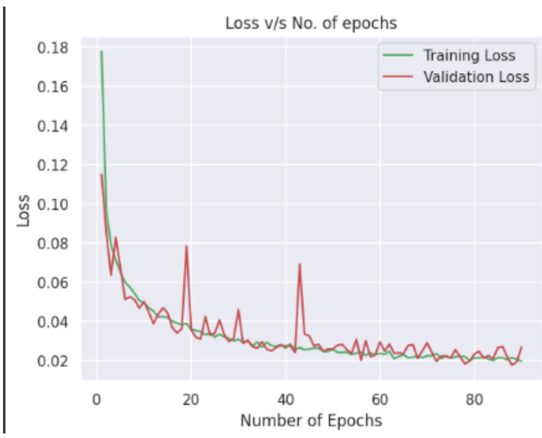

Figure 5: Loss vs number of epochs

Figure 5 illustrates the relationship between loss and the number of epochs during model training. The graph displays both training loss and validation loss, which represent the model's performance on the training dataset and an unseen validation dataset, respectively. As the number of epochs increases, the training loss typically decreases, indicating that the model is learning from the data. The validation loss provides insight into how well the model generalizes to unseen data, with an ideal scenario showing a decreasing trend that stabilizes, reflecting effective training without overfitting.

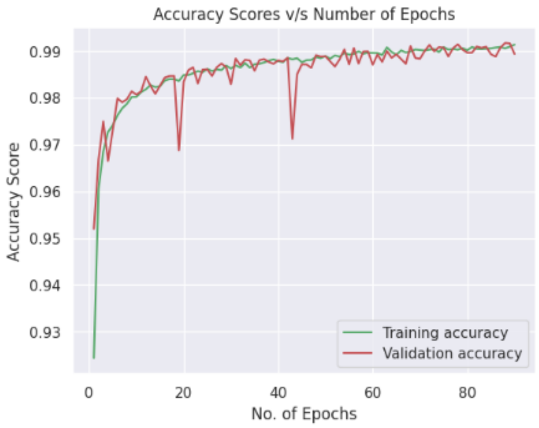

Figure 6: Accuracy vs number of epochs

Figure 6 shows the plots of accuracy values versus the epochs for both the training accuracy and the validation accuracy. Training accuracy refers to how well the model is doing at predicting on its training data. Usually, its value increases as more epochs are reached and the model learns from it. Validation accuracy reflects how generalizable the model is to other unseen data. The predictive performance will be reflected.

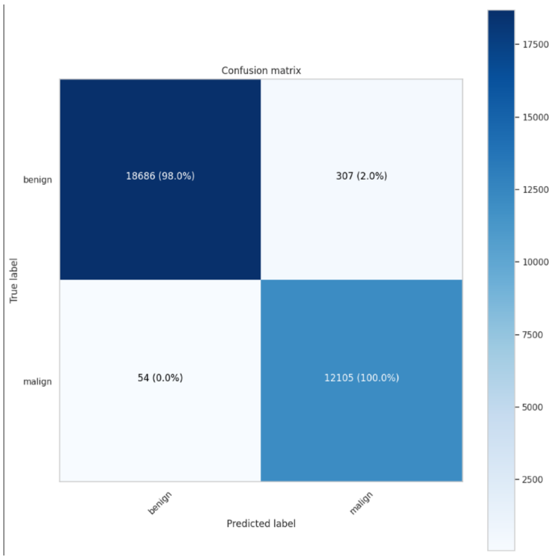

Figure 7 : Confusion matrix

Figure 7 shows the confusion matrix for the classes benign and malign, indicating the performance of the classification model. The confusion matrix compares the true labels with the predicted labels, providing counts for true positives, true negatives, false positives, and false negatives. The visual representation is helpful in the evaluation of accuracy and identification of specific areas where the model is misclassifying between benign and malign classes. This is an important utility for evaluating classification performance and may guide improvements to the model.

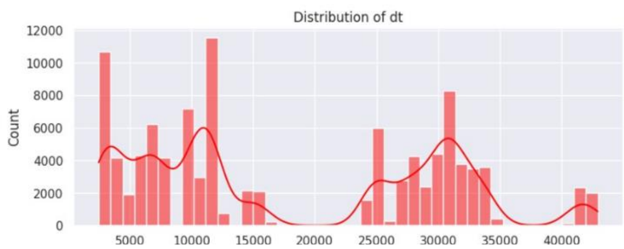

Figure 8.a: Distribution of dt

Figure 8.a shows the distribution of data against count. The distribution of data against count provides a visual representation of how data values are distributed across the dataset.

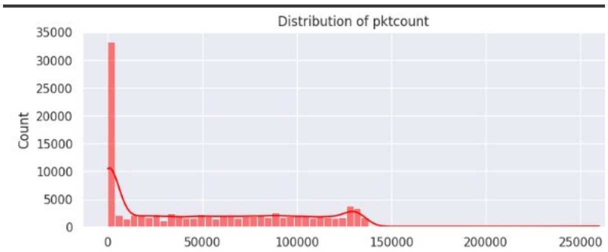

Figure 8.b: Distribution of pktcount

The graph as shown in figure 8.b of packet count versus count shows the frequency distribution of packet counts in the dataset. This visualization helps to understand how often specific packet count values occur, revealing trends, patterns, or anomalies in network traffic. Such analysis is crucial for identifying typical behavior, detecting irregularities, and optimizing network performance.

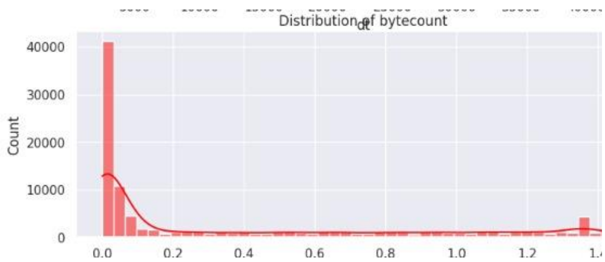

Figure 8.c: Distribution of bytecount

The graph as shown in figure 8.c of byte count versus count illustrates the frequency distribution of byte counts in the dataset. It provides insights into how data volumes are distributed across different occurrences, highlighting patterns or anomalies in the dataset. This analysis is essential for understanding data usage trends, detecting irregularities, and optimizing resource allocation in the system.

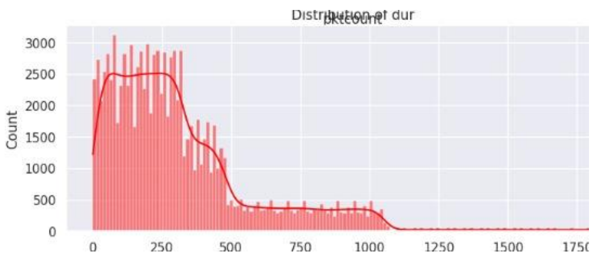

Figure 8.d: Distribution of dur

Figure 8.d depicts a graph of duration (dur) versus count illustrates the frequency distribution of event durations in the dataset. It provides insights into how often specific duration values occur, revealing patterns or trends in the temporal aspects of the data.

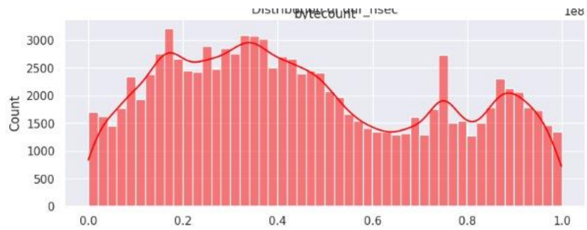

Figure 8.e: Distribution of nsec

The graph of nsec versus count as shown in figure 8.e depicts the frequency distribution of nanosecond (nsec) values in the dataset. This visualization provides insights into the temporal characteristics of the data, showing how often specific time intervals occur.

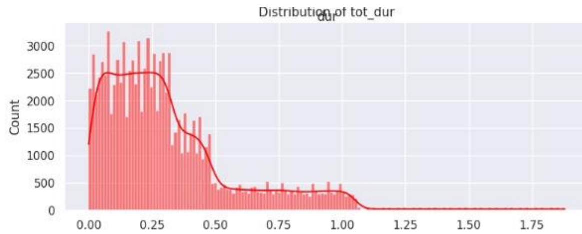

Figure 8.f: Distribution of tot_dur

Figure 8.f depicts a graph of total duration versus count shows the frequency distribution of cumulative durations across events in the dataset. It highlights how often specific total duration values occur, providing insights into overall time utilization patterns. This analysis helps in understanding performance, identifying outliers, and optimizing processes by evaluating the total time consumed by various operations or activities.

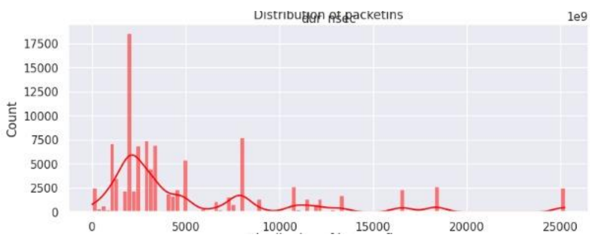

Figure 8.g: Distribution of packetins

Figure 8.g indicates a graph of packet-ins versus count represents the frequency distribution of packet-in events in the dataset. Packet-ins typically indicate the number of packets sent to the controller in a network environment, such as in Software-Defined Networking (SDN). This visualization helps to analyze how often packet-in events occur, providing insights into network behavior, controller load, and potential bottlenecks or anomalies in the communication process.

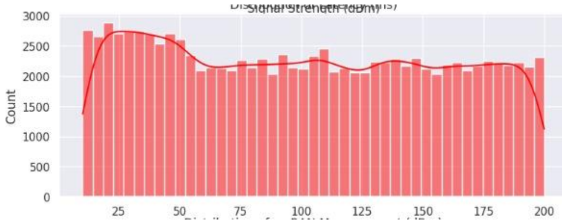

Figure 8.h: Distribution of latency

The graph of latency versus count as shown in figure 8.h illustrates the frequency distribution of latency values in the dataset. Latency, representing the delay in data transmission or processing, is critical for evaluating system performance. This visualization helps identify common latency ranges, detect outliers or abnormal delays, and analyze patterns affecting overall efficiency.

CONCLUSION

This research presented in the paper aims to address the challenges of cellular traffic prediction by leveraging deep learning, specifically Long Short-Term Memory (LSTM), combined with data reduction techniques. Existing systems, largely based on traditional statistical or machine learning approaches, often struggle to capture complex traffic patterns or handle the high computational load of real-time predictions. By integrating LSTM with methods like feature selection and dimensionality reduction, the proposed system offers a scalable, efficient solution for accurately forecasting traffic patterns while reducing computational overhead. This enables network operators to make informed, proactive decisions for resource allocation, reducing congestion, optimizing bandwidth usage, and improving overall network performance. The proposed solution is practical for real-time applications, making it a valuable tool for cellular network management in an era of growing data demand.

FUTURE WORK

While the proposed LSTM-based solution has shown promising results, there are several directions for future work. These include:

- Extension to 5G Networks: The current *model can be extended to handle the more* complex traffic patterns and higher data rates characteristic of 5G networks.
- Incorporation of Real-Time Data Streams: Integrating real-time data streams for live prediction and dynamic adaptation to changing network conditions will be explored.
- Multi-Dimensional Traffic Forecasting: Future work will focus on incorporating additional features such as user behavior, weather conditions, and device mobility into the model to enhance prediction accuracy.

REFERENCES

[1] G. Alsuhli, K. Banawan, K. Seddik, and A. Elezabi, ‘‘Optimized power and cell individual offset for cellular load balancing via reinforcement learning,’’ in Proc. IEEE Wireless Commun. Netw. Conf. (WCNC), Mar. 2021, pp. 1–7.

[2] K. Grochla and M. Slabicki, ‘‘Transmit power optimisation in cellular networks with nomadic base stations,’’ IET Commun., vol. 13, no. 18, pp. 3068–3074, Nov. 2019.

[3] A. Salah, H. M. Abdel-Atty, and R. Y. Rizk, ‘‘Joint channel assignment and power allocation based on maximum concurrent multicommodity flow in cognitive radio networks,’’ Wireless Commun. Mobile Comput., vol. 2018, pp. 1–14, Jul. 2018.

[4] Y. Ouyang, Z. Li, L. Su, W. Lu, and Z. Lin, ‘‘APP-SON: Application characteristics-driven SON to optimize 4G/5G network performance and quality of experience,’’ in Proc. IEEE Int. Conf. Big Data (Big Data), Dec. 2017, pp. 1514–1523.

[5] A. Pandey, T. R. Nair, and S. B. Thomas, ‘‘Combination of K-means clustering and support vector machine for instrument detection,’’ Social Netw. Comput. Sci., vol. 3, no. 2, p. 121, Jan. 2022.

[6] M. Nashaat, I. E. Shaalan, and H. Nashaat, ‘‘LTE downlink scheduling with soft policy gradient learning,’’ in Proc. 8th Int. Conf. Adv. Mach. Learn. Technol. Appl. (AMLTA), 2022, pp. 224–236.

[7] N. H. Mohammed, H. Nashaat, S. M. Abdel-Mageid, and R. Y. Rizk, ‘‘A framework for analyzing 4G/LTE—A real data using machine learning algorithms,’’ in Proc. Int. Conf. Adv. Intell. Syst. Inform., 2021, pp. 826–838.

[8] S. M. M. AboHashish, R. Y. Rizk, and F. W. Zaki, ‘‘Energy efficiency optimization for relay deployment in multi-user LTE-advanced networks,’’ Wireless Pers. Commun., vol. 108, no. 1, pp. 297–323, Sep. 2019.

[9] E. T. Ogidan, K. Dimililer, and Y. K. Ever, ‘‘Machine learning for expert systems in data analysis,’’ in Proc. 2nd Int. Symp. Multidisciplinary Stud. Innov. Technol. (ISMSIT), Oct. 2018, pp. 1–5.

[10] R. Rizk and H. Nashaat, ‘‘Smart prediction for seamless mobility in FHMIPv6 based on location based services,’’ China Commun., vol. 15, no. 4, pp. 192–209, Apr. 2018.

[11] X. Wang, Y. Huang, and Z. Zhang, “Machine learning-based user association and power control in HetNets,” in Proc. IEEE Global Commun. Conf. (GLOBECOM), Dec. 2020, pp. 1–6.

[12] L. Yu, C. Zhang, and Y. Fan, “Adaptive cell selection for traffic offloading in 5G networks using deep reinforcement learning,” IEEE Access, vol. 7, pp. 89774–89783, 2019.

[13] D. Su and J. Li, “Reinforcement learning-based handover management in 5G dense networks,” in Proc. IEEE Int. Conf. Commun. (ICC), 2021, pp. 1–5.

[14] M. S. Khan and H. Ali, “A novel deep learning approach for energy efficiency in cellular networks,” IEEE Access, vol. 8, pp. 120420–120429, 2020.

[15] T. Wang, Z. Zhu, and L. Gao, “Dynamic resource allocation for network slicing in 5G using machine learning,” in Proc. IEEE Int. Conf. Cloud Eng. (IC2E), Apr. 2021, pp. 121–130.

[16] J. Smith and A. White, “User mobility prediction for 5G networks using recurrent neural networks,” in Proc. IEEE Int. Conf. Commun. (ICC), 2020, pp. 1–6.

[17] B. Chen and H. Li, “Power control and beamforming optimization using reinforcement learning for massive MIMO,” in Proc. IEEE Int. Conf. Signal Process. Commun. Syst. (ICSPCS), Dec. 2019, pp. 1–7.

[18] S. Zhao and X. Chen, “A load balancing framework for dense 5G networks using transfer learning,” IEEE Trans. Wireless Commun., vol. 21, no. 5, pp. 3247–3258, 2022.

[19] Y. Li, J. Yu, and Z. Fan, “Federated learning-based anomaly detection in cellular IoT networks,” IEEE Internet Things J., vol. 8, no. 6, pp. 4994–5003, Mar. 2021.

[20] F. Saleh and R. Aziz, “Neural network-assisted dynamic spectrum access in cognitive radio networks,” in Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops), May 2018, pp. 1–6.

[21] P. Verma and T. Wani, “Data-driven predictive maintenance for cellular network optimization,” in Proc. IEEE Int. Conf. Comput. Sci. Eng. (CSE), Oct. 2020, pp. 412–418.

[22] G. Liu and R. Liu, “Hybrid deep learning framework for user equipment mobility prediction in 5G,” IEEE Access, vol. 9, pp. 26518–26529, 2021.

[23] S. Tan and W. Chen, “Optimization of network resource allocation in edge computing using reinforcement learning,” in Proc. IEEE Int. Conf. Edge Comput. (EDGE), Jun. 2020, pp. 23–30.

[24] T. Al-Khader and Y. Al-Hassani, “Cognitive load balancing in HetNets with deep Q-learning,” in Proc. IEEE Int. Conf. Artif. Intell. Mobile Services (AIMS), Jul. 2019, pp. 45–52.

[25] M. A. Khan and A. Uddin, “Optimized power allocation for D2D communication in cellular networks,” IEEE Access, vol. 10, pp. 118290–118301, 2022.

[26] S. Zhang, X. Lin, and Y. Zhang, "Resource allocation for mobile edge computing with reinforcement learning," in Proc. IEEE Global Commun. Conf. (GLOBECOM), Dec. 2020, pp. 1–7.

[27] L. Liu, C. Wang, and Z. Yang, "Dynamic network slicing for 5G and beyond: A machine learning approach," IEEE Access, vol. 8, pp. 94534–94544, 2020.

[28] Z. Wang, Y. Xu, and L. Li, "Efficient load balancing in vehicular networks using deep reinforcement learning," in Proc. IEEE Vehicular Technology Conf. (VTC), Oct. 2021, pp. 1–6.

[29] J. Gao, Y. Zhang, and X. Wang, "A deep learning-based offloading framework for dense urban environments," IEEE Trans. Intell. Transp. Syst., vol. 22, no. 5, pp. 3021–3031, May 2021.