MACHINE LEARNING BASED NETWORK TRAFFIC PREDICTIVE ANALYSIS

***Valla Shyam Kumar*[1]*., CH.Vennela*[2]*.,G.Alekhya*[3]*.,N.Likitha*[4]*.,T.Laharika*[5]**

1 Assistant Professor, Department Of CSE., Malla Reddy College Of Engineering For Women., Maisammaguda.,

Medchal.,Ts, India (vsmca15@gmail.com)

2, 3, 4, 5 B.Tech CSE, (19RG1A05D0, 19RG1A05E6, 19RG1A05G1, 19RG1A05H6),

Malla Reddy College Of Engineering For Women., Maisammaguda., Medchal., Ts, India

Abstract

If you know roughly how much traffic your network will need in the future, you may fine-tune the network's settings to optimize its performance. In order to achieve this goal, several analysis and traffic forecast methods are created with the use of machine learning techniques. In this research, we investigate the use of several online time series to forecast traffic patterns over a range of frame sizes. We start by describing the actual network traffic data that was collected and analysed for seasonal patterns and correlations. Second, we offer three machine learning techniques for predicting network data and compare them under different models and input features: linear regression, k closest neighbours, and random forest. Root-mean-squared percentage error (RMSPE) is used to measure how accurate a prediction is. We create three machine learning models where traffic is predicted for individual frame sizes using just that size's historical data, data for multiple frame sizes, and data for all frame sizes. Numerical studies on four datasets showed that linear regression was more accurate than the other two approaches. According to the findings, the algorithm's accuracy is improved by using previous data on all frame sizes to anticipate summary traffic of a specific frame size, but at the expense of longer execution times. This time cost may, however, be reduced with an almost imperceptible drop in accuracy by the strategic selection of input characteristics depending on seasonality.

Keywords:

The use of machine learning and an application-aware network to foretell traffic patterns.

introduction

Network operators may benefit from analysing and forecasting internet traffic in a number of contexts, especially given the industry's rapid expansion. They make sense when thinking about how to allocate resources during a network move or resize. Even if resources are limited, the most congested connections may be improved.Kaniapiskau, A., et al. kept up with in the first place. Proactive traffic routing and virtual topology adaption [14] is another potential use case for traffic analysis and prediction. Real-time or online models play a vital part in this process by constantly fine-tuning their forecasts based on observations of the current state of the network. The network is capable of rapidly reconfiguring itself in response to congestion or an unanticipated increase in traffic. Energy efficiency is an intriguing potential use for traffic prediction. In order to save energy, idle network connections or transceivers may be set to a low-power state for a time period of the user's choosing [4]. Different services and applications have varying needs, including those for latency, security, and reliability, all of which must be met by the network's optical layer. It is possible for multilayer application-aware networks to classify different optical traffic categories and then optimize for their needs [11, 12]. The resource allocation and planning procedures would benefit greatly by knowing the volume and

general trends of various forms of network traffic. However, the network operator may not have access to or be able to effectively process in real time information about the precise distribution of network traffic generated by various applications and services.

Connected Tasks

Several papers in the last five years have conducted in-depth analyses of network traffic analysis and prediction, either as a standalone paper [9] or as a chapter of more comprehensive surveys on machine learning in optical networks [3, 8, 14, 17]. These machine learning algorithms may be used for a wide range of tasks, and can be broken down into two groups: supervised and unsupervised learning [14]. Contrarily, in supervised learning, algorithms are aware of predicted outcomes throughout the training phase and may be used for tasks like traffic forecasting using historical data, estimating transmission quality [13] or routing [19]. However, in unsupervised learning, such as in traffic anomaly detection [5] or attack detection [6], the expected results are not known beforehand, and it is used to find patterns (similarities) and structures in the traffic or to extract features.

Many different approaches to traffic prediction can be found in the literature, with the vast majority employing either autoregressive moving average

(ARIMA) or long short-term memory (LSTM) recurrent neural networks [9]. However, recent evidence has pointed to the limitations of pure time-series forecasting approaches, and the results of non-time-series forecasting methods are highly dependent on the datasets used [3]. As a result, we need novel strategies, both in terms of processing complexity and precision. Using data analysis techniques to provide new input characteristics for the algorithms is an intriguing option to boost the forecast accuracy beyond only the quantity of network traffic between two timestamps. In [15], the authors enhance the performance of an LSTM-based traffic prediction model by using an autocorrelation coefficient. In this approach, the time series' seasonal information may be preserved. In addition, three data-driven LSTM approaches are presented in [10], where the utilization of daily and weekly seasonal trends is also investigated.

Analysing the Data

Multiple real-world datasets were used for analysis and additional testing. The initial data collection is a four-week snapshot of activity at the Seattle Internet Exchange Point (SIX), beginning on November 1, 2020, and ending on November 28, 2020. The second data collection includes information from SIX for the next four weeks, from November 28th, 2020, through December 26th, 2020. Both sets of data are sampled every 5 minutes. The third dataset was produced by resampling the November SIX dataset using a 1-hour maximum aggregation. The resulting dataset covered the same time frame as the original, albeit being much smaller. Raw 5-minute sample results are available throughout the whole examined period thanks to weekly data collection from the SIX website1. The aggregated traffic in bits and the frame size distribution databases both come from the RRD format and are utilized here. The databases' numeric values were retrieved with the help of RRDtool2. The raw information consists of thirteen distinct frame size bands. In this study, for the sake of brevity, they are represented by alphabetic symbols, as indicated in table 1. Figure 1 shows a weekly dispersion of frame sizes from the collected input data.

Table 1: Frame sizes - letter representation Frame size in bytes

Frame size in bytes	64	65-128	129-256	257-384	385-512	513-640	641-768	769-896	897-1024	1025-1152	1153-1280	1281-1408	1409-1536
letter representation	a	b	c	d	e	f	g	h	i	j	k	l	m

The bit value of traffic in different frame sizes was calculated from the collected databases as follows. Let n be the total number of frames in a given time point, xi the size of a frame of it size in bits, Yi the percentage of frames of i-th size divided by 100, s the aggregate traffic in given time point in bits per second. From the data, we know the values of xi , yi and s in a given time point. The aggregated traffic s can be expressed as

$$s = \sum_{i=1}^{13} x_i \cdot y_i \cdot n.$$

This makes it simple to ascertain both the total number of frames, n, and the volume of traffic in frames of size i. The estimated volume of traffic for the various window widths during the course of the studied four-week period is shown in Fig. 2. As can be seen, that's where the great majority of visitors go.

A Look at the Top 4 Models and Algorithms

We undertake data analysis and use the results to offer three models for predicting network traffic in varying frame sizes. That is, model 1 predicts traffic in frame x based on traffic in all frame sizes, model 2 predicts traffic in frame x based on traffic in three less correlated frame sizes, and model 3 predicts traffic in frame x based on traffic in only frame x. Additional input features were selected because of the autocorrelations' demonstration of seasonality in the data, allowing the models to make more accurate predictions. This implies that in addition to the volume of traffic at the moment under consideration, we also provide indicators of the volume of traffic at key times in the past, such as five minutes, twenty-four hours, and seven days before. In Section 5 we go into further detail on how to choose between available optional extras. After developing models and gathering supplementary data, the next step is to settle on a set of machine learning (ML) methods. Given that the time period we are trying to predict is just 5 minutes long, we can only consider regressors that can be trained and generate predictions quickly. Although techniques such as

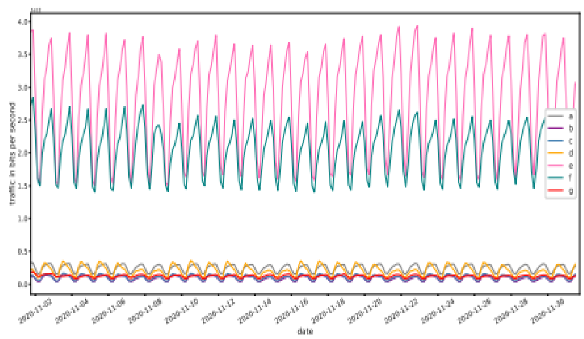

Fig. 1: Internet traffic in different frame sizes in the European dataset

learning tend to be very accurate, but they need a lot of time to train, which makes them impractical in our case. After experimenting with a number of techniques, such as Support Vector Machines and AdaBoost, we settled on the three regressors shown below due to their respective ease of use and speed. Trying to fit a linear model to the connection between observed linear data, linear regression (LR) is a straightforward method. The ideal generalization would minimize the inaccuracy in making predictions on untested facts. For instance, in [16], this method was utilized to create a self-updating model for seasonal prediction of various time series. The ease with which this algorithm may be trained and made predictions is a major factor in its favor. The k-closest-neighbors (kNN) approach takes an input data point and predicts its output by comparing it to that of its nearest k neighbours. Since the values of new data points are predicted solely by checking the most similar (nearest) ones, this algorithm can handle non-linearity in the data well. For instance, in [1], a multiple time series model was constructed with extra characteristics distinguishing between weekdays and weekends, since these two time periods exhibit distinct seasonality patterns. One of the primary benefits of this algorithm is the speed that comes with its relative ease of usage. Decision-tree-based ensemble methods like random forest (RF) are one popular option. To increase accuracy and minimize over-fitting, the outcomes of many decision trees that employ different subsets of characteristics are averaged. Here we have an instance of integrating many weak models into a single robust one. Successful use of this method to build a prediction model for multiple time series with additional features, such as correlations between the considered time series', can be found, for example, in [18]. Among the algorithms studied in [18], RF was found to produce the most reliable forecasts. This technique may be time and resource intensive when used to huge datasets; however, in our instance, the number of features is limited, thus only a few trees are needed for the prediction.

Computer simulations

Scikit-learn is the ML algorithm implementation used in this project. The optimized values for these parameters, determined using grid search, are shown in Table 2. All of the techniques use discrete data points to represent the epoch-specific network activity.

Table 2: Tuning chosen parameters for the algorithm

Algorithm	Parameter	Tested values	Chosen value
kNN	weights	'uniform', 'distnace'	'uniform'
	n_neighbours	1, 3, 5, 8, 10	8
RF	n_estimators	3, 5, 10, 15, 20, 50	10

0 A proper error measure is required to assess and compare the predictions of various algorithms. The absolute error metrics vary greatly between frame sizes because of the vast differences in traffic volume. Percentage error is the most intuitive measure to use when comparing the efficacy of different regressors across different frame sizes. That's why we choose to measure success using the root-mean-squared percentage error (RMSPE). The optimal model was determined after a battery of trials. Considering both prediction quality and timing is crucial for developing an online traffic prediction model that can adapt rapidly to changing network circumstances. As a result, we make use of several supplementary input characteristics, which we introduce in Section 4. Five minutes ago (previous timestamp), an hour ago (anchorage), twenty-four hours ago (yesterday), and a week ago (last week) are the four additional attributes we provide initially to show the volume of visitors. We can train the models' import by hand with the use of supplemental features.

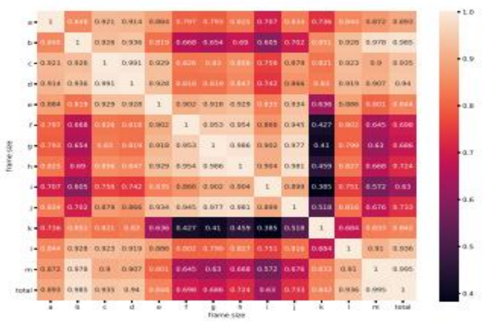

Fig. 2: Correlations between traffic in different frame sizes and aggregate traffic similar datapoints, which makes it possible not to consider them in order. For that reason, in all the models and algorithms we use 10-fold cross validation.

Table 3: RMSPE comparison in different models and algorithms, SIX November dataset, 5-minute sampling; best model for each algorithm highlight.

Alg.	Frame size												
	a	b	c	d	e	f	g	h	i	j	k	l	m
Model 1													
LR	0.0126	0.0094	0.0115	0.0188	0.0466	0.0444	0.0274	0.0261	0.0781	0.0383	0.0165	0.0163	0.0107
kNN	0.0305	0.0179	0.0364	0.0409	0.0627	0.0566	0.0493	0.0500	0.0952	0.0685	0.0479	0.0267	0.0138
RF	0.0160	0.0128	0.0149	0.0209	0.0580	0.0519	0.0295	0.0284	0.0872	0.0405	0.0194	0.0198	0.0141
Model 2													
LR	0.0148	0.0123	0.0161	0.0231	0.0517	0.0479	0.0325	0.0319	0.0866	0.0443	0.0199	0.0171	0.0136
kNN	0.0302	0.0141	0.0185	0.0287	0.0706	0.0692	0.0619	0.0619	0.1071	0.0794	0.0196	0.0182	0.0143
RF	0.0166	0.0132	0.0163	0.0233	0.0648	0.0494	0.0342	0.0338	0.0867	0.0478	0.0207	0.0213	0.0148
Model 3													
LR	0.0164	0.0131	0.0184	0.0253	0.0555	0.0491	0.0354	0.0351	0.0898	0.0478	0.0225	0.0177	0.0142
kNN	0.0164	0.0131	0.0167	0.0245	0.0579	0.0489	0.0333	0.0337	0.0860	0.0434	0.0212	0.0185	0.0143
RF	0.0175	0.0137	0.0187	0.0261	0.0630	0.0499	0.0353	0.0357	0.0937	0.0489	0.0231	0.0205	0.0152

In Table 3, we present the RMSPE values for three models described in section 4, for the SIX November dataset with 5-minute sampling. As can be concluded, the choice of the model depends on the choice of the regressor - LR and RF get their lowest RMSPE values in model 1, and in - in model 3. However, the overall lowest RMSPE values are obtained by LR. As an illustration, in Fig. 6, we present the comparison of the prediction results between model 1 and 3 obtained by LR for a sample frame size. Some differences between the models can be spotted in the presented zoomed-in fragment, showing that the predict

Knapińska et al.

Table 4: RMSPE comparison in different models and algorithms, SIX December dataset, 5-minute sampling; best model for each algorithm highlighted

Alg.	Frame size												
	a	b	c	d	e	f	g	h	i	j	k	l	m
Model 1													
LR	0.0151	0.0110	0.0109	0.0177	0.0336	0.0304	0.0247	0.0258	0.0609	0.0329	0.0155	0.0182	0.0111
kNN	0.0284	0.0181	0.0368	0.0353	0.0501	0.0532	0.0483	0.0485	0.0841	0.0629	0.0448	0.0287	0.0135
RF	0.0180	0.0146	0.0152	0.0206	0.0379	0.0347	0.0268	0.0276	0.0718	0.0356	0.0189	0.0213	0.0144
Model 2													
LR	0.0153	0.0140	0.0129	0.0205	0.0374	0.0343	0.0312	0.0321	0.0713	0.0407	0.0174	0.0191	0.0137
kNN	0.0317	0.0150	0.0206	0.0238	0.0606	0.0669	0.0649	0.0663	0.0999	0.0774	0.0186	0.0191	0.0142
RF	0.0185	0.0155	0.0158	0.0226	0.0414	0.0390	0.0323	0.0332	0.0741	0.0419	0.0192	0.0210	0.0148
Model 3													
LR	0.0183	0.0152	0.0180	0.0237	0.0411	0.0372	0.0341	0.0352	0.0731	0.0440	0.0215	0.0198	0.0145
kNN	0.0184	0.0148	0.0166	0.0229	0.0414	0.0366	0.0318	0.0328	0.0717	0.0407	0.0208	0.0211	0.0144
RF	0.0196	0.0158	0.0179	0.0247	0.0433	0.0394	0.0345	0.0348	0.0750	0.0439	0.0217	0.0206	0.0152

Table 5: RMSPE comparison in different models and algorithms, SIX November dataset, 1-hour maximum aggregation; best model for each algorithm highlighted

Alg.	Frame size												
	a	b	c	d	e	f	g	h	i	j	k	l	m
Model 1													
LR	0.0236	0.0205	0.0254	0.0315	0.1219	0.0918	0.0424	0.0384	0.1411	0.0706	0.0315	0.0427	0.0225
kNN	0.0598	0.0379	0.1064	0.0996	0.1491	0.1543	0.1437	0.1338	0.2084	0.1986	0.1364	0.0768	0.0406
RF	0.0299	0.0286	0.0344	0.0381	0.1345	0.1347	0.0491	0.0429	0.1450	0.0788	0.0475	0.0556	0.0334
Model 2													
LR	0.0249	0.0350	0.0579	0.0635	0.1349	0.1394	0.0643	0.0652	0.1492	0.0909	0.0746	0.0516	0.0394
kNN	0.0477	0.0315	0.0523	0.0647	0.1598	0.1716	0.1580	0.1551	0.2247	0.2325	0.0562	0.0375	0.0343
RF	0.0290	0.0337	0.0490	0.0592	0.1119	0.1289	0.0711	0.0679	0.1714	0.0966	0.0666	0.0355	0.0363
Model 3													
LR	0.0411	0.0380	0.0679	0.0688	0.1539	0.1189	0.0816	0.0797	0.1754	0.1084	0.0675	0.0536	0.0517
kNN	0.0419	0.0342	0.0515	0.0599	0.1521	0.1270	0.0781	0.0720	0.1629	0.0966	0.0596	0.0449	0.0452
RF	0.0457	0.0356	0.0452	0.0585	0.1500	0.1477	0.0804	0.0709	0.1793	0.0926	0.0603	0.0541	0.0462

based on the historical traffic from all the frame sizes rather than a single one is generally closer to the real values.

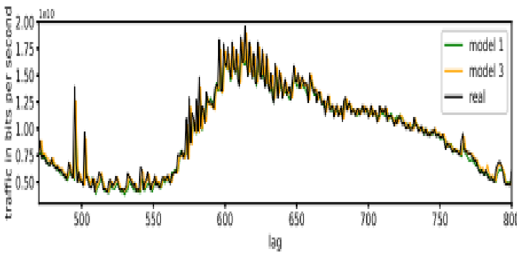

Fig. 3: Prediction results for traffic in frames of size I, LR regressor, SIX November dataset, 5-minute sampling - zoomed-in fragment The same trends were observed in all the remaining datasets.

Table 4 presents the results for the SIX December dataset, while Table 5 presents for the SIX November dataset with 1h aggregation, with additional features anchorage, yesterday and last week. In Table 6, we present the RMSPE values obtained for the European dataset. Because of the 3-hour average aggregation, after calculating the autocorrelation function values, we decided to use the following additional input features: the amount of traffic 24 hours before (yesterday), 48 hours before (two_days_ago) and 1 week before (last week). As can be seen, similarly.to the other datasets, the prediction quality is higher taking into account the historical traffic in all the frame sizes simultaneously, with LR being the most accurate among considered regressors.

Table 6: RMSPE comparison in different models, European dataset, 3-hour average aggregation; best model for each algorithm highlighted.

Algorithm	Frame size						
	a	b	c	d	e	f	g
Model 1							
LR	0.0156	0.0225	0.0193	0.0247	0.0139	0.0165	0.0306
kNN	0.0262	0.1084	0.0895	0.1220	0.0351	0.0281	0.0332
RF	0.0202	0.0333	0.0299	0.0314	0.0250	0.0257	0.0345
Model 2							
LR	0.0236	0.0369	0.0331	0.0386	0.0276	0.0287	0.0350
kNN	0.0323	0.1275	0.1075	0.1283	0.0357	0.0324	0.0487
RF	0.0296	0.0421	0.0411	0.0378	0.0331	0.0340	0.0384
Model 3							
LR	0.0300	0.0441	0.0422	0.0370	0.0343	0.0345	0.0422
kNN	0.0307	0.0760	0.0591	0.0618	0.0359	0.0350	0.0398
RF	0.0318	0.0460	0.0440	0.0372	0.0365	0.0382	0.0410

In Table 7, we present the mean percentage advantage of the RMSPE values obtained by the best regressor, LR, in model 1 over model 3 for all considered datasets. As can be concluded, the prediction of the amount of traffic in a specific frame size is better considering the historical data of the traffic for all the frame sizes when compared to the prediction based only on one frame size.

Table 7: Mean percentage advantage of RMSPE of model 1 over model 3 for the LR regressor in considered datasets

Dataset	Frame size												
	a	b	c	d	e	f	g	h	i	j	k	l	m
SIX November, 5min sampling	22%	28%	36%	25%	16%	10%	22%	25%	12%	18%	25%	9%	24%
SIX November, 1h aggregation	43%	46%	63%	54%	21%	23%	48%	52%	20%	35%	53%	20%	56%
SIX December, 5min sampling	18%	26%	38%	24%	18%	18%	26%	25%	17%	24%	27%	9%	22%
European, 3h aggregation	48%	49%	54%	33%	60%	52%	27%	-	-	-	-	-	-

Table 8: Time of execution in seconds, SIX November dataset, 5-minute sampling

Algorithm	Model 1	Model 2	Model 3	Model 1a	Model 1b
LR	0.0084	0.0043	0.0033	0.0095	0.0062
kNN	0.0379	0.0198	0.0155	0.0485	0.0277
RF	0.3931	0.1370	0.0588	0.4790	0.2880

Table 8 presents average time of execution for considered regressors for tested models (note, model 1a and 1b are described further). The measurements were performed on a machine with an Intel Core i5-1038NG7 processor with 16 GB RAM. As can be observed, all the algorithms are the fastest in model 3 because of the smallest dataset size. Nevertheless, at this stage, the prediction quality is more important than the time of execution, especially considering the very short time of execution for the best regressor - LR. For that reason, we chose model 1 for further analysis.

Conclusions

In this paper, we focus on the prediction and analysis of network traffic composed of various frame sizes. In more detail, the developed model is able to forecast traffic of a certain type based on the historical data. Firstly, we described gathered real network traffic data and its preparation process required for featureaugment extraction and further analysis. After that, we detected seasonality patterns by calculating autocorrelations for different lag values showing similar patterns for different frame sizes. Moreover, the correlations between different traffic types were investigated, indicating similarities between traffic patterns in certain frame sizes. Next, we proposed three machine learning algorithms and ran extensive numerical experiments on four datasets to evaluate their efficiency. According to the results, linear regression yields the highest accuracy having its RMSPE values on average 50% lower than kNN and 15% lower than random forest. Additionally, we investigated the impact of different models and input features choices, finding the best compromise between prediction quality and time of execution.

References

[1]. Al-Qahtani, F.H., Crone, S.F.: Multivariate k-nearest neighbour regression for time series data—a novel algorithm for forecasting uk electricity demand. In: The 2013 international joint conference on neural networks (IJCNN). pp. 1–8. IEEE (2013)

[2]. Biernacki, A., Krieger, U.R.: Session level analysis of p2p television traces. In: International Workshop on Future Multimedia Netw. pp. 157–166. Springer

[3] Boutaba, R., Salahuddin, M.A., Limam, N., Ayoubi, S., Shahriar, N., EstradaSolano, F., Caicedo, O.M.: A comprehensive survey on machine learning for networking: evolution, applications and research opportunities. Journal of Internet Services and Applications 9(1), 16 (2018)

[4]. Cenedese, A., Tramarin, F., Vitturi, S.: An energy efficient ethernet strategy based on traffic prediction and shaping. IEEE Trans. on Commun. 65(1), 270–282 (2016)

[5]. Chen, X., Li, B., Shamsabardeh, M., Proietti, R., Zhu, Z., Yoo, S.J.B.: On realtime and self-taught anomaly detection in optical networks using hybrid unsupervised/supervised learning. In: 2018 European Conference on Optical Communication (ECOC). pp. 1–3 (2018)

[6]. Furdek, M., Natalino, C., Lipp, F., Hock, D., Giglio, A.D., Schiano, M.: Machine learning for optical network security monitoring: A practical perspective. Journal of Lightwave Technology 38(11), 2860–2871 (2020)

[7]. Garsva, E., Paulauskas, N., Grazulevicius, G.: Packet size distribution tendencies in computer network flows. In: 2015 Open Conference of Electrical, Electronic and Information Sciences (eStream). pp. 1–6. IEEE (2015)

[8]. Gu, R., Yang, Z., Ji, Y.: Machine learning for intelligent optical networks: A comprehensive survey. J. of Network and Computer Applications 157, 102576 (2020)

[9]. Joshi, M., Hadi, T.H.: A review of network traffic analysis and prediction techniques. arXiv preprint arXiv:1507.05722 (2015)

[10]. Krishnaswamy, N., Kiran, M., Singh, K., Mohammed, B.: Data-driven learning to predict wan network traffic. In: Proceedings of the 3rd International Workshop on Systems and Network Telemetry and Analytics. pp. 11–18 (2020)

[11]. Lehman, T., Yang, X., Ghani, N., Gu, F., Guok, C., Monga, I., Tierney, B.: Multilayer networks: an architecture framework. IEEE Communications Magazine 49(5), 122–130 (2011)

[12]. Lopez, V., Konidis, D., Siracusa, D., Rozic, C., Tomkos, I., Fernandez-Palacios, J.P.: On the benefits of multilayer optimization and application awareness. Journal of Lightwave Technology 35(6), 1274–1279 (2017)

[13]. Mata, J., de Miguel, I., Durán, R.J., Aguado, J.C., Merayo, N., Ruiz, L., Fernández, P., Lorenzo, R.M., Abril, E.J.: A SVM approach for lightpath QoT estimation in optical transport networks. In: 2017 IEEE International Conference on Big Data (Big Data). pp. 4795–4797 (2017)

[14]. Musumeci, F., Rottondi, C., Nag, A., Macaluso, I., Zibar, D., Ruffini, M., Tornatore, M.: An overview on application of machine learning techniques in optical networks. IEEE Communications Surveys & Tutorials 21(2), 1383–1408 (2019)

[15]. Shihao, W., Qinzheng, Z., Han, Y., Qianmu, L., Yong, Q.: A network traffic prediction method based on lstm. ZTE Communications 17(2), 19–25 (2019)

[16]. Wagner, N., Michalewicz, Z., Schellenberg, S., Chiriac, C., Mohais, A.: Intelligent techniques for forecasting multiple time series in real-world systems. International Journal of Intelligent Computing and Cybernetics (2011)

[17]. Xie, J., Yu, F.R., Huang, T., Xie, R., Liu, J., Wang, C., Liu, Y.: A survey of machine learning techniques applied to software defined networking (sdn): Research issues and challenges. IEEE Communications Surveys & Tutorials 21(1), 393–430 (2018)

[18]. Zagorecki, A.: Prediction of methane outbreaks in coal mines from multivariate time series using random forest. In: Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, pp. 494–500. Springer (2015)

[19]. Zhong, Z., Hua, N., Yuan, Z., Li, Y., Zheng, X.: Routing without routing algorithms: An ai-based routing paradigm for multi-domain optical networks. In: 2019 Optical Fiber Communications Conference and Exhibition (OFC). pp. 1–3 (2019).