Educational Administration: Theory and Practice
2024,30(5), 1277-1284
ISSN:2148-2403
https://kuey.net/

Research Article

Forecasting Wireless Traffic In Mesh Networks Using Deep Metric Algorithms

P.Suganya[1*], I.Varalakshmi[2], A.Gugan[3], J.Gunasekaran[4], R.YuthayaKarthick[5]

[1,2]AssistantProfessor,Department of Computer Science and Engineering, Manakula Vinayagar Institute of Technology, Puducherry, India.
[3,4,5] B. Tech, Department of Computer Science and Engineering, Manakula Vinayagar Institute of Technology, Puducherry, India.
[1*]suganyamit23@@gmail.com, [2]gugang262@gmail.com,

Citation: P.Suganya et.al (2024), Forecasting Wireless Traffic In Mesh Networks Using Deep Metric Algorithms, *Educational Administration: Theory And Practice*, *30*(5), 1277-1284 Doi: 10.53555/kuey.v30i5.3079

ARTICLE INFO

ABSTRACT

The proposed system introduces a novel Deep Metric Algorithm as a cornerstone for enhancing the feasibility and optimal of network traffic prediction within the dynamic landscape of cloud technologies. The Deep Metric Algorithm is designed to operate as a physical interpretation probabilistic model that captures network-wide traffic patterns. Unlike traditional methods that rely solely on current traffic states observed at specific intervals, this algorithm leverages an initially expensive set of measurements to construct a network-specific traffic model. The algorithm's innovation lies in its ability to utilize this network-specific model alongside readily available, less expensive measurements, thereby significantly improving the accuracy of traffic predictions. By employing the Deep Metric Algorithm, the system can predict traffic fluctuations over observed links but also extend its predictions to unobserved links. This is a crucial advancement as it enables the system to offer more comprehensive insights into network behavior, even where direct measurements might be limited. One notable strength of the proposed Deep Metric Algorithm is its demonstrated applicability across various traffic periods within the same network. This adaptability ensures that the learned model remains effective over time, showcasing its potential utility in optimizing network performance as it evolves. The algorithm's predictive capabilities, coupled with its ability to adapt to changing network conditions, position it as a valuable tool for load-aware resource management and predictive control, thereby contributing to the efficient management of multi-provider networks in the evolving cloud landscape.

Keywords: Quality of Service(QoS), network traffic, accurate predictions, network-specific traffic model, Deep Metric Algorithm.

1 INTRODUCTION

The current landscape of wireless traffic prediction in communication networks is predominantly centered around a centralized learning strategy. This approach involves the transfer of substantial amounts of raw data to a centralized data center, where a generalized prediction model is developed. However, this method has inherent challenges, particularly the regular transfer of training data and overhead which can strain network capacity and adversely affect payload transmissions. To address these issues, there is a growing need for innovative wireless traffic prediction approaches that can navigate these challenges more effectively [1]. Wireless traffic prediction has gained significant attention due to its critical role in various tasks within wireless accurate traffic modeling and forecast capabilities are required for communications. By considering wireless traffic forecast as a time series prediction problem, the main difficulty is avoided. There are three main categories in which current approaches to this issue might be placed: parametric approaches and straightforward methods and non-parametric methods. In the context of computer networks, comprising nodes connected by physical links, data is transmitted from a source node to a destination node over predetermined path routes. The data stream between a specific source- destination pair is referred to as a "flow." One connection could be used for this flow-level traffic or multiple links, depending on the connectivity between source and destination nodes. The aggregate data traversing [2] each link, known as link- level traffic, is also of interest. The literature extensively studies both flow-level and link-level traffic.

While flow-level data provides direct information about individual flows, it is costly to obtain and process. Link-level data, on the other hand, is less expensive to obtain but provides less understanding of the underlying flows [3]. Network tomography has been instrumental in studying link-level data. The surge in network traffic, particularly due to real-time applications, poses a significant challenge to the management systems and TCP/IP network infrastructures that are in place now To proactively address this issue, management systems leverage traffic predictions to anticipate changes and handle the increasing network traffic efficiently. This proactive approach is crucial for maintaining the stability and performance of network infrastructures in the face of evolving traffic patterns.

Data collection: The data used in this study was sourced from Kaggle[4], a renowned open-source platform that hosts a vast array of datasets contributed by the global data science community. Kaggle provides a collaborative environment where researchers, data scientists, and enthusiasts can access, share, and analyze datasets for various purposes, including machine learning and predictive modeling. By utilizing Kaggle's open-source repository, the researchers gained access to relevant data for their investigation into network traffic prediction at the user level. In [5], Leveraging Kaggle's diverse and comprehensive datasets likely played a pivotal role in conducting a robust evaluation of their proposed meta-learning scheme, ensuring that their findings are applicable and well-supported by real-world data. The use of Kaggle's open-source data highlights the collaborative and accessible nature of the platform in supporting data-driven researching diverse domains, including network traffic analysis.

2 FEATURE EXTRACTION

Feature extraction is a crucial step in data pre-processing that involves transforming raw data

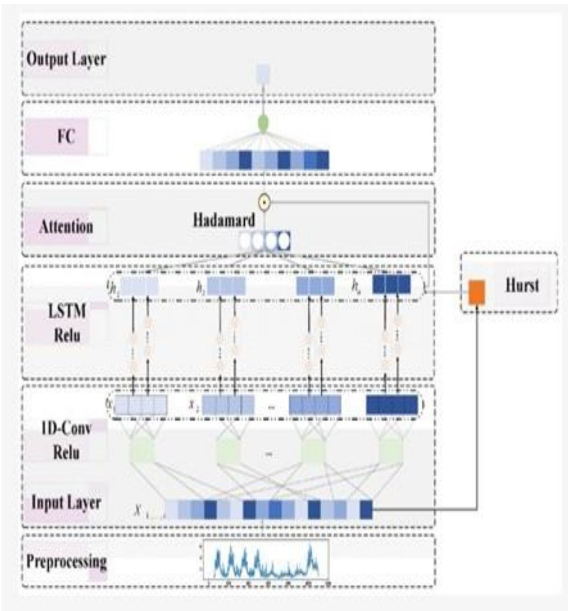

Fig 1 Diagram for Extracting features

into a set of meaningful and representative features shown in Fig.1.In various fields such as machine learning, signal processing, and image recognition, this process aims to highlight relevant information and reduce dimensional while retaining essential characteristics of the data. By selecting or creating features that best capture the patterns and relationships within the dataset, feature extraction enhances the efficiency and effectiveness of subsequent analysis or modeling tasks[6]. Common techniques for feature extraction include statistical methods[14], dimensional reduction algorithms, and domain-specific methods tailored to the nature of the data. The ultimate goal is to extract a concise and informative set of features that can contribute significantly to the accuracy and interoperability of the subsequent analytical or predictive models.

3 MODEL CREATION

The process of model creation using the Deep Metric Algorithm involves leveraging this algorithm's capabilities to understand and capture intricate network-wide traffic patterns. The Deep Metric Algorithm is a probabilistic model designed for its interoperability and efficacy in modeling temporal

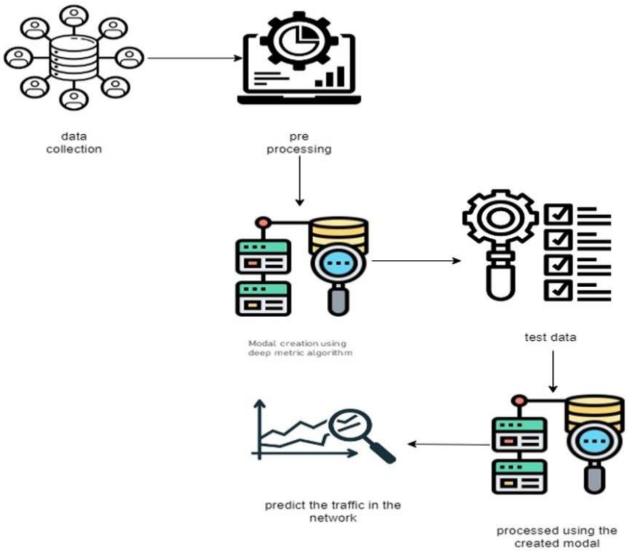

Fig 2. Model Creation using Deep Metric Algorithms

dependencies within network traffic. In the context of creating a predictive model, the algorithm is applied to extract relevant features from the dataset [7], focusing on traffic flow differences in Fig 2.

Unlike traditional methods, the Deep Metric Algorithm offers a physically interpretative framework that enhances the understanding of the network's behavior. The model creation involves training the algorithm on historical traffic data, enabling it to learn the dynamic relationships and patterns within the traffic flow. The resulting model is adaptive and excels in capturing short-term variations in traffic, making it suitable for dynamic rolling predictions. By incorporating the Deep Metric Algorithm into the model creation process, the approach aims to improve the precision of short-term network traffic predictions by extracting and utilizing features that effectively represent the temporal dynamics inhere in traffic patterns. The process of creating a predictive model using the Deep Metric Algorithm involves harnessing the capabilities of this algorithm to comprehend and capture intricate network-wide traffic patterns [8].

The Deep Metric Algorithm, in this context, acts as the core of the model, providing a probabilistic and physically interpretable framework. Its uniqueness lies in its capacity to offer a comprehensive understanding of network behavior, making it a well-suited form of modeling temporal dependencies within network traffic. During the model creation, the algorithm is applied to extract relevant features from the dataset, emphasizing traffic flow differences as input parameters.

Unlike conventional methods, the Deep Metric Algorithm enhances the interoperability of the model, allowing for a clearer understanding of the dynamics influencing network traffic[9]. The model, once trained on historical traffic data, becomes adept at discerning dynamic relationships and patterns within the traffic flow, making it adaptive and effective for short-term predictions. By incorporating the Deep Metric Algorithm into the model creation process, this approach aims to enhance the precision of short-term network traffic predictions by utilizing features that effectively represent the temporal dynamics inherent in traffic patterns.

4 RESULTS AND DISCUSSIONS

INPUT LAYER

The input layer of a neural network serves as the initial stage where the model receives and processes input data. In this layer, each neuron corresponds to a specific feature or attribute of the input data. The primary purpose of the input layer is to pass the input values to the subsequent layers for further processing. Each input neuron is associated with a weight, denoted as w_{ij},where It represents the input feature index and **j** denotes the neuron index in the subsequent layer. The input values **Xi** are multiplied by their corresponding weights, and the weighted sum (**Zi**) is calculated for each neuron in the subsequent layer[10]. Additionally, a bias term (**bj**) may be added to the weighted sum to introduce an offset. Mathematically, for the neuron in the subsequent layer, the computation is expressed as.

$$Zj = \sum_{i=1}^{n} wij\ Xi + bj$$

Where X_i represents the i-th input parameter

The output of this layer, after passing through an activation function, becomes the input for the hidden layers, enabling the network to learn complex representations and patterns from the input data. The input layer essentially acts as the interface between the input data and the neural network's computational processes.

HIDDEN LAYER

The hidden layer in a neural network is an intermediary layer between the input and output layers, responsible for Capturing complex patterns and representations within the data. Neurons in the hidden layer receive inputs from the input layer, perform computations, and pass the results to the output layer or another hidden layer if the network is deeper. Each neuron in the hidden layer is associated with a set of weights and a bias term, similar to the input layer. The computation within a hidden layer involve staking a weighted sum of the inputs, adding a bias term, and then applying an activation function to introduce non-linearity[11].Mathematically, f or the **j**th neuron in the hidden layer

$$Zj = \sum_{i=1}^{n} wij\, Xi + bj$$

$$a_j = \text{Activation}\ (Z_j)$$

The purpose of the hidden layer is to learn and extract hierarchical attributes from the data input, allowing the network to understanding intricate relationships and representations that may not be apparent in the raw input. The number of neurons in the hidden layer and the choice of activation function are crucial hyper parameters[12], that influence the network's capacity to learn and generalize .The hidden layer contributes to the neural network's ability to model complex mappings and solve intricate to asks through its hierarchical feature extraction mechanism.

$$\text{Loss} = -\log P(Y_{true} \mid X_{input}, \theta)$$

OUTPUTLAYER

The output layer in a neural network is the final Layer responsible for producing the model's predictions outputs based on the processed information from the preceding layers. The number of neurons in the output layer typically corresponds to the number of distinct classes or dimensions in the output space, depending on the nature of the task (e.g., classification, regression)[13]. In classification task where the goal is to categorize input data into distinct classes, each neuron in the output layer typically represents a class, and the activation of a neuron indicates the probability or confidence that the input belongs to that class. Common activation functions used in the output layer for classification tasks include a function, which normalizes the output values into a probability distribution over the classes. In a regression task where the goal is to predict continuous values, such as prices or quantities, the output layer usually consists of a single neuron that directly produces the predicted value. The activation function in this case is typically chosen based on the nature of the problem, with linear or identity activation functions commonly used for regression tasks[15][22]. Mathematically, for a classification task with **K** classes, the output of the ***k***-th neuron in the output layer (**Y*k***) is computed as:

$$Y_k = \sigma(Zk)$$

Where **Zk** is the weighted sum plus bias for the ***k***-th neuron and σ is the activation function.

LOSS FUNCTION

The selection of an appropriate loss function is crucial in training neural networks, and for probabilistic models, the negative log-likelihood often serves as a common choice. In probabilistic modeling, the goal is to estimate the underlying probability distribution of the data.

The negative log-likelihood loss, where minimized during training, encourages the model to predict outputs that align closely with the actual probability distribution of the observed data.This loss function penalizes deviations

$$Y_k = \sum_{j=1}^{m} w'jk\, aj + b'k$$

between predicted probabilities and the true distribution, effectively capturing the model's uncertainty and adjusting its parameters to improve the alignment with the training data[16]. Utilizing negative log-likelihood as the loss function is particularly relevant when dealing with probabilistic models, as it aligns with the probabilistic interpretation of the model's predictions, fostering more accurate and calibrated uncertainty estimates.

REGULARIZATION

Regularization is a crucial technique in training neural networks to prevent over fitting and enhance model generalization. L1 and L2 regularization terms are commonly added to the loss function to achieve this regularization effect. L1 regularization involves adding the absolute values of the weights as a penalty term to the loss function, encouraging sparsely in the model. This helps to prioritize essential features and discard less important ones. On the other hand, L2 regularization adds these squared values of the weights to the loss, penalizing large weights and promoting smoother weight distributions [17]. The regularization terms control the trade-off between fitting the training data well and maintaining a simpler model. By penalizing overly complex models, regularization mitigates the risk of memorizing noise in the training data, making the model more robust to unseen examples. The choice between L1 and L2 regularization, or a combination of both (Elastic Net), depends on the specific characteristics of the problem and the desired properties of the learned model.

5. PERFORMANCE ANALYSIS

The GRU MODEL

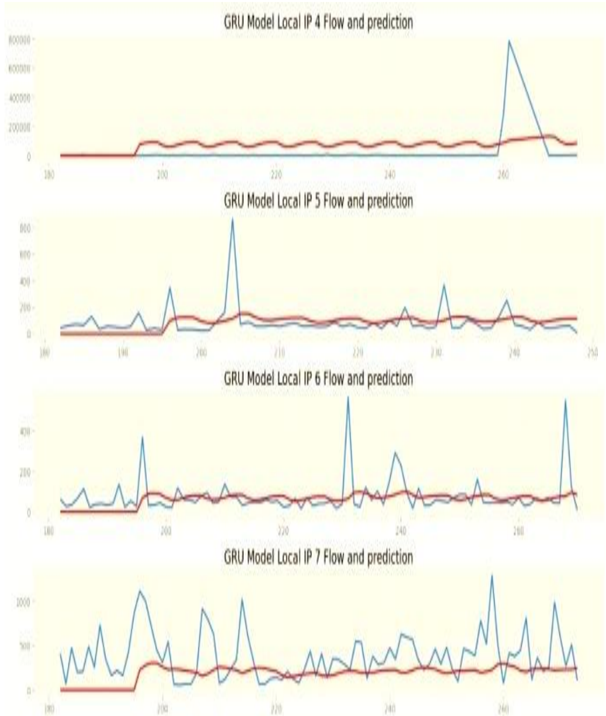

Fig 3. GRU Model

In fig 3, the GRU Model the graph represents the flow and prediction outputs with probability ratio of observed data.

LSTM MODEL

In fig 4, the LSTM Model the graph represents the flow and prediction outputs with of observed data with minimized training data set with actual probability ratio

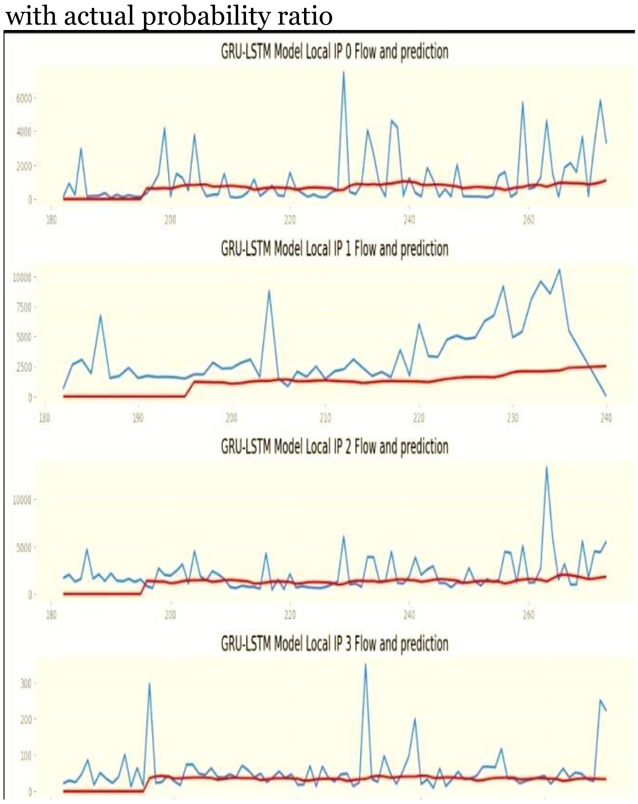

Fig.4 LSTM MODEL

GRU-LSTM MODEL

In GRU-LSTM makes a strong case for network traffic prediction, where the use of historical(time series data) data is collected over the wireless mesh network

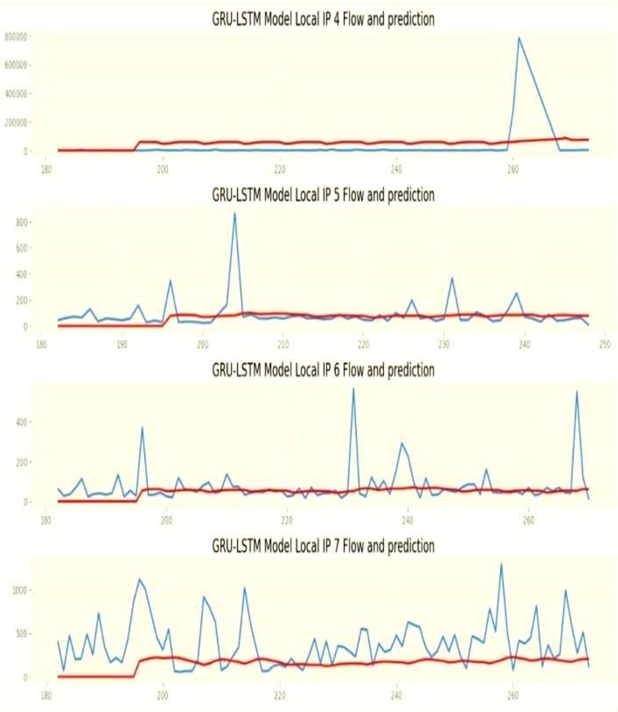

Fig 5(a), the GRU-LSTM Model

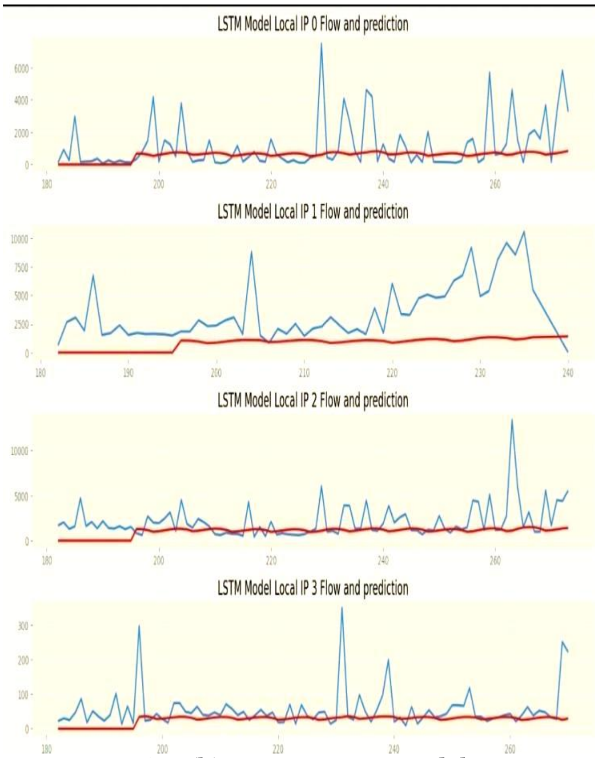

Fig 5(b)- GRU-LSTM Model

In Fig 5(a), the GRU-LSTM Model the graph is shown. In Fig 5(b), this graph shows the predicted values with more accuracy compared with flow and prediction outputs with exact probability ratio L1 and L2 are the regularization squared values, which predicts the data with a learned model.

CONCLUSION

This paper is proposed with Deep Metric Algorithm to represent a noteworthy innovation in the realm of network traffic prediction, offering a unique and interpretable probabilistic model. By leveraging the dynamic rolling prediction approach and introducing traffic flow differences as key input parameters, this algorithm addresses the challenges posed by the complex, non-linear, and uncertain nature of network traffic. The incorporation of the algorithm aims to enhance short-term prediction accuracy, providing valuable insights for network operators and service providers. The adaptability of the model to changing traffic conditions, coupled with its ability to capture network-wide traffic patterns, contributes to its effectiveness in real-world applications. Beyond theoretical advancements, the Deep Metric Algorithm holds practical significance for improving the efficiency of network management, aiding in congestion avoidance, and optimizing resource allocation where the time-series multivariate data set is similar to that of wireless mesh networks data set and gives a direction to the researchers to implement the proposed algorithm to predict the volume of network traffic. The Future work in the domain includes the application of some contemporary methods like the attention mechanism and transformers for traffic prediction with Multi modal networks that combine the physical configuration values and the network system log values and can automatically predict the incoming traffic and accordingly allocate the resources

REFERENCES

1. Sutskever, O. Vinyals, and V. Le, Sequence to sequence learning with neural networks, in Proc. Adv. Neural Inf. Process. Syst., 2014, p.3215seriesanalysisandsupervisedJun.2013.
2. C. Dong, Z. Xiong, C. Shao, and H. Zhang, Aspatial temporal-based state space approach forfreeway network traffic ow modeling and Aug.2015.
3. R.N.BracewellandR.N.Bracewell, TheFourier Transform and Its Applications. NewYork, NY,USA:McGraw-Hill,1986.
4. Z. Xiong, Y. Zhang, D. Niyato, R. Deng, P. Wang, and L. Wang, Deeprein forcementl earning for mobile 5 Gandbeyond:Fundamentals,applications-June.2019.
5. Jindal,G.S.Aujla,N.Kumar,R.Prodan,andM. S. Obaidat, DRUMS: Demand response management in a smart city using deep learning And efficient operations and management challenges of next generation network(NGN),GlobalJ.Comput.Sci.
6. Osseiran, F. Boccardi, V. Braun, K. Kusume, P.Marsch, M.Maternia, O.Queseth, M.Schellmann, H.Schotten, H.Taoka, and H.Tullberg, Scenarios for 5G mobile and wireless communications: The vision of May2014.
7. R.Vinayakumar,K.P.Soman,andP.Poornachandran,“Applyingdeeplearningapproaches for network traffic prediction,” in Proc. Int .Conf. Adv. Comput.,Commun.Informat.(ICACCI),Sep.2017,pp.2353–2358.
8. Gowri shankar and P. S. Satya narayana, ‘Neural network based traffic prediction for wireless data networks,” Int. J. Comput. Intell. Syst.,Dec.2008.
9. L. Xiang, X.-H. Ge, C. Liu, L. Shu, and C.-X.Wang, “A new hybrid network traffic prediction method,”in.IEEEGlobalTelecommun.Conf.,Dec.2010.
10. A. Y. Nikravesh, S. A. Ajila, C.-H. Lung, and W.Ding,“MobilenetworktrafficpredictionusingSVM,” in Proc. IEEE Int. Congr. Big Data, May2016.
11. S.Coxe, S.G. West, and L. S. Aiken, “The analysis of count data: A gentle introduction toPoissonregressionanditsalternatives,”J.PersonalityAssessment,vol.91,no.2,pp.121–136,Feb.2009.
12. L. Nie, D. Jiang, S. Yu, and H. Song, “Networktraffic prediction based on deep belief networkin wireless mesh backbone networks, inProc.IEEEWirelessCommun.Netw.Conf.(WCNC),Mar.2017,pp.1–5.
13. L. Nie, Z. Ning, M. S. Obaidat, B. Sadoun, H.Wang,S.Li,L.Guo,andG.Wang,“Areinforcementlearning-based network traffic prediction mechanism in intelligent Internet of Things,” IEEE Trans. Ind. Informat., vol. 17,no.3,pp.2169–2180,Mar.2021.
14. I.Varalakshmi, Mrs.D.Mohanapriya, “Detection of Abnormal Activities using Diffusion Convolutional-Recurrent Neural Network” in UGC approved “JASC: Journal of Applied Science and Computations” November/2019.
15. S.A.,Fung,C.,George,R.&Yazici,A.(2017,July).An efficient fuzzy path selection approach to mitigate selective forwarding attacks in wireless sensor networks.In2017 IEEE International Conference on Fuzzy Systems(FUZZ-IEEE)(pp.1-6).IEEE.
16. Haseeb,K.,Islam,N.,Almogren,A.&Din,I.U.(2019).Intrusion prevention framework for secure routing in WSN-based mobile Internet of Things.IEEEAccess,7,185496-185505
17. M. Amin, M. N. Akram, and M. Amanullah, “On the James-Stein estimator for the Poisson regression model,” in Proc. Commun. Statistics-Simulation Comput., pp. 1–13, 2020.
18. C. Katris and S. Daskalaki, “Comparing forecasting approaches for internet traffic,” Expert Syst. Appl., vol. 42, no. 21, pp. 8172–8183, 2015.
19. E. Elakkiya, M. Radha, and R. Sathy, “Application of Arima model forpredicting cashew nut production in India—An analysis,” Int. J. Res. Bus. Manage., vol. 5, pp. 45–52, 2017.
20. W. Lu, J. Li, Y. Li, A. Sun, and J. Wang, “A CNN-LSTM-based model to forecast stock prices,”

Complexity, vol. 2020, May 2020, Art. no. 6622927.
21. M. Alawad and M. Lin, ‘‘Stochastic-based deep convolutional networks with reconfigurable logic fabric,’’ IEEE Trans. Multi-Scale Comput. Syst.,vol. 2, no. 4, pp. 242–256, Oct. 2016.
22. S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman,‘‘1d convolutional neural networks and applications: A survey,’’ Mech. Syst. signal Process., vol. 151, May 2021, Art. no. 107398.
23. I.Varalakshmi, Dr.M.Thenmozhi, "Mitigation of DDoS attack using Machine Learning Algorithms in SDN_IoT environment" in Design Engineering (Toronto) October 2021.