International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Revue d'Intelligence Artificielle
Vol. 36, No. 4, August, 2022, pp. 647-655
Journal homepage: http://iieta.org/journals/ria

Network Data Center Traffic Predictive Model Analysis Based on Machine Learning

Ayushi Kamboj[1], Harikrishnan R[1*], Dayanand Waghmare[2], Priyanka Tupe Waghmare[1]

[1] Symbiosis Institute of Technology (SIT) Symbiosis International (Deemed University), Pune-412115, India
[2] Networking Manager, PTC Software, Pune-411014, India

Corresponding Author Email: dr.rhareish@gmail.com

Special Issue: Technology Innovations and AI Technology in Healthcare

https://doi.org/10.18280/ria.360419

Received: 18 June 2022
Accepted: 22 June 2022

Keywords:
network traffic prediction, prediction, networking, machine learning

ABSTRACT

Due to the transient nature and uncertainty of traffic produced by applications and services, data center networks have a lot of challenges. As a response, networking as a domain is continually evolving to maintain the exponential growth in network traffic. The primary objective of this paper is to predict the network traffic before it impacts the system's performance. This paper first describes existing Machine Learning (ML) applications in telecommunications and then lists the most prominent difficulties and probable remedies for implementing them. We tried to implement different ML algorithms to predict the network traffic like Gradient Boosting (GB), Random Forest (RF), K-Nearest Neighbor (KNN), Adaptive Boosting (AB), Neural Network (NN), Decision Tree (DT), and Support Vector Machines (SVM) with different sub-parameters for predicting network traffic. Relying on a sequential dataset, we create the corresponding ML environment and present a comparison table of Mean square error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Coefficient of Determination (R2) for each model. The simulation results show that the AB and GB are the best-fitted models with performance matrix parameters like MSE 0.000 and RMSE 0.002 and 0.011, respectively. The orange tool is used to stimulate the predictive models.

1. INTRODUCTION

The rapid expansion of the internet and telecommunication equipment has resulted in larger and higher intricate network architectures, necessitating the adaptation and creation of more significant hubs, routers, switches, and other network components. This network complication has resulted in a massive influx of traffic data, posing issues in network monitoring and traffic control, such as traffic measurement (e.g., traffic categorization) and traffic forecast. Wireless mesh networks are frequently utilized to give distributed access to users and other intelligent devices. Meanwhile, it might offer last-mile connectivity for various network applications, including IoT and mobile networks. Advanced telecommunication machines and infrastructures, such as IoT and wireless technologies, produce massive amounts of diverse traffic data. Traditional connectivity management methodologies for observing and data analytics face several challenges in such networks, including precision and efficient real-time big data processing.

Data centers (DC) enable crucial architecture such as clustering, memory, and networking for diverse operations that rely on these resources. Cloud services, which are often housed in vast data centers, are now responsible for a large portion of the growth in Internet data traffic. The surge causes challenges for intra- and inter-data center networks due to the dynamic nature and uncertainty of the traffic generated by such services. Data Center Networks (DCN) traffic trends include massive data transfers of several gigabytes. The network's business demands are like tides; there are high tides and low tides, and a cloud service with flexible services was established to be more resource-efficient. Variable network needs are unavoidable since different users have different business demands. Cloud services may provide consumers with their expected benefits through flexible network administration. Data must travel through routers, switches, and other network element equipment at the network layer. As per Cisco`s yearly internet survey, by 2023, more than two-thirds of people worldwide will have internet connectivity. The growth of online customers will have climbed from 3.9 billion to 5.3 billion by 2023 (Sneha, 2020). The rapid surge in network activity will cause considerable network bottlenecks. The bottleneck is one of the most critical concerns because of the network's excess traffic compared to its execution capability. Many strategies have been developed to detect and address network bottlenecks.

Therefore, network traffic will become the primary cause of concern before it impacts the lifestyle of ordinary human beings. We are looking for a quick, reliable, cost-effective method to predict network traffic to overcome this situation accurately. Machine learning technology is being employed to forecast network traffic will yield more precise and faster outcomes. Several perspectives have been proposed in the literature. The data is collected from the data center for this work, which is one month's worth of appliance data, including complete details of transmitted and received bits. AB model with SAMME classification algorithm using exponential regression loss function and linear regression loss function, Extreme Gradient Boosting (xgboost) model also outperforms when network traffic is predicted using seven different machine learning models. Hence, the predictive models used in this paper will be implemented in the healthcare department in the future to compare the precision of other ML models.

In the real networking world, a large amount of data transfer occurs from source to destination. While sharing the data from source to destination, proper network infrastructure is required to minimize bandwidth utilization, mitigate packet drops, and reduce the transmission time. The congestion problem can be controlled if enabling the better utilization of shared network infrastructure.

We will make the following contributions:

- The author used the data center (real-time) data to predict network traffic.
- The author used the regression models to predict the network traffic, provide a comparative analysis of the simulation results, and proposed the best-suited model for network traffic prediction using the time-series data.

The next sections are organized as follows: Section 2 discusses reviews of relevant work, section 3 discusses the suggested methodology, section 4 analyses simulation results and predictive models, and section 5 discusses the conclusion and future scope.

2. RELATED WORK

Traffic forecasting is crucial for efficiency evaluation and networking design. Different mechanisms are used to avoid congestion and predict network traffic. Author (Abdullah Baz, 2018) formed the Bayesian Machine Learning (BML) framework to enable switches to attenuate the fundamental stochastic approach by which the controller separated frames into flows. The suggested program's efficacy was evaluated to the conventional process outlined in the present state-of-the-art SDN deployment employing rigorous computation [1]. As a result, the discipline of networking is constantly evolving to keep up with the massive increase in network traffic. Even though techniques like Software Defined Networking (SDN) can enable a centralized control framework for network traffic monitoring, management, and forecasting, the amount of data collected by the SDN controller is enormous. Machine Learning has lately been offered as a way to handle data (ML) [2].

Arjun Roy, the author, looks into network activity in a few of Facebook's data centers. On the other hand, Facebook uses specialized data center services like Hadoop. Its leading Web service and underlying cache technology have several characteristics that haven't been seen before in the literature. Disparities in network traffic localization, reliability, and predictability in Facebook's data centers have ramifications for network topology, traffic engineering, and switch layout, according to the authors [3].

The author, Larsen Nie, offers a deep learning architecture and the Spatiotemporal Compressive Sensing approach for network traffic prediction. The suggested technique uses a discrete wavelet transform to extract the low-pass component of network traffic, which characterizes the network's long-range reliance. As a result, a prediction model is created by learning a deep architecture based on the extracted low-pass component's deep belief network. Otherwise, the Spatiotemporal Compressive Sensing approach estimates the remaining high-pass element, reflecting network traffic's gusty and irregular oscillations. The author can create a network traffic predictor using the predictors of two parts. The suggested prediction approach beats three current methods in simulation [4]. Due to the rapid growth in worldwide web congestion, the development of social media networking, and the rise in mobile phone usage, the demand for uplink traffic is expanding. On the other hand, prior traffic forecasting research has focused on the downlink, with the uplink receiving less attention. The findings demonstrate that ARIMA (0,1,5), ENN (1-25-1), and MLP (1-25-1) modeling techniques can accurately anticipate 3G uplink traffic hourly, for daily ARIMA (4,1,1), ENN (1-24-1) and MLP (1-24-1) are the best-suited models, and ARIMA (2,1,2), ENN (1-10-1) and MLP (1-35-1) are best-suited models for weekly prediction [5]. The KNN, with an accuracy of 92.4214, outperforms the Nave Bayes Algorithms with an accuracy of 81.7922%. The Decision Tree Algorithm takes an accuracy of 92.0104%. The SVM has a mean accuracy value of 89.6061% in terms of efficiency. With the constantly growing customer base of online social networks (OSNs), the investigation into OSN application-specific mobile network traffic has gotten much attention in recent decades. The suggested paradigm offers significant forecasting reliability 10% better than that using a NN with minimal computation complexity and little requirement for the size of the dataset, according to experimental observations relying on real Twitter traffic gathered in central London. The author plans to look into the model's adaptability, durability, and viability in the future [6].

In wireless IoT, allocating appropriate channels is essential for high transmission. However, because of the highly dynamic traffic volumes in the IoT, traditional constant channel assignment techniques are ineffective. Software-Defined Networking-based IoT (SDN-IoT) has lately been recommended to increase transmission efficiency. Furthermore, deep learning has been extensively investigated in high computational SDN. Fengxiao Tang presents a unique intelligent channel assignment methodology (TP-DLPOCA) that uses deep learning-based traffic forecasting and channel allocation to minimize traffic congestion and swiftly allocate appropriate channels to SDN-IoT wireless networks. Extensive modeling findings show that our solution surpasses traditional channel allocation algorithms by a wide margin [7].

The impact of seasons and covid-19 attributes on domestic electricity consumption was studied and predicted using five different machine learning algorithms: RF, Linear Regression, SVM, NN, and AB. The author [8] used the orange tool to simulate the predictive frameworks. The NN model is the most well-fitting prototype in the predictive modeling study. The performance matrix parameters of the NN model have MSE, RMSE, and MAE values of 0.558, 0.747, and 0.562, respectively [8]. While the orange tool can be implemented for classification, development of predictive modeling, testing, and score analysis [9], it has a vast library of data mining components and is appropriately managed.

Using genuine DCN traces, author Jeandro de M. Bezerra analyzes the efficacy of various elephant flow predictions. They use sequential traffic data from a Facebook data center, modeling the elephant flow predictions on a short-term basis and providing qualitative statistics and inference about the flows. Also, a composite forecasting framework was proposed by merging parts of the FARIMA and Recurrent NN (FARIMA-RNN) models, showing an RMSE of 0.0159. Also, FARIMA-MLP shows an RMSE of 0.0871. To evaluate the effectiveness of the hybrid paradigm with the ARIMA shows RMSE of 0.15, GARCH with RMSE of 0.14, RBF takes the

RMSE of 0.26, MLP gives RMSE of 0.15, and LSTM models with RMSE of 0.0304, a framework relying on the score of the forecasting precision metrics is employed. The analysis enables DCNs to choose an optimal framework for implementing elephant flow scheduling approaches [10].

During hurricanes Matthew and Irma, the author Kamol Chandra Roy used traffic sensors and Twitter data to estimate transportation demand during evacuation and show a machine learning technique based on Long-Short Term Memory NN (LSTM-NN) that was trained on real-world traffic data during storm migration (hurricanes Irma and Matthew) using a variety of source factors and anticipated timeframes [11].

EL Hocine Bouzidi first formulates the QoS-aware routing issue as a Linear Program (LP) to minimize end-to-end (E2E) latency and network usage. Then, to solve it, suggest a simple yet effective heuristic algorithm. According to numerical data obtained through simulation, the proposed technique can significantly increase network performance by lowering link usage, packet loss, and E2E latency, according to numerical data obtained through simulation utilizing the ONOS controller and Mininet [12]. In almost all networking applications, specific network traffic load (TL) forecasting is required. Lo Pang-Yun Ting looks into a real-world network traffic database to evaluate actual TL characteristics and see if distance-correlation among regions in a spatial graph will help predict. As a result, we present a time-series model-based strategy for efficiently considering distance correlation. Empirical tests on real data show that our suggested strategy can significantly minimize error values by at least 10% in regions having increased TLs [13].

Deep learning methodologies improve analytical and knowledge acquisition in massive data sets by recognizing hidden and complicated patterns. Networking researchers perform network traffic monitoring and analysis (NTMA) operations such as traffic categorization and forecasting using deep learning frameworks [14]. The embedding theorem and residual modeling with a hybrid Elman–NARX NN are used to evaluate and forecast the chaotic time series. An Elman neural network is deployed to input and train the embedded phase space points. The residuals of anticipated time series are investigated, and they are found to have chaotic behavior. According to numerical empirical outcomes, the suggested technique may predict chaotic time series in a more optimized way and precisely than already presented predictive mechanisms [15]. Many networking tasks, like dynamic provisioning and power optimization, necessitate accurate real-time traffic forecasting. Muhammad Faisal Iqbal looks at a variety of forecasts with the hopes of finding one with high precision, low processing complexity, and common energy usage. According to the outcomes, a dual exponentially smoothed estimator provides a proper equilibrium between efficiency and cost overhead [16].

The subject of energy conservation in data centers has lately piqued researchers' interest, and the adaptive datacenter activation paradigm has arisen as a viable solution for energy conservation. Due to its complexity, this model does not include adaptive activation of switches and hosts in data centers. Min Sang Yoon offers an adaptive datacenter activation model that combines adaptive switch and host activation while also incorporating a statistical request prediction technique. The learning system uses a cyclic window learning technique to predict user requests in a predefined interval. The data center then activates an appropriate number of switches and hosts to reduce anticipated power usage [17]. The author utilizes the Simulated Annealing approach to solve the model because it is NP-hard. As a result, the author will use Google cluster trace data to test our prediction model. The projected data is then used to evaluate the adaptive activation model and track the energy savings rate at each interval. The experiment discovered that the adaptive activation model saves 30 to 50% of energy in practical data center operating conditions compared to the entire functioning state [17].

Flow size is essential for computer network routing, load balancing, and scheduling, according to author Pascal Poupart. Because flow patterns are constantly changing, forecasting flow size is extremely difficult. To avoid delays, projections must be prepared milliseconds in advance. Analyze the prediction nature of a set of variables and the efficacy of three online forecasters based on NN, Gaussian process regression, and online Bayesian Moment Matching on three datasets of real traffic. The use of online forecasts to optimize routes is demonstrated in a network simulation [18].

DCNs have experienced some issues because of the variable nature and uncertainty of traffic produced by apps and resources. It’s become a challenging task to collect the data from the DC as to open-source data. Therefore, in this research, we are using one of the appliance data from PTC software India Pvt. Ltd. [19].

Author Shi Dong uses network flow-level features; therefore, a new SVM technique called cost-sensitive SVM (CMSVM) is proposed to overcome the disproportion issue in network traffic detection. Also, using two separate datasets, the MOORE SET and NOC SET datasets, analyze the CMSVM algorithm's categorization efficiency and throughput. Compared to other machine-learning approaches, the CMSVM methodology reduces computing costs, improves the classification efficiency, and solves the imbalance challenge [20]. Internet Service Providers (ISPs) typically employ network traffic classifications to assess the attributes required to build a network, impacting the network's overall effectiveness. There are various approaches for categorizing network protocols, including port-based, payload-based, and ML-based methods, all of which have benefits and drawbacks. Machine Learning has been increasingly popular in recent years due to its extensive use in various fields and growing knowledge among investigators of its better precision than other approaches [21].

The network's SDN Controller has a complete understanding of the networking framework and its constituents, allowing it to alter the routing info of the switches. SD predicts uses sequence-to-sequence modeling trained on-network data congestion to anticipate traffic in the SDN, then modeled employing an Artificial Neural Network (ANN) to estimate packet routing [22]. For traffic prediction and concurrent route optimization, SDPredictNet achieved an RMSE score of 0.07 and a precision of 99.88% [22].

Author Yi Li combined wavelet transform with an ANN to increase forecasting effectiveness. Add sub-link traffic and elephant flows, the least anticipated but most dominant traffic in the inter-DC network, to the proposed prediction model. As a result, the author can diminish forecasting errors by 5% to 30% compared to conventional methods. Our forecast is in use at Baidu, one of China's top Internet corporations, as part of the traffic scheduling system, assisting in reducing peak network capacity [23]. The author Y. V. Sneha discusses the importance of machine learning methodologies in networking. Relying on queue parameters, the proposed machine learning framework can identify congestion in the router. Using an ns-3 simulator finally, effectiveness measures evaluate the ML model's performance. The simulation findings show that the Naive Bayes approach outperforms the SVM method in terms of accuracy [24]. Table 1 shows the summary of workdone for network traffic prediction.

SDN is a cutting-edge network approach that divides network switch equipment's control and data surfaces, allowing increased network monitoring and management flexibility. The flow-based processing concept's SDN architecture claims to collect and anticipate network traffic [25]. Finally, the suggested method is validated by a significant number of experiments that have been demonstrated and constructed. The proposed strategy is viable and efficient, according to modeling findings.

Table 1. Summary of workdone for network traffic prediction

Model Used (Category)	Significant Contribution	Results
ARMA Model (Time-series)	For SDN systems, propose a basic traffic forecasting approach.	The ARMA paradigm would reliably anticipate network traffic behavior trends [1].
KNN, Naïve, Bayes Algorithm, DT Algorithm, and SVM (Machine Learning)	Machine Learning methodology is employed to manage network efficiency and classify unknown applications.	KNN gave a mean accuracy of 92.4214 and outperformed the other algorithms [2].
Statistical analytics, linear regression, and neural network (Machine Learning)	Suggest a technique for predicting Twitter traffic that incorporates statistical analysis and machine learning approaches.	It offers 10% improved results than without quantitative statistical evaluation and significantly improved results than employing a NN [3].
Cost-sensitive SVM (CMSVM), (Machine Learning)	A hybrid predictive framework is proposed by merging parts of the FARIMA, and the RNN (FARIMA-RNN) approaches.	FARIMA – RNN shows the RMSE of 0.0159 and outperforms other learning algorithms [10].
K-means clustering (Machine Learning)	To address the unbalanced issues in network traffic categorization, a novel support vector machine (SVM) technique known as cost-sensitive SVM is recommended (CSSVM).	CMSVM gets 70% of the geometric mean, outperforming the other learning algorithms [11].
Deep Belief Network and Spatiotemporal Compressive Sensing (DBNSTCS) (Deep Learning)	Utilizing Long-Short Term Memory Neural Networks, describe a machine learning approach (LSTM-NN).	With a 90% confidence range, LSTM-NN is anticipated [15]

3. METHODOLOGY USED

This paper uses the seven supervised ML models to predict network traffic such as GB, RF, KNN, AB, NN, DT, and SVM with different sub-parameters. ML plays a vital role in networking to handle the extensive database. The ML approach is used for forecasting and classification in this work. The different researchers used different methods and technologies to predict the data center data. In this work, we use supervised learning models to predict network traffic. Multiple supervised ML techniques have been used to forecast network traffic. The main aim of making network traffic predictions is to avoid congestion. If we can expect the network traffic before it impacts the network performance, it might increase the network performance, as well as efficiency, will increase. The transmission time between source and destination will decrease indirectly. And therefore, it will reduce the latency. The proposed model for predicting network traffic data is shown in Figure 1. At the same time, the description of each block is explained in the other sections.

3.1 Raw database

The datasets for network traffic prediction are taken from one of the data center appliances from PTC Software India Private Ltd. For this paper, real-time data, a sequential one-month database, is used (https://www.ptc.com) [19].

3.2 Data preprocessing

Two tools are used for this research to pre-process the raw data, such as the excel tool and the orange tool (https://orangedatamining.com). With the excel tool, able to transform the data into a more precise form, and we can remove the unwanted components such as columns or rows. While the second tool used for data pre-processing is the orange tool, we can remove the rows with missing values by making our data into a more precise form. Our raw data is in the form of an excel sheet. We have to save the excel sheet in the csv format first. Figure 2 shows the implementation workflow for pre-processing data using the orange tool. As shown in Figure 2, we imported our raw data Comma Separated Values (CSV) file by selecting the CSV file import widgets after using the preprocess widget. Figure 3 shows the chosen attributes, which imputes the missing values, then sub-parameter select has removed the rows with missing values. We refine our data by using the preprocessor. It removes all the rows which consist of no importance or no data. Therefore, we get data in a more precise form with no missing values after implementation.

3.3 Prediction model used

Several supervised learning models anticipate network traffic using the real-time data center database. As we are using the regression model to predict the network traffic, the statistical accuracy index cannot be used to evaluate prototype performance, unlike the classification approach, where the accuracy of predictions depicts the outcome of modeling techniques.

The RMSE, R2, and MAE indices are instead used to evaluate the models. The RMSE and MAE are statistical measures that show how true and false statements differ. R2 explains the difference between the actual and estimated response. The value of R2 lies between 0 and 1. As a result, a measurement close to 1 is required, indicating that the projected response is very similar to the actual response.

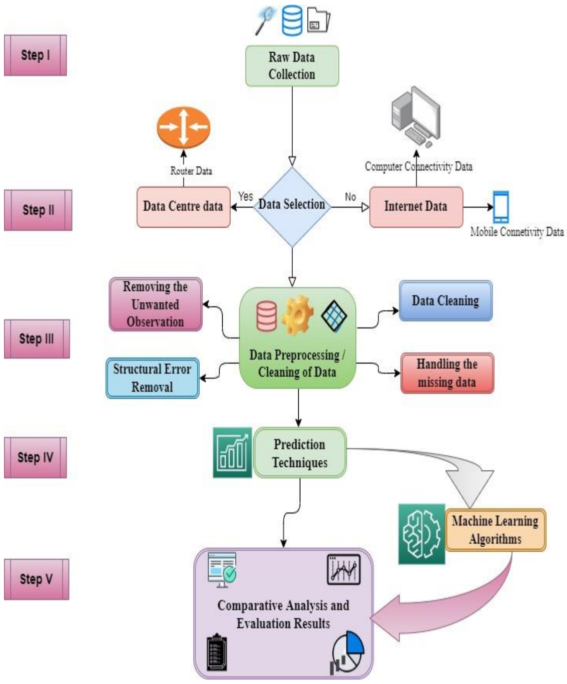


Figure 1. Proposed model for network traffic prediction

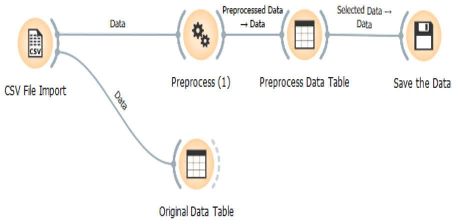
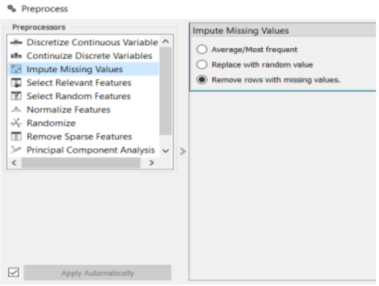


Figure 2. Workflow of preprocessing of data

Figure 4 shows the different prediction models employed to suggest network traffic. AB, GB, DT, KNN, SVM, RF, and NN with variable sub-parameters are the seven prediction models used to predict network traffic.

- Adaptive Boosting (AB) – AB is a composite meta-algorithm that can help weak learners improve their proficiency. On the input stage, the learning method is also available; the AB learning algorithm is employed as a learner on the output edge [9].
- Gradient Boosting (GB) – Gradient Boosting is a machine learning methodology for regression and categorization problems that generate forecasting models from an array of weak forecasting models, usually decision trees [9].
- Decision Tree (DT) - DT is a simple method for dividing data into nodes based on the purity of the class. It is the forerunner to RF. Tree in Orange is an in-house tool for categorized and numeric data management [9].
- Random Forest - The RF comprises a collection of decision trees. Each tree is created using a bootstrap sample from the training dataset. Individual trees are constructed by selecting the best feature for the split from an arbitrary set of attributes. The final design is based on a majority vote among the forest's independently grown trees (https://orangedatamining.com/widgetcatalog/model/randomforest/) [9].
- Support Vector Machine (SVM) - The SVM is an ML technique that splits the feature space by employing the hyperplane. This approach improves forecasting effectiveness by fine-tuning hyperparameters like regression cost (C), regression loss epsilon, and kernel size [9].
- K-Nearest Neighbour (KNN) - The KNN methodology uses 'feature correlation' to predict the value of new data bits. As a result, the value of the newest point is determined by how similar it is to the bits in the training group [9].
- Neural Network (NN) – A backpropagating multi-layer perceptron (MLP) methodology. The NN widget uses sklearn's Multi-layer Perceptron method, which can learn both linear and non-linear structures [9].


Preprocess

Preprocessors

- Discretize Continuous Variable
- Continuize Discrete Variables
- Impute Missing Values
- Select Relevant Features
- Select Random Features
- Normalize Features
- Randomize
- Remove Sparse Features
- Principal Component Analysis

Impute Missing Values

○ Average/Most frequent
○ Replace with random value
● Remove rows with missing values.

☑ Apply Automatically

Figure 3. Attributes selected for data preprocessing

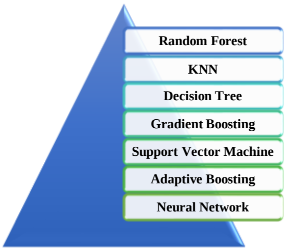


Figure 4. Network traffic prediction algorithms

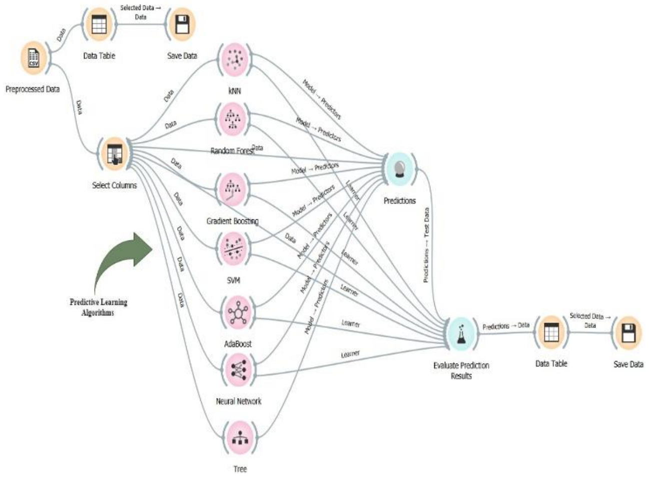


Figure 5. Workflow of the methodology used

Using these regression models for prediction, we use one of the appliance data from DC, a time-series continuous data, and it`s impossible to classify or categorize the data as discrete values.

For simulation orange tool with version 3.31.1 (https://orangedatamining.com) is used; the reason behind using the orange tool is because it is an open-source tool; we can customize the sub-parameters for the different models according to our requirements. At the output, we get a comparison table of varying error metrics. So, it became easy to understand and evaluate the model outputs [8, 9].

Figure 5 depicts the workflow of network traffic prediction models. The workflow consists of main parts such as training and testing the database, learning algorithms, prediction, and evaluation by using tests and scores. While for training-testing and validation, we use a training set size of 80% while a 20% set size for testing and implementing the sampling as a test on train data. The prediction models with tunned parameters are implemented to get the best results.

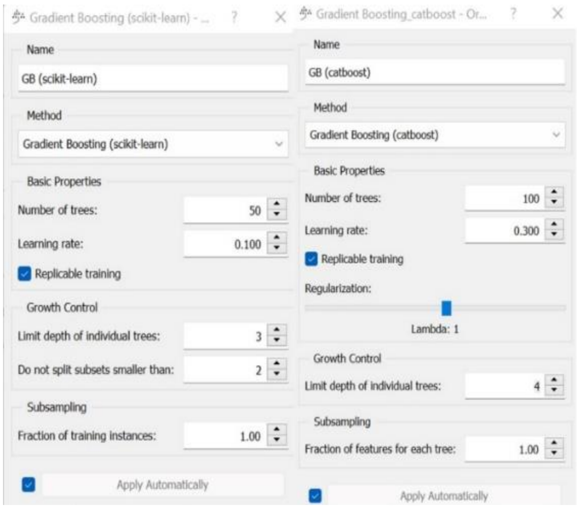
4. SIMULATION RESULT ANALYSIS AND DISCUSSION OF PREDICTIVE MODELS

This paper implements seven different prediction algorithms with multiple tunned sub-parameters. Table 2 shows the prediction results obtained after executing the various algorithms with tunned parameters and sampling models. The prediction results are analyzed concerning four parameters as MSE, RMSE, MAE, and Coefficient of Determination (R2).

Table 2. Prediction outcomes after numerous methods have been implemented for test on train data

Model Used	MSE	RMSE	MAE	R2
Random Forest_Tunned	0.042	0.206	0.062	1.000
KNN_Chebyshev	0.191	0.437	0.242	0.997
KNN_Euclidean	0.229	0.479	0.252	0.997
KNN_Manhattan	0.271	0.520	0.270	0.996
KNN_Mahalanobis	103.605	10.179	7.028	-0.479
Gradient Boosting _ Scikit-learn	0.007	0.082	0.059	1.000
Extreme Gradient Boosting (xgboost)	0.000	0.011	0.008	1.000
Extreme Gradient Boosting Random Forest (xgboost)	84.709	9.204	6.180	0.063
Gradient Boosting _ Catboost	0.036	0.189	0.135	1.000
Tree (DT)	0.060	0.244	0.045	0.999
Neural Network (NN)	176.672	13.292	9.105	-0.884
AdaBoost _ SAMME _ Square	0.002	0.045	0.023	1.000
AdaBoost _ SAMME _ Exponential	0.000	0.002	0.001	1.000
AdaBoost _ SAMME _ Linear	0.000	0.005	0.002	1.000

Figure 6 and Figure 7 depict the parameter selected for GB for scikit-learn, Catboost, xgboost, and xgboost RF to predict the network traffic. Figure 8 shows the parameters chosen for the AB with the linear and square regression loss function. In contrast, Figure 9 demonstrates the parameters selected for AB with the exponential loss function while SAMME. R classification algorithm remains the same and represents the parameter chosen for RF. Therefore, Figure 10 depicts the parameters selected for the KNN model with different metrics such as Mahalanobis, Chebyshev, Manhattan, and Euclidean while maintaining the weight uniformly to predict the network traffic. While Figure 11 depicts the evaluation results obtained for stratified 20-folds cross-validation.



Gradient Boosting (scikit-learn) - ...

Name
GB (scikit-learn)

Method
Gradient Boosting (scikit-learn)

Basic Properties
Number of trees: 50
Learning rate: 0.100
☑ Replicable training

Growth Control
Limit depth of individual trees: 3
Do not split subsets smaller than: 2

Subsampling
Fraction of training instances: 1.00

☑ Apply Automatically

Gradient Boosting_catboost - Or...

Name
GB (catboost)

Method
Gradient Boosting (catboost)

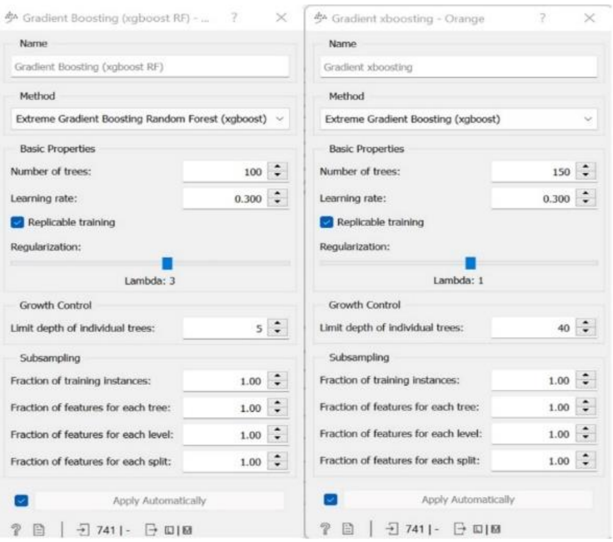
Basic Properties
Number of trees: 100
Learning rate: 0.300
☑ Replicable training
Regularization:
Lambda: 1

Growth Control
Limit depth of individual trees: 4

Subsampling
Fraction of features for each tree: 1.00

☑ Apply Automatically

Figure 6. Parameters selected for GB scikit-learn and Catboost



Gradient Boosting (xgboost RF) - ...

Name
Gradient Boosting (xgboost RF)

Method
Extreme Gradient Boosting Random Forest (xgboost)

Basic Properties
Number of trees: 100
Learning rate: 0.300
☑ Replicable training
Regularization:
Lambda: 3

Growth Control
Limit depth of individual trees: 5

Subsampling
Fraction of training instances: 1.00
Fraction of features for each tree: 1.00
Fraction of features for each level: 1.00
Fraction of features for each split: 1.00

☑ Apply Automatically

741 | -

Gradient xboosting - Orange

Name
Gradient xboosting

Method
Extreme Gradient Boosting (xgboost)

Basic Properties
Number of trees: 150
Learning rate: 0.300
☑ Replicable training
Regularization:
Lambda: 1

Growth Control
Limit depth of individual trees: 40

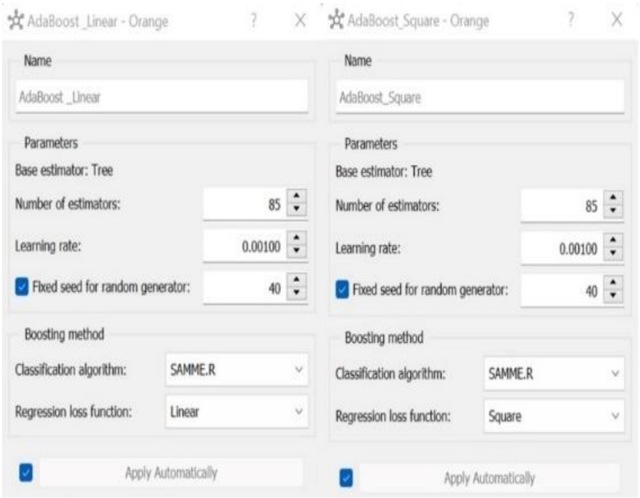
Subsampling
Fraction of training instances: 1.00
Fraction of features for each tree: 1.00
Fraction of features for each level: 1.00
Fraction of features for each split: 1.00

☑ Apply Automatically

741 | -

Figure 7. Parameters selected for GB xgboost and xgboost RF

Table 2 shows that AB with SAMME classification algorithm for exponential regression loss function gives the best result with MSE 0.000, followed by the AB with SAMME classification algorithm, for the linear regression loss function with MSE 0.000, Extreme Gradient Boosting (xgboost) with MSE 0.000, AB with SAMME classification algorithm, for square regression loss function with MSE 0.002, Gradient Boosting _ Scikit-learn with MSE of 0.007, Gradient Boosting _ Catboost with MSE of 0.036, Random Forest_Tunned with MSE of 0.042, Decision tree with MSE of 0.060. While the NN gives the worst results with an MSE of 176.672, followed by KNN_Mahalanobis with an MSE of 103.605, and Extreme Gradient Boosting Random Forest (xgboost) with an MSE of 84.709.


AdaBoost_Linear - Orange

Name: AdaBoost_Linear

Parameters
Base estimator: Tree
Number of estimators: 85
Learning rate: 0.00100
☑ Fixed seed for random generator: 40

Boosting method
Classification algorithm: SAMME.R
Regression loss function: Linear

☑ Apply Automatically

AdaBoost_Square - Orange

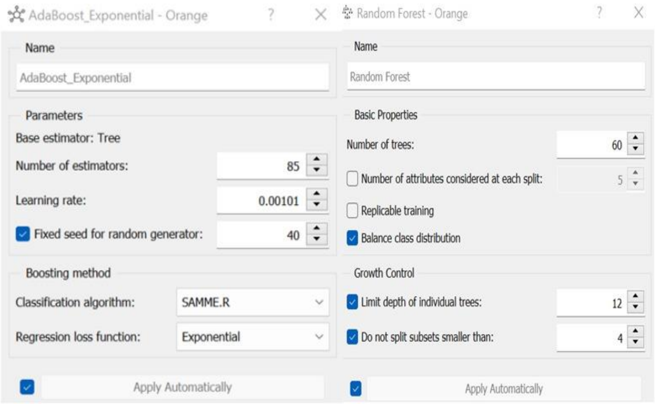
Name: AdaBoost_Square

Parameters
Base estimator: Tree
Number of estimators: 85
Learning rate: 0.00100
☑ Fixed seed for random generator: 40

Boosting method
Classification algorithm: SAMME.R
Regression loss function: Square

☑ Apply Automatically

Figure 8. Parameters selected for the AB with the linear and square regression loss function


AdaBoost_Exponential - Orange

Name: AdaBoost_Exponential

Parameters
Base estimator: Tree
Number of estimators: 85
Learning rate: 0.00101
☑ Fixed seed for random generator: 40

Boosting method
Classification algorithm: SAMME.R
Regression loss function: Exponential

☑ Apply Automatically

Random Forest - Orange

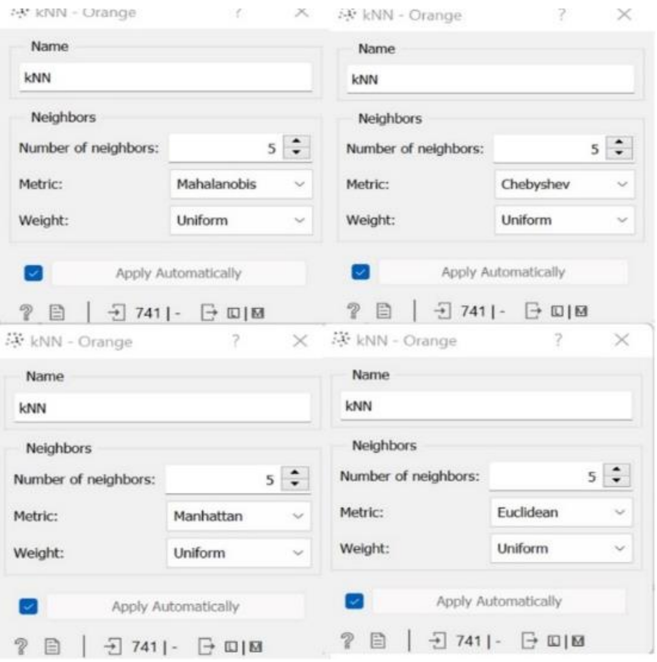
Name: Random Forest

Basic Properties
Number of trees: 60
□ Number of attributes considered at each split: 5
□ Replicable training
☑ Balance class distribution

Growth Control
☑ Limit depth of individual trees: 12
☑ Do not split subsets smaller than: 4

☑ Apply Automatically

Figure 9. Parameters selected for AB with the exponential loss function and RF


kNN - Orange

Name: kNN

Neighbors
Number of neighbors: 5
Metric: Mahalanobis
Weight: Uniform

☑ Apply Automatically

741 | -

kNN - Orange

Name: kNN

Neighbors
Number of neighbors: 5
Metric: Chebyshev
Weight: Uniform

☑ Apply Automatically

741 | -

kNN - Orange

Name: kNN

Neighbors
Number of neighbors: 5
Metric: Manhattan
Weight: Uniform

☑ Apply Automatically

741 | -

kNN - Orange

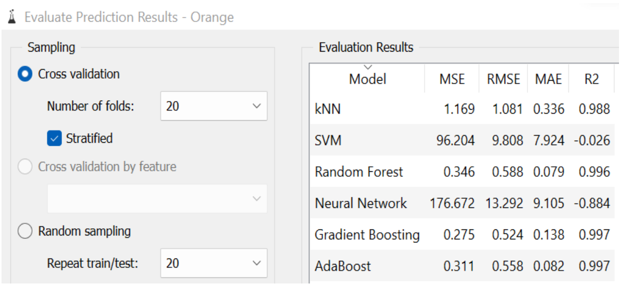
Name: kNN

Neighbors
Number of neighbors: 5
Metric: Euclidean
Weight: Uniform

☑ Apply Automatically

741 | -

Figure 10. Evaluation parameters were selected for KNN with uniform weight.


Evaluate Prediction Results - Orange

Sampling
● Cross validation
Number of folds: 20
☑ Stratified
○ Cross validation by feature
○ Random sampling
Repeat train/test: 20

Evaluation Results

Model	MSE	RMSE	MAE	R2
kNN	1.169	1.081	0.336	0.988
SVM	96.204	9.808	7.924	-0.026
Random Forest	0.346	0.588	0.079	0.996
Neural Network	176.672	13.292	9.105	-0.884
Gradient Boosting	0.275	0.524	0.138	0.997
AdaBoost	0.311	0.558	0.082	0.997

Figure 11. Evaluation results obtained for stratified 20-folds cross-validation

Therefore, the average error of the AB is 0.000001, while the average error for GB is 0.000005; for xgboost, the average calculated error is 0.008705, followed by GB_Catboost 0.19026. While for the Decision Tree, the deliberate average error is 0.04, for RF, it is 0.0588. From Figure 11, we can conclude that if we implement the stratified cross-validation, the error increases compared to test on train data, as depicted in Table 2. Hence, we can conclude that the AB and GB give good results that are less prone to errors for tests on train data.

5. CONCLUSION AND FUTURE SCOPE

The primary purpose of this paper is to predict network traffic. The author proposed to use machine learning methodologies to forecast network traffic. Real-time, sequential data is used for this work. The author seeks to implement various machine learning techniques for network traffic prediction in this study. Seven supervised learning approaches, including KNN, NN, RF, AB, DT, GB, and SVM, are employed to anticipate network traffic. The simulation results show that Adaptive Boosting (MSE of 0.000) followed by gradient boosting (xgboost shows MSE of 0.000) gives the best results followed by other models. In contrast, the Neural Network (NN) (shows the MSE of 176.672) followed by KNN_Mahalanobis (gives MSE of 103.605), and Extreme Gradient Boosting Random Forest (xgboost RF) (shows MSE of 84.709) gives the worst result. For this real-time work, the database is used.

From the literature also, we can conclude that the bootstrapping algorithms are the least implemented algorithms over the real-time data to predict the network traffic. We implemented the Boosting algorithms and other machine learning algorithms in this work. We found that the boosting algorithms such as AB with linear and exponential loss function (shows the MSE 0.000) and xgboost (gives MSE of 0.000) outperformed the machine learning algorithms implemented to real-time data.

The analysis could be extended to predict network traffic with different boosting models as a future scope. It will provide a path for the new learner to compare the performance of time-series algorithms with boosting and other machine learning algorithms. Hence, the prediction accuracy can be analyzed and evaluated. Therefore, similar predictive models can be implemented in healthcare in the future.

ACKNOWLEDGMENT

PTC Software India Pvt. Ltd provided the data used in this work.

REFERENCES

[1] Baz, A. (2018). Bayesian machine learning algorithm for flow prediction in SDN switches. 2018 1st International Conference on Computer Applications & Information Security (ICCAIS), Riyadh, Saudi Arabia, pp. 1-7. https://doi.org/10.1109/CAIS.2018.8441969

[2] Mohammed, A.R., Mohammed, S.A., Shirmohammadi, S. (2019). Machine learning and deep learning based traffic classification and prediction in software defined networking. 2019 IEEE International Symposium on Measurements & Networking (M&N), pp. 1-6. https://doi.org/10.1109/IWMN.2019.8805044

[3] Roy, A., Zeng, H., Bagga, J., Porter, G., Snoeren, A.C. (2015). Inside the social network's (datacenter) network. SIGCOMM '15: Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication, 45(4): 123-137. https://doi.org/10.1145/2785956.2787472

[4] Bouzidi, E.H., Outtagarts, A., Langar, R., Boutaba, R. (2021). Deep Q-Network and traffic prediction based routing optimization in software defined networks. Journal of Network and Computer Applications, 192(March): 103181. https://doi.org/10.1016/j.jnca.2021.103181

[5] Oduro-Gyimah, F.K., Boateng, K.O. (2018). A comparative analysis of telecommunication network traffic forecasting: A three model approach. 2018 IEEE 7th International Conference on Adaptive Science & Technology (ICAST), pp. 1-11. https://doi.org/10.1109/ICASTECH.2018.8506973

[6] Li, F., Zhang, Z., Zhu, Y., Zhang, J. (2020). Prediction of twitter traffic based on machine learning and data analytics. IEEE INFOCOM 2020 - IEEE Conf. Comput. Commun. Work. INFOCOM WKSHPS 2020, no. February, pp. 443-448. https://doi.org/ 10.1109/INFOCOMWKSHPS50562.2020.9162959

[7] Tang, F., Fadlullah, Z.M., Mao, B., Kato, N. (2018). An intelligent traffic load prediction-based adaptive channel assignment algorithm in SDN-IoT: A deep learning approach. IEEE Internet Things J., 5(6): 5141-5154. https://doi.org/10.1109/JIOT.2018.2838574

[8] Ramnath, G.S., R, H. (2021). A statistical and predictive modeling study to analyze impact of seasons and covid-19 factors on household electricity consumption. J. Energy Syst., 5(4): 252-267, https://doi.org/10.30521/jes.933674

[9] Demšar, J., Curk, T., Erjavec, A., Gorup, C. (2013). Orange: data mining toolbox in python tomaž curk mitar milutinovič matija polajnar anže starič. Journal of Machine Learning Research, 14: 2349-2353.

[10] Bezerra, J.M., Pinheiro, A.J., de Souza, C.P., Campelo, D.R. (2020). Performance evaluation of elephant flow predictors in data center networking. Future Generation Computer Systems, 102: 952-964. https://doi.org/10.1016/j.future.2019.09.031

[11] Roy, K.C., Hasan, S., Culotta, A., Eluru, N. (2021). Predicting traffic demand during hurricane evacuation using real-time data from transportation systems and social media. Transportation Research Part C: Emerging Technologies, 131: 103339. https://doi.org/10.1016/j.trc.2021.103339

[12] Nie, L., Wang, X., Wan, L., Yu, S., Song, H., Jiang, D. (2018). Network traffic prediction based on deep belief network and spatiotemporal compressive sensing in wireless mesh backbone networks. Wireless Communications and Mobile Computing, vol. 2018, Article ID 1260860. https://doi.org/10.1155/2018/1260860

[13] Ting, L.P.Y., Rodrigues, T.K., Kato, N., Chuang, K.T. (2020). Prediction of network traffic load on high variability data based on distance correlation. 2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall), pp. 1-5. https://doi.org/10.1109/VTC2020-Fall49728.2020.9348769

[14] Abbasi, M., Shahraki, A., Taherkordi, A. (2021). Deep learning for network traffic monitoring and analysis (NTMA): A survey. Computer Communications, 170: 19-41. https://doi.org/10.1016/j.comcom.2021.01.021

[15] Ardalani-Farsa, M., Zolfaghari, S. (2010). Chaotic time series prediction with residual analysis method using hybrid Elman-NARX neural networks. Neurocomputing, 73(13-15): 2540-2553. https://doi.org/10.1016/j.neucom.2010.06.004

[16] Iqbal, M.F., Zahid, M., Habib, D., John, L.K. (2019). Efficient prediction of network traffic for real-time applications. Journal of Computer Networks and Communications, vol. 2019, Article ID 4067135. https://doi.org/10.1155/2019/4067135

[17] Yoon, M.S., Kamal, A.E., Zhu, Z. (2017). Adaptive data center activation with user request prediction. Computer Networks, 122: 191-204. https://doi.org/10.1016/j.comnet.2017.04.047

[18] Poupart, P., Chen, Z.T., Jaini, P., Fung, F., Susanto, H., Geng, Y.H., Chen, L., Chen, K., Jin, H. (2016). Online flow size prediction for improved network routing. 2016 IEEE 24th International Conference on Network Protocols (ICNP), pp. 1-6. https://doi.org/10.1109/ICNP.2016.7785324

[19] Parametric Technology Corporation. (1985). PTC Software India Pvt. Ltd., Pune, India. https://www.ptc.com/, accessed on Feb. 11, 2022.

[20] Dong, S. (2021). Multi class SVM algorithm with active learning for network traffic classification. Expert Systems with Applications, 176: 114885. https://doi.org/10.1016/j.eswa.2021.114885

[21] Patel, S., Gupta, A., Nikhil, Kumari, S., Singh, M., Sharma, V. (2018). Network traffic classification analysis using machine learning algorithms. 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), pp. 1182-1187. https://doi.org/10.1109/ICACCCN.2018.8748290

[22] Sanagavarapu, S., Sridhar, S. (2021). SDPredictNet-A Topology based SDN neural routing framework with traffic prediction analysis. 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), pp. 0264-0272, https://doi.org/10.1109/CCWC51732.2021.9376123

[23] Li, Y., Liu, H., Yang, W., Hu, D., Wang, X., Xu, W. (2016). Predicting inter-data-center network traffic using elephant flow and sublink information. In IEEE Transactions on Network and Service Management, 13(4): 782-792. https://doi.org/10.1109/TNSM.2016.2588500

[24] Sneha, Y.V., Vimitha, Vishwasini, Boloor, S., Adesh, N.D. (2020). Prediction of network congestion at router using machine learning technique. 2020 IEEE

International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER), pp. 188-193. https://doi.org/10.1109/DISCOVER50404.2020.9278028

[25] Yang, Y. (2021). A new network traffic prediction approach in software defined networks. Mobile Networks and Applications, 26 (2): 681-690. https://doi.org/10.1007/s11036-019-01413-5