Journal of Information Systems Engineering and Management
2025, 10(24s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

SDN Traffic Prediction using Empirical Mode Decomposition

Sarika Nyaramneni[1], Anjusha Pimpalshende[1], Anil Kumar Gujja[2]
[1]VNR Vignana Jyothi Institute of Engineering and Technology, Bachupally, Hyderabad, India
[2]Sr Network Manager, SP Sysnet Pte Ltd, Vertex, Singapore, Singapore

ARTICLE INFO

Received: 25 Dec 2024

Revised: 12 Feb 2025

Accepted: 24 Feb 2025

ABSTRACT

Internet traffic prediction is essential for effective network management, resource allocation, and ensuring efficient quality of service. Network resources can be dynamically managed by forecasting future traffic using past traffic patterns. Network traffic prediction enables the dynamic resource allocation to avoid the congestion and conflicts in the network. An Empirical mode decomposition (EMD) based machine learning models were proposed in this paper for the prediction of Software Defined Networks (SDN) traffic. SDN is a modern network architecture which separates the data plane from the control plane to provide centralized control over the network. EMD is an adaptive signal decomposition technique which extracts various frequencies from the collected network traffic at specified sampling intervals which are intrinsic mode frequencies (IMFs). Ensemble machine learning models such as Random Forest (RF), eXtreme Gradient Boosting (XGBoost), and Light Gradient Boosting Machine (LightGBM), were applied on intrinsic mode frequencies to generate the accurate predictions. SDN traffic traces were generated using CAIDA traffic traces and the experimental results suggests that EMD-XGBOOST outperforms than the other models with low root mean squared error (RMSE) and mean squared error (MSE).

Keywords: SDN(Software defined network) traffic prediction; Internet traffic; Empirical mode decomposition; Random Forest(RF); eXtreme Gradient Boosting(XGBoost).

1. Introduction

The process of forecasting future network behavior based on historical data is termed as network traffic prediction. Which plays a vital role in modern network resource allocation, management, and maintaining quality of service (QoS) across various types of networks such as local area networks (LANs), and Software Defined Networks (SDNs). By analyzing traffic patterns, network administrators can take the corrective measures to optimize the bandwidth utilization, congestion prevention, dynamic allocation, and deallocation of resources to improve the overall network performance and reliability.

In traditional networks, traffic prediction can be used for capacity planning, load balancing, and fault management [1]. With the rise of SDN architecture shown in figure 1., where the control plane is decoupled from the data plane and centralized control is provided to the network administrator, the traffic prediction becomes even more crucial [2]. In SDNs, accurate traffic forecasts enable more efficient decision-making processes for routing, switching, and other network functions, improving the overall performance and scalability of the network.

Software Defined Networks (SDN) represent a significant difference from traditional network architectures by separating control plane and the data plane. Control plane and data planes tightly integrated within the network devices (routers, switches, etc.), making network management more complex and inflexible in traditional networks. SDN, however, introduces centralized control, where the control plane responsible for managing and controlling the flow of every packet across the network [3]. It communicates with both the data plane and the application layer. It updates the flow tables of switches and provides instructions for routing packets [4]. If a switch encounters a packet that does not match any entries in its flow table, it directs the packet to the controller, which then updates the flow table with new rules. This allows the data plane to follow the actions or decision defined by the control plane which handles the actual data forwarding, to be managed more flexibly and dynamically.

Predicting SDN traffic is essentially valuable, as it enables more intelligent decision making. As the controller has a centralized overall view of the net- work, it can use traffic predictions to allocate resources dynamically, prevent congestion, and ensure high levels of quality of service (QoS). Accurate traffic prediction also helps in network planning, improving security by detecting anomalies, and managing bandwidth.

The remainder of the paper is organized as follows: Section 2 presents the literature survey, Section 3 explains the proposed methodology, Section 4 shows the experimental results, and conclusions discussed in Section 5.

2. Related Work

A dual-stage attention-based method for SDN traffic matrix prediction was proposed and authors ensured that the proposed model employed temporal attention for adaptive feature extraction and capturing long-term dependencies additionally the dynamic traffic volume handled by an autoregressive module [5]. [1] introduced a novel Spatial-Temporal Residual Graph Convolutional Network (STRGCN) model for SDN Traffic Matrix (TM) prediction. Authors of [6] proposed a network traffic prediction model using deep learning which captures dynamic variations in network traffic by using previous data to predict future traffic. Network Traffic Prediction (NTP) method for local area network using a linear feature-enhanced Convolutional Long Short-Term Memory (ConvLSTM) integrated with auto-regressive unit to enhance the linear prediction capability [7]. Advanced ensemble stacking models using random forest, xgboost and lstm were proposed to predict the SDN traffic in [8]. A Recurrent neural network (RNN) based LSTM model applied on GEANT network traffic matrix prediction [9]. Bidirectional Long Short-Term Memory (BiLSTM) and Gated Recurrent Unit (GRU) models utilized as traffic matrix (TM) prediction mechanisms within an SDN network architecture. These prediction models were evaluated on two datasets: the well-known GEANT backbone network traffic data and the dataset generated from a testbed environment [10]. [11] focuses on improving real-time traffic prediction in Software-Defined Networks (SDNs) by utilizing time-series based machine learning models, specifically XGBoost Regressor, LSTM, and Seasonal ARIMA, on live traffic data collected from SDN switches. [12] introduces a predictive traffic management method that utilizes a deep Recurrent Neural Network (RNN) model to forecast network state information, such as link latency and available bandwidth, before traffic flows reach the Software-Defined Wide Area Network (SD-WAN) infrastructure. The method aims to optimize Quality of Service (QoS) metrics, including end-to-end delay and bandwidth utilization, and demonstrates improved performance over traditional shortest-path algorithms. The integration of Machine Learning (ML) methods with Software Defined Networking (SDN) to develop new traffic engineering techniques aimed at dynamically managing and routing network traffic, which is essential for ensuring service quality and enhancing user experience [13]. [14] addresses the need for improved quality of service (QoS) in the context of increasing multimedia applications, which generate elephant flows that can lead to network congestion, packet loss, and delays. It emphasizes the importance of traffic prediction and the explanation of prediction models to enhance QoS delivery for Internet users. Authors of [15] discussed the challenges of network bandwidth management, emphasizing the need for traffic forecasting methods to predict future link usage, which is crucial for network operators to plan for increased traffic demands and optimize resource deployment. [20] use Artificial Intelligent such as SARIMA, multi-variable regression, ridge regression, and KNN regression for prediction.

The contributions of the proposed work:

- To generate and capture the SDN traffic using fat tree topology by replaying the traditional network traffic.
- Extract the different frequencies from the captured traffic patterns at different sample intervals using EMD.
- Predicting the SDN traffic and identifying the efficient combination of EMD and ensemble machine learning models.

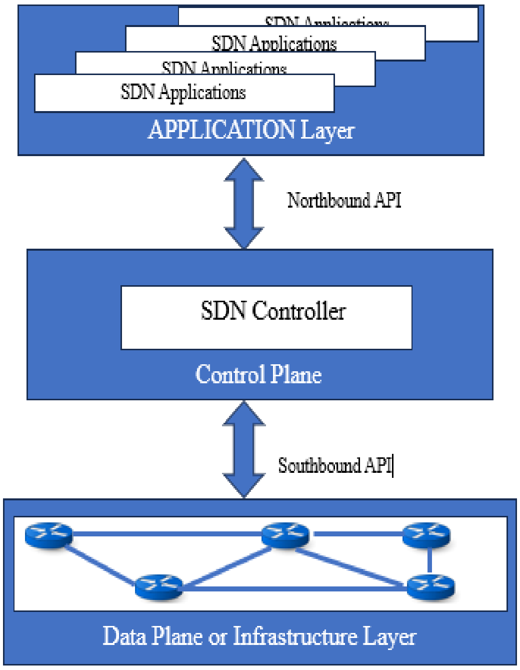

Figure 1: SDN Architecture

3. Proposed Methodology

An EMD based ensemble machine learning models were proposed for the prediction of SDN traffic. The proposed models were applied on the CAIDA traffic traces as well as the SDN traffic generated by the testbed. A test bed is created using the fat tree topology to replay the caida traffic traces to regenerate the SDN traffic traces as they were publicly not available. SDN traffic traces were capture at every node present in the fat tree topology.

3.1. Dataset

CAIDA's passive network traffic traces dataset contains pcap (packet capture files) collected from high-speed commercial backbone link. These traffic traces collected from the period starting from April 2008 to January 2019 and publicly available for the research on various characteristics such as application breakdown, security events, geographic and topological distribution, flow volume and duration, etc. [20].

3.2. Fat tree topology for network traffic replay

Fat tree topology is widely used in data centers due to its high bandwidth and fault tolerance [17]. Which consists of three different layers of switches core layer, aggregation layer, and edge layer. Core layer serves as the backbone of the fat-tree topology which provides high-speed interconnections across aggregation layers. Aggregation layer connects the switches of access layer to core layer switches, to allow traffic to flow between different groups of servers(pods) [18]. Every switch in a layer consists of k number of connections where, k represents the number of switches in core layer. 4 pod fat tree architecture used in this work to capture the SDN traffic traces. The test bed was built using mininet emulator. Pcap files collected from the caida were replayed in the fattree topology using tcpreplay command. The basic operation of tcpreplay is to resend all the packets from the source file with the same speed as they were recorded, or at a specified or modified data rate, up to as fast as the hardware is capable. Below command used to replay the caida traffic traces saved in different pcap files using tshrak command.

```
$tcprewrite --mtu-trunc --infile=input.pcap --outfile=output.pcap --
dstipmap=0.0.0.0/0:10.0.0.14 --enet-dmac=00:00:00:00:00:0e --
srcipmap=0.0.0.0/0:10.0.0.3 --enet-smac=00:00:00:00:00:03
```

Here, the tcprewrite command used to modify the headers of existing pcap files. Mtu-trunc option specifies the maximum transmission unit allowed on a link. The network traces in input.pcap file were modified and written to output.pcap file. After capturing the modified pcap files as mentioned below command1 adds a port to Open vSwitch (OVS) to capture the SDN traffic traces on that port as shown in command3. Where as command2 configures a mirror on an Open vSwitch (OVS) bridge to monitor traffic from specific ports and sends the mirrored traffic to a designated output port (s1001- eth5). Command3 captures the incoming traffic and writes it to output.pcap file for every 3600 seconds.

command1 $ovs-vsctl add-port s1001 s1001-eth5

command2 $ovs-vsctl -- set bridge s1001 mirrors=@m1 -- --id=@s1001-eth1 get Port s1001-eth1 -- --id=@s1001-eth2 get Port s1001-eth2 id=@s1001-eth3 get Port s1001-eth3 -- --id=@s1001-eth4 get Port s1001-eth4 id=@s1001-eth5 get Port s1001-eth5 -- --id=@m1 create Mirror name=mymirror1 select-dst-port=@s1001-eth1,@s1001-eth2,@s1001-eth3,@s1001-eth4 select-src-port=@s1001- eth1,@s1001-eth2,@s1001-eth3,@s1001-eth4 output-port=@s1001-eth5

command3 $tcpdump **–i** s1001-eth5 -G 3600 -w <output.pcap>

Traffic volume per every second extracted from the SDN traffic traces then the dataset considered as univariet time series data. Then emperical mode decomposition applied on the SDN traffic traces collected by the core layer switches named as s1001, s1002, s1003, and s1004. As shown in figure 2. EMD technique generates different intrinsic mode frequencies (IMFs)based on the volume of the traffic(in bytes). Machine learning models were applied on the IMFs to generate the predictions then these predictions were summed together to measure the errors and accuracy of the models.

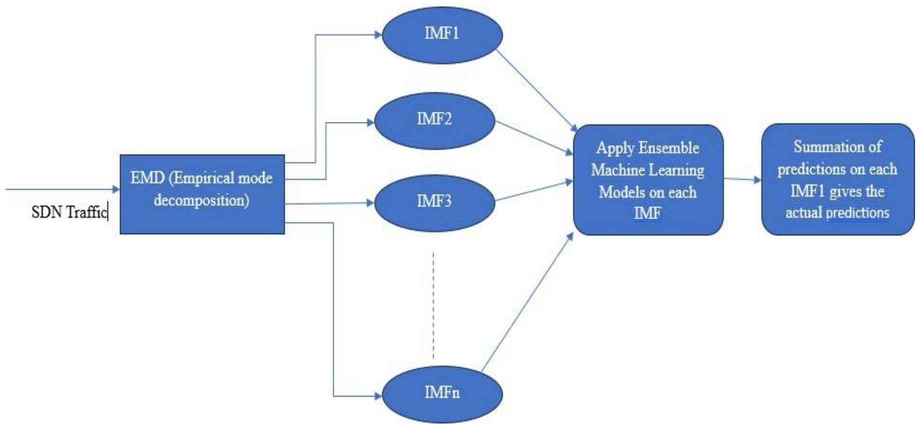

Figure 2: Proposed Model

3.3. Empirical Mode Decomposition

Empirical Mode Decomposition (EMD) is a signal processing technique used to decompose and analyze nonlinear and non-stationary signals into their constituent components, known as Intrinsic Mode Functions (IMFs). This approach, developed by N. E. Huang and etc. in the late 1990s, is particularly useful for frequency extraction because it decomposes a signal without need of a prior basis function, unlike traditional Fourier or wavelet transforms [19]. Ensemble machine learning models applied on each IMF generates predictions and the summation of each IMF predictions produces the final predictions of the ensemble model as shown in figure 2. Equation 1 describes the IMFs of time series using EMD technique.

$$F(t) = R(t) + \sum_{i=1}^{n} IMFi(t) \quad (1)$$

n

Data captured from the SDN core switches were extracted with number of bytes per every 5mins, 10mins, and 15mins sampling time intervals. Figure 3 describes the different intrinsic mode frequencies produced by EMD technique, there are 13 IMFs and one residue. From the figure 3. it is clear that some are high frequency IMFs, mid frequency IMFs, and some are low frequency IMFs.

3.4. Ensemble Models

Combination of multiple machine learning models referred as ensemble mod- els, ensemble models used to improve the prediction accuracy of the machine learning models. These models combine various weak models to produce a stronger model. There are various ensemble models such as bagging, boosting, stacking, and voting.

3.4.1. Random Forest Model

Random forest (RF) is bagging ensemble machine learning model which in- tern uses decision tree regressor models to predict the SDN traffic. Bagging reduces the variance by training the same model on different random subsets (bootstraps) of the training data (with replacement). The predictions of these individual machine learning models are then averaged (for regression) or voted upon (for classification) to produce the final output. All the models were par- allely trained on the subsets of training data.

3.4.2. eXtreme Gradient boosting model

XGBoost is a boosting ensemble machine learning model, follows the principles of gradient boosting, where an ensemble of decision trees is built sequentially. Each new tree attempts to correct errors made by the previous trees, minimizing an objective function.

3.5. Light gradient boosting model

LightGBM (Light Gradient Boosting Machine) is a gradient boosting ensemble machine learning model developed by Microsoft, optimized for speed and efficiency, especially with large datasets. It introduces several innovations that make it faster and more memory-efficient than traditional gradient boosting implementations like XGBoost. Traditional gradient boosting methods grow trees level-wise (depth-first), which means each level of the tree is expanded until a specified depth is reached. LightGBM, however, uses a leaf-wise (best-first) approach. In leaf-wise growth, the algorithm splits the leaf with the largest loss reduction at each step, which can lead to deeper, more complex trees.

4. Experimental Analysis and Results

The proposed ensemble models EMD-RF, EMD-XGBoost, and EMD-LGBM were applied on the SDN traffic traces collected from SDN switches as shown in figure 2. Initially univariate time series data extracted from the collected pcap files which shown in figure 4. Different IMFs were extracted from these univariate traffic traces in figure 3. Proposed ensemble models were applied on each intrinsic mode frequencies and the predictions were summed together to get the final predictions.

Table 1 gives the detailed description and the performance of the proposed EMD based ensemble machine learning models. MAE, RMSE, MSE, MAPE and NRMSE were used to measure the accuracy of the proposed models. The model yielding the lowest MAE, RMSE, MSE, MAPE, and NRMSE values was considered the best-performing model. From the results for 5mins sampling interval EMD XGB Regressor outperforming than the EMD-Random forest and EMD-LGBM models For remaining 10mins and 15mins sampling intervals

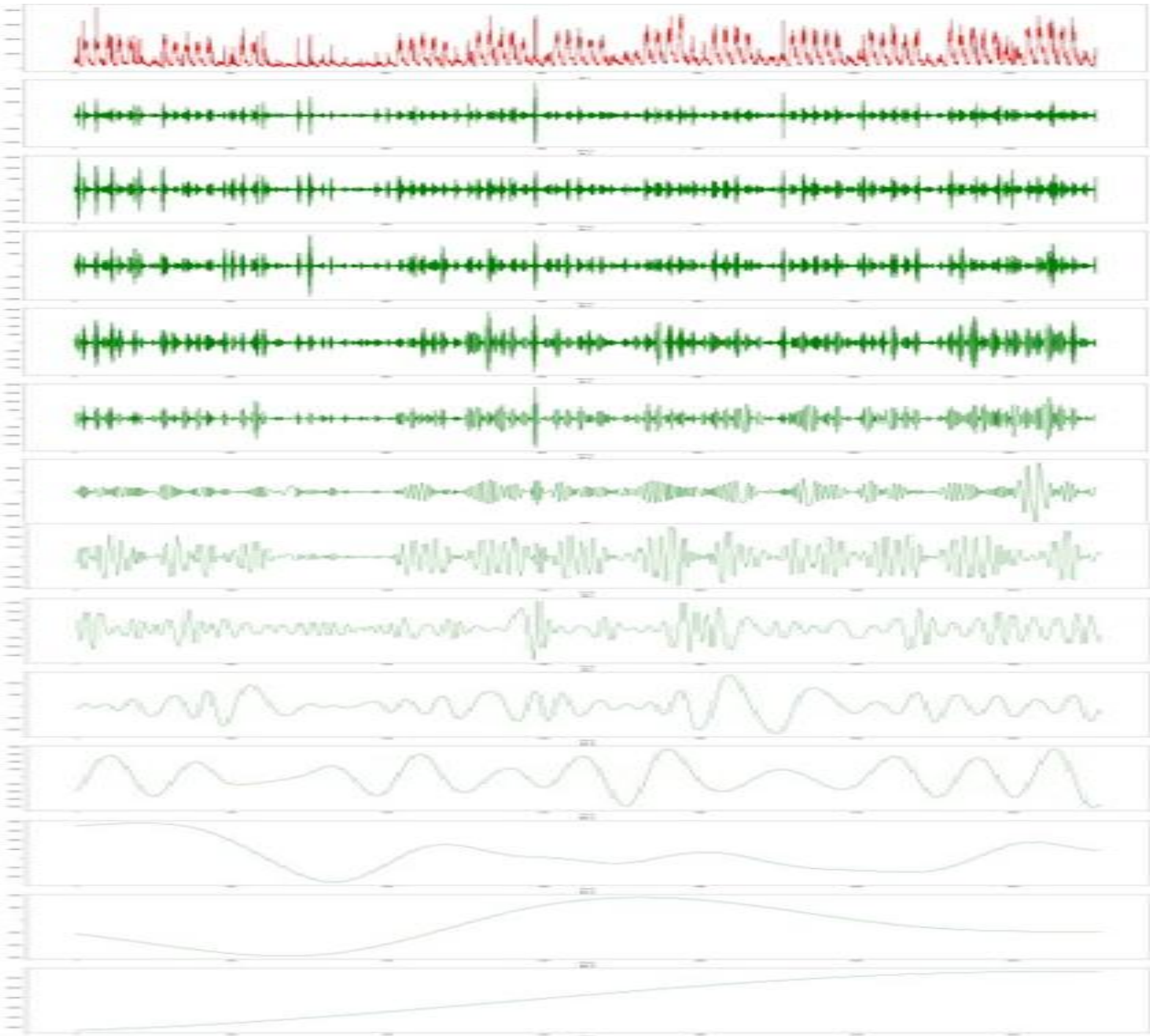

Figure 3: Intrinsic Mode frequencies for 10min interval

```
                      frame.len
Dateandtime
2003-12-07 05:30:00   94839.435
2003-12-07 05:40:00   59537.895
2003-12-07 05:50:00   60293.757
2003-12-07 06:00:00   83634.900
2003-12-07 06:10:00  130818.833
```

Figure 4: Univariate time series data with 10 min sampling interval

EMD-LGBM model performing better than other models. Figure 5 demonstrates the detailed graphical representation of the proposed model performance.

Algorithm	**Minutes**	**MAE**	**RMSE**	**MSE**	**MAPE**	**NRMSE**
EMD Random Forest	5mins	0.025777	0.038515	0.001483	8.230495	0.044434
EMD XGBRegressor	5mins	0.022519	0.035508	0.001261	6.834889	0.040965

EMD LGBM	5mins	0.022949	0.035321	0.001248	7.131885	0.040749
EMD Random Forest	10mins	0.028658	0.041522	0.001724	6.369799	0.047638
EMD XGBRegressor	10mins	0.025838	0.039135	0.001532	5.585143	0.044899
EMD LGBM	10mins	0.025003	0.037083	0.001375	5.456623	0.042545
EMD Random Forest	15mins	0.030261	0.044253	0.001958	7.554563	0.050930
EMD XGBRegressor	15mins	0.027050	0.041507	0.001723	6.473319	0.047770
EMD LGBM	15mins	0.026520	0.039655	0.001572	6.392891	0.045638

Table 1: Experimental results for different sampling intervals

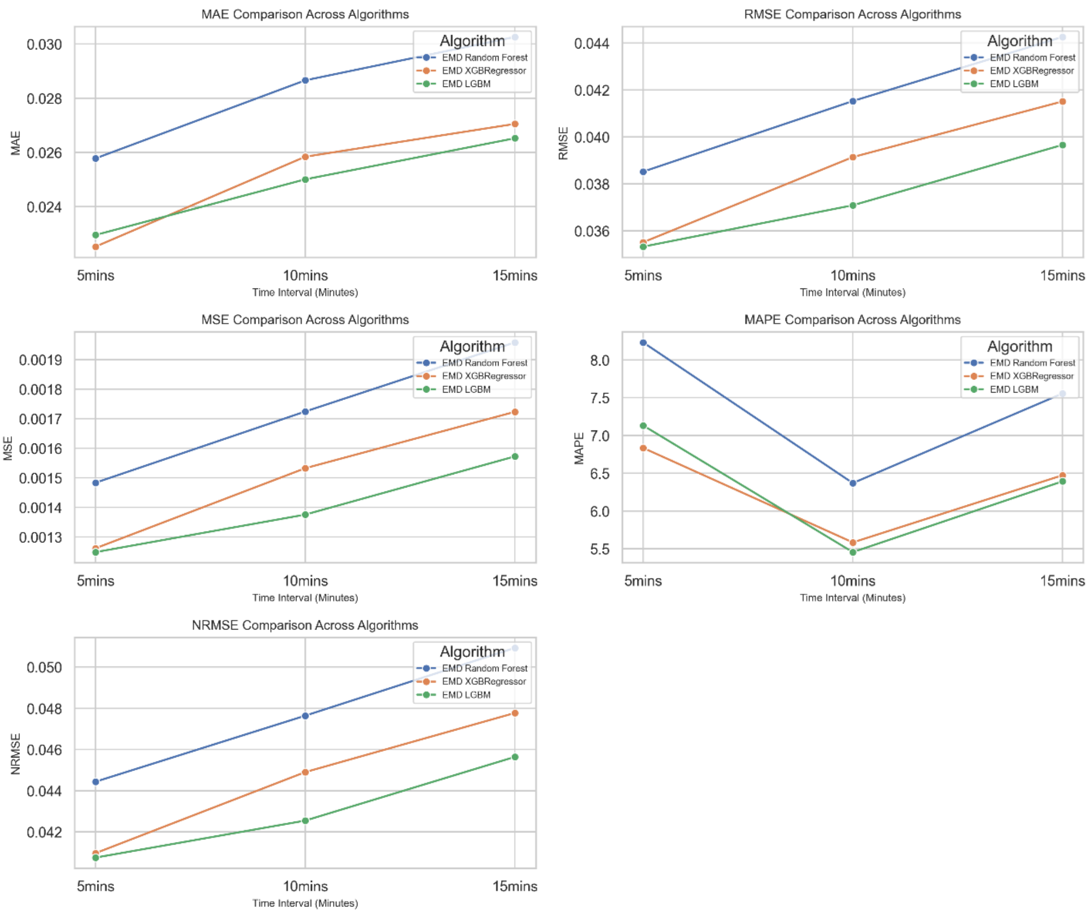

Figure 5: Comparison of models at different time stamps with different metrics

5. Conclusions

A testbed was created using Mininet emulator to collect the SDN cumulative traffic traces. The SDN traffic traces were captured from the core layer switches of fat tree topology by rewriting and replaying the traffic traces collected from CAIDA. Univariate timeseries data sets were produced by extracting traffic volume at three different sampling intervals i.e., 5Mins, 10Mins, and 15Mins. Once the incoming traffic is predicted on a particular link or a switch it can be dynamically adjusted by providing the required resources on demand which reduces the congestion, data loss and improves the quality of service. Upon applying the proposed models on the univariate time series EMD-LGBM outperformed than other models for 10Mins, 15Mins sampling intervals, and EMD- XGBRegressor well performed on 5Mins sampling interval. The analysis of the proposed models concludes that EMD-XGBoost is suitable for short sampling intervals whereas EMD-LGBM can be applied for long sampling intervals to predict SDN traffic.

References

[1] X. Wang, Y. Sun, X. Wang, E. Wang, Z. Sun, Traffic matrix prediction in sdn based on spatial-temporal residual graph convolutional network, in: 2023 35th Chinese Control and Decision Conference (CCDC), 2023, pp. 3382–3387. doi:10.1109/CCDC58219.2023.10327651.

[2] A. Rahman, M. J. Islam, S. S. Band, G. Muhammad, K. Hasan, P. Tiwari, Towards a blockchain-sdn-based secure architecture for cloud computing in smart industrial iot, Digital Communications and Networks 9 (2) (2023) 411–421. doi:https://doi.org/10.1016/j.dcan.2022.11.003. URL https://www.sciencedirect.com/science/article/pii/S2352864822002449

[3] S. Lahlou, Y. Moukafih, A. Sebbar, K. Zkik, M. Boulmalf, M. Ghogho, Td-ra policy-enforcement framework for an sdn-based iot architecture, Journal of Network and Computer Applications 204 (2022) 103390. doi:https://doi.org/10.1016/j.jnca.2022.103390. URL https://www.sciencedirect.com/science/article /pii/S1084804522000510

[4] A. Galal, X. Hesselbach, W. Tavernier, D. Colle, Sdn-based gateway ar- chitecture for electromagnetic nano-networks, Computer Communications 184 (2022) 160–173. doi:https://doi.org/10.1016/j.comcom.2021.12.017. URL https://www.sciencedirect.com/science/article /pii/S0140366421004898

[5] C. Gao, J. Wang, C. Yan, A dual-stage attention based sdn traf- fic prediction method, in: 2022 IEEE Intl Conf on Parallel & Dis- tributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Net- working (ISPA/BDCloud/SocialCom/SustainCom), 2022, pp. 315–322. doi:10.1109/ISPA-BDCloud-SocialCom-SustainCom57177.2022.00047.

[6] Q. Kong, X. Zhang, C. Zhang, L. Zhou, M. Yu, Y. He, Y. Chen, Y. Miao, H. Yuan, Network traffic prediction: Apply the transformer to time series forecasting, Mathe- matical Problems in Engineering 2022 (1) (2022) 8424398. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1155/2022/8424398, doi:https://doi.org/10.1155/2022/8424398. URL https://onlinelibrary.wiley.com/doi/abs /10.1155/2022/8424398

[7] A. Name(s), Network traffic prediction method for lan using linear feature- enhanced convlstm model, [8]Journal/Conference Name (Year).

[8] S. Nyaramneni, T. Potluri, J. Somavarapu, Advanced ensemble ma- chine learning models to predict sdn traffic, Procedia Computer Science 230 (2023) 417–426, 3rd International Conference on Evolu- tionary Computing and Mobile Sustainable Networks (ICECMSN 2023). doi:https://doi.org/10.1016/j.procs.2023.12.097. URL https://www.sciencedirect.com/science/article/pii/S1877050923021026

[9] A. Azzouni, G. Pujolle, Neutm: A neural network-based framework for traffic matrix prediction in sdn, in: NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium, 2018, pp. 1–5. doi:10.1109/NOMS.2018.8406199.

[10] D.-H. Le, H.-A. Tran, S. Souihi, A. Mellouk, An ai-based traffic ma- trix prediction solution for software-defined network, in: ICC 2021 IEEE International Conference on Communications, 2021, pp. 1–6. doi:10.1109/ICC42927.2021.9500331.

[11] Shahul, Hamead, Haja, Moinudeen. (2024). 2. A Consensus-based Traffic Prediction for Time Series Data in SDN. Indian journal of science and technology, doi: 10.17485/ijst/v17i34.2061

[12] Zeinab, Nazemi, Absardi., Reza, Javidan. (2024). 5. A predictive SD-WAN traffic management method for IoT networks in multi-datacenters using deep RNN. Iet Communications, doi: 10.1049/cmu2.12810

[13] Yanying, Xu. (2024). 6. Machine Learning Based Traffic Prediction and Congestion Control Algorithms in Software Defined Networks. doi: 10.1109/iist62526.2024.00035

[14] Getahun, Wassie., Jianguo, Ding., Yihenew, Wondie. (2023). 8. Traffic prediction in SDN for explainable QoS using deep learning approach. Dental science reports, doi: 10.1038/s41598-023-46471-8

[15] Daniel, Adanza., Lluís, Gifre., Pol, Alemany., J.P., Fernández-Palacios., Oscar, González-de-Dios., Raül, Muñoz., Ricard, Vilalta. (2024). 9. Enabling traffic forecasting with cloud-native SDN controller in transport networks. Computer networks, doi: 10.1016/j.comnet.2024.110565

[16] Maltare, N. N., Sharma, D. & Patel, S. (2023). An Exploration and Prediction of Rainfall and Groundwater Level for the District of Banaskantha, Gujrat, India. International Journal of Environmental Sciences, 9(1), 1-17. https://www.theaspd.com/resources/v9-1-1-Nilesh%20N.%20Maltare.pdf

[17] M. Alonso, S. Coll, J. Martinez, V. Santonja, P. Lopez, Power consumption management in fat-tree interconnection networks, Parallel Computing 48 (2015) 59–80. doi:https://doi.org/10.1016/j.parco.2015.03.007.

[18] S. Wei, Y. Liu, Athlete body fat rate monitoring and motion image simu- lation based on sdn data center network and sensors, Preventive Medicine 174 (2023) 107617. doi:https://doi.org/10.1016/j.ypmed.2023.107617.

[19] N. Nava, T. Di Matteo, T. Aste, Financial time series forecasting using empirical mode decomposition and support vector regression, Risks 6 (1) (2018). doi:10.3390/risks6010007.

[20] The Center for Applied Internet Data Analysis (CAIDA), CAIDA Anonymized Internet Traces - Dataset Catalog, accessed: 2023-12-17 (2024).