

A Performance Analysis of Supervised Learning Classifiers for QoT Estimation in ROADM-based Networks

Alan A. Díaz-Montiel and Marco Ruffini

Connect Centre, Trinity College Dublin, Ireland
{adiazmon,marco.ruffini}@tcd.ie

Abstract. Machine learning techniques for optimization purposes in the optical domain have been reviewed extensively in recent years. While several studies are pointing in the right direction towards building enhanced transport network control systems including estimation algorithms, the physical effects encountered in the optical domain raise several challenges that are hard to learn from and mitigate. In this paper, we provide a performance analysis of various supervised learning algorithms when predicting the Quality of Transmission (QoT), in terms of signal to noise ratio (OSNR), of lightpaths when erbium doped fiber amplifier (EDFA) power excursions and fiber nonlinearities are taken into account. The analysis considers F1-scores and computational training times as the main comparison metrics. A customized optical data network simulator was used for the generation of synthetic labeled data samples. Our results depict similar performance among groups of classifiers, and a correlation between the data sample size and the prediction accuracy.

Keywords: Supervised Learning · Transport Networks

1 Introduction

Transport metro-access networks need fast and dynamic reconfiguration procedures to overcome the challenges imposed by 5G-service driven requirements [1]. While several advancements at the hardware level (e.g., higher capacity transceivers) are being achieved, the disaggregation between such components and their control systems remains a challenge [2]. With the emergence of the software-defined networking (SDN) paradigm a decade ago, multiple consolidated projects have tackled the openness and programmable features of optical network elements (i.e., switches) [3, 4]. Due to the heterogeneous and complex nature of equipment at deployment sites today, the development of robust protocols and standards is ongoing but is a lengthy process. In the meantime, the community has been evaluating different control system strategies, which have resulted in a set of recommendations well documented in the ITU-T G.7702 [5]. More recently, the application of artificial intelligence (AI) for multiple use cases in the optical domain has been widely evaluated. The latter has been surveyed

by Mata et al., in [6]. Particular attention is being paid to the utilization of estimation/classification algorithms that could predict the performance of signals traversing networks at various domains and configuration settings (e.g., heterogeneous equipment).

Commercial optical networks are today rather static, and their evolution towards a fully flexible and dynamic system will face control and management challenges. For instance, state-of-the-art wavelength division multiplexed (WDM) networks consider reconfigurable add/drop multiplexers (ROADMs), built up with colorless wavelength selective switches (WSS) and optical cross-connects (OXC) as the de facto switching and routing networking elements, enhancing higher levels of flexibility. Additionally, in order to reach longer distances with fewer resources, the consideration of automatic gain controlled (AGC)-EDFA systems is the default approach. However optical reconfigurations are still carried out slowly (e.g., in terms of several minutes), mainly due to the effects of power excursions in gain-controlled optical amplifiers [7, 8].

Hence, the development of tools capable of handling such impairments at low cost without compromising the current performance is of high interest. Thus, the inclusion of ML-based classification algorithms in the control system of optical networks has been a promising area of research in recent years. For instance, Morais and Pedro [9] reviewed the models of K-nearest neighbors (KNN), logistic regression (LR), support vector machines (SVM), and artificial neural networks (ANN), analysing their performance in terms of predicting the QoT of unestablished lighpaths using simulated data. Their results suggest that ANN outperforms the other models with 99.9% accuracy. On a different study [10], Mata et al., evaluated the models of SVM, LR, classification and regression trees (CART), bagging trees (TREEBAG) and random forests (RF), comparing them in terms of estimation performance and prediction time. In their results, SVM, RF and TREEBAG achieved an accuracy performance up to 99.9%, while keeping their prediction time low. While these studies suggest that the use cases associated with the optical domain may be successfully addressed with ML, we believe further research is required that considers extra physical layer phenomena in the training of the ML-models, including EDFA power excursions and non-linear physical impairments.

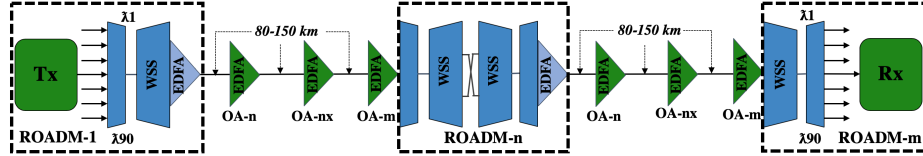


Fig. 1. Linear topology. Deployment may vary in different network settings as per features in Table 1.

In this study, we analyze the performance of various classification algorithms, when considering both topological features (e.g., number of nodes, fibre distance,

amplifier span, etc.) and wavelength load [11] (e.g., the active wavelengths in the system before the new test channel is added). Indeed, as shown in [8], the wavelength load configuration is one of the main parameters affecting power excursions in EDFAs. With regard to the comparison of the ML models, two crucial metrics are considered: the accuracy at classifying different classes of traffic (based on OSNR levels), and the computational time required to train (build) the models, together with the size of the data samples. For the execution of the experiments, we used the Scikit-learn: Machine Learning in Python software [12], which provides powerful APIs for ML classification models. Then, for the data generation, we used the Optical-MAN simulator first introduced in [13], which allows for the creation of large scale optical networks deployed with WSS-based ROADMs and EDFA-amplified links, and the simultaneous transmission of 90 channels in the C-band spectrum (1529.6 nm - 1565.2 nm).

The remainder of the paper is structured as follows. In section 2, we introduce the system setup and the data generation process with our customized tools. Then, in section 3 we briefly describe the classification algorithms used for the performance analysis. The experiments and results are discussed in section 4. Finally, we present our conclusions in section 5.

2 System setup and data generation

2.1 System setup

Experimentation procedures with ML-models typically require large data sets in order to achieve good performance when tackling complex systems. In the optical domain, common network-QoT metrics for supervised learning models are the OSNR and bit error rate (BER) of the transmission systems. Hence, in order to collect large amount of samples from a physical testbed, constant monitoring procedures are needed at least at one point of the network. Alas, the collection of physical layer data in the optical domain (e.g., signal power levels) is an expensive task, mainly due to the high costs and complexities of the monitoring equipment. In addition, the deployment of large-scale, topology-variable optical networks is an infeasible task for many of the academic laboratories today. Because of that, the exploration of optical networks via simulation tools has been used in several cases [9, 14-16]. Although these virtual environments enable the low cost, fast generation of physical layer optical data, the experimentation tools (ML-models) would always require further trial with *real* testbed data, which would account for their validation.

For the purposes of this study, we used the customized Optical-MAN simulator first introduced in [13], which allows to deploy WDM network topologies and collect synthetic labeled data to train ML models, as we recently presented in [11] for the analysis of support vector machines (SVM). Our tool simulates WDM networks considering: optical nodes consisting of ROADMs equipped with WSSs, optically-amplified AGC-EDFA links, and also AGC-EDFAs for pre-/post-signal amplification, together with fiber non-linear effects such as stimulated Raman scattering. An abstracted representation of a linear network topology composed

of the network elements described is presented in Fig. 1. End-to-end transmission is enabled for up to 90 wavelength channels in the C-band (1529.6 nm - 1565.2 nm). For the EDFAs in our system it is assumed gain flattening filtering (GFF) is used for gain equalization. While accounting for GFFs at line-amp sites covers for the smoothing of unequal signal intensities, a residual wavelength-dependent gain (WDG) still remains after the filtering due to imperfect equalization. The latter, may result in the detrimental performance of the transmitted signals. Parting from the models proposed by Junio and Kilper in [8], we take under consideration the power excursions, enabling the analysis of the effects these cause in the system, and how ML-models could learn from these.

For all the simulated network scenarios, we assume optical performance monitoring is implemented at the receiver end, which enables for the tracking of the signal performance in terms of signal power and noise. Then, as first introduced in [13], we compute the OSNR levels of all the individual signals being transmitted.

2.2 Data generation

Our goal is to analyze the performance of multiple supervised learning classifiers at predicting the QoT of newly established lightpaths in a network with a given topology and wavelength load configuration.

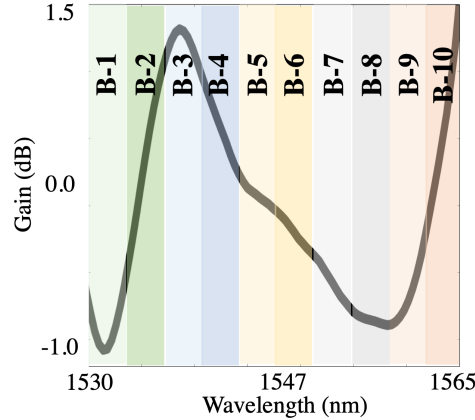


Fig. 2. Segmented spectrum of EDFA wavelength-dependent gain.

Our approach consists in extending the system parameters used to train the ML models, to not only consider network topological settings (i.e., number of nodes, number of links, etc.) as it is the common case, but to also include the number of active wavelength channels (WCs) and their position in the spectrum before installing a new WC. For the latter, we introduce the concept of a 1×10 data-array, which represents 10 segments of the transmission spectrum (i.e., the

C-band in our study). In a 90-WC WDM network, each slot (bin) of the 1x10 data-array would correspond to a number of active-WCs in the range 1-10. A graphic representation of this concept is depicted in Fig. 2.

We further exploit the capabilities of the classifiers by attempting to predict the performance of newly established WCs in a network considering WL-scenarios for which we do not train the ML models. That is, in a 90-WC WDM system, for the training data sets we only consider WL-scenarios for n active-WCs so that $\{n|n \in \{1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70\}\}$. Subsequently, we would register the labeled OSNR levels of the $(n+1)$ -WC to be installed. On the opposite, for the testing data set we consider WL-scenarios where n can take any value so that $\{n|n \in \{1 - 90\}\}$. Hence, the classifiers would attempt to predict the OSNR levels of the $(n+1)$ -WC, which it may have not been trained for before.

The input labeled data used to train our models consisted in: the wavelength (measure in nm) of the monitored WC, 5 topology configuration settings (depicted in Table 1), and the 1x10 segmented spectrum data-array described above, accounting for 16 parameters in total. The data generation process with the Optical-MAN simulator consist in the following steps:

- (i) Set a linear WDM network topology and configuration settings by randomly selecting a combination of values from Table 1.
- (ii) Randomly select n -WCs to add to the network as per WL case, and add them to the network.
- (iii) Randomly select another $(n+1)$ -WC and add it to the network.
- (iv) For the $(n+1)$ -WC, monitor power and noise levels, compute the OSNR levels, and generate a register (labeled sample).

Table 1. Topology configuration parameters.

Feature	Value
Number of ROADMs	2 to 8
Number of fiber spans	2 to 6
Length of fiber span	80 to 150 km
Launch channel power	-10 to 4 dBm
EDFA Preamp gain	Fiber compensation adjustment dB

Through this sequence of steps we can study the effects of the various topological and network configuration settings perceived on the transmitted signals. Also, because the deployed networks may be significantly different from one another, the generated data used to train the ML classifiers enable the performance analysis of such estimation tools in generic use cases. In our case, we attempt for the classifiers to learn about the physical effects resulting of the interaction of the WCs (i.e., power excursions) and the correlation that exists between the number of WCs active in the system, their position in the signal spectrum, and the transmission performance.

Furthermore, we were interested in studying the scalability performance of the ML classifiers. For this we used six sets of independent data samples of different size. In total, we generated 294,343 data samples, which we split as 22.7 K, 42.3 K, 45 K, 51 K, 57.7 K, 75.6 K, for the six sets, respectively. Also, we balanced the input data in order to mitigate misperformance of the classifiers.

3 Classification algorithms and optimization techniques

This section describes the classification algorithms analyzed in this study. Only supervised learning-based algorithms were used for the development of this study. We have reviewed 11 algorithms, which we have split in two classes: normal and ensemble-based. The normal class is composed by K-Near Neighbors (K-NN), Linear-Support Vector Machine (L-SVM), Radial Basis Function SVM (RBF-SVM), Logistic Regression (LR), Decision Tree (DT), Artificial Neural Network (ANN), Naive Bayes (NB), and Linear Discriminant Analysis (LDA). The ensemble-based class is composed by Random Forest, Ada Boost, and Bagging. We have used the Scikit-learn: Machine Learning in Python API [12] for the development of the experiments. We briefly describe the main components of each algorithm and their functionality, for details of the models we redirect to the documentation.

3.1 Normal Classifiers

K-Near Neighbors this algorithm attempts to classify values in multiple classes by clustering the data and comparing individual values to their K number of immediate neighbours.

Time complexity: $O(kn^2)$, for k nearest neighbors.

Linear-Support Vector Machine this algorithm attempts to learn from the input data by categorising and building different classes (*hyperplanes*), which are subsequently used for future classification. It relies on regularization parameters C and γ to avoid misclassification and correlation of individual points, respectively. It also transforms the problem with linear algebra - kernel functions.

Time complexity: $O(n_{features}n_{samples}^2)$, with linear kernel function.

Radial Basis Function SVM Same as before.

Time complexity: $O(n_{features}n_{samples}^2)$, with radial basis kernel function.

Logistic Regression this algorithm utilises a logistic function to build a model of dependent variables, more commonly used for binary scenarios. However, by applying optimization algorithms it is possible to use logistic regression for classifying multiple classes. Our implementation considers the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) optimization algorithm.

Time complexity: $O(n^2)$, with L-BFGS.

Decision Tree this algorithm consists in identifying relations between the input parameters and their output values building comparison points (tree branches), to subsequently perform as a binary tree.

Time complexity: $O(n_{features}n_{samples}\log(n_{samples}))$.

Artificial Neural Network this algorithm is based on a multilayer perceptron, which learns correlations between inputs and outputs and generates ‘weights’ for future inputs that are used for minimizing the error of classification.

Time complexity: $O(nmh^koi)$, for n training samples, m features, h neurons per layer, k hidden layers, o output neurons, and i number of iterations.

Naive Bayes this algorithm considers the assumption of conditional independence between the multiple features given the values of the various classes. Our implementation of this algorithm considered the likelihood of the sample features to be Gaussian.

Time complexity: $O(nK)$, for K number of classes.

Linear Discriminant Analysis this algorithm operates by finding linear correlations of the input features.

Time complexity: $O(n_{features}^2n_{samples})$.

3.2 Ensemble Classifiers

These types of classifiers are the result of the combination of various classifiers altogether. These can be understood as optimization techniques for algorithms that do not perform well for given use cases, such as overfitting the data or performing weakly due to the lack of data. For the purposes of this study, we reviewed the ensemble classifiers of random forests, boosting and bagging.

Random Forest this ensemble learning method for classification consists in the combination of multiple decision trees at training time, and operates the classification of various classes by selecting the classification/prediction mode among the decision trees.

Ada Boost this algorithm, adaptive boosting (AdaBoost), operates by combining multiple “weak” classifiers in order to create a much stronger/accurate tool.

Bagging this algorithm, bootstrap aggregation (bagging), operates by bootstrapping (random sampling with replacement) multiple models in parallel, and come up with hypothesis of more accurate classifications, making a decision based on the most accurate hypothesis.

4 Experiments and results

In order to speed up the execution time, our experiments were run in a Linux x86_64 server with 10 Intel(R) Xeon(R) CPU E5-2699 v4 @ 2.20 GHz processors. We used the Scikit-learn: Machine Learning in Python software [12] because it allows for a simplistic and variable implementation of the various classification algorithms. For instance, it enables the declaration of multiple parameters for training the models. Additionally, it enables a dynamic search of the best possible parameters to train specific models for given use cases with the *GridSearchCV* function. Thus, after a wide search of such parameters for our use case, we found those that enhanced a mean tolerable prediction accuracy among the various classifiers, which are depicted in Table 2.

The visual representation of the results of the normal classifiers are depicted in Fig. 3(a) and 3(b), showing the training computational time and the F1-score against the multiple sample sizes, respectively. We can demonstrate the trade-off between high classification accuracy and high training computational time, which is a well-known feature among these type of classifiers.

However, the novelty of this analysis focuses on the ability for these statistical tools to learn from our selected optical domain features (i.e., active wavelength load) and the topological configurations (i.e., number of nodes and fiber spans) together. Taken these under consideration, the comparison shows that some algorithms perform better than the others, in some cases achieving up to 90% score. While in the literature we can find similar work achieving higher prediction accuracy [9], [15], [16], our analysis considers EDFA power excursions and nonlinearities (SRS) that introduce power dynamics that are harder to predict. Thus, we believe the results can be further improved in the future by better depicting the physical layer parameters for training the algorithms.

In addition, there are two main patterns we can identify in Fig. 3(a). *i)* Contrary to an expected passive incremental accuracy performance as the volume of the sample size increases, our results show a decreased performance in the classifiers that can achieve the highest accuracy after 45,000 samples, and an increased/convergent performance in the classifiers that achieve a higher accuracy up to 70%. *ii)* In the figure, we can also perceive a group-like behaviour among some of the classifiers. Neglecting the RBF-SVM model (due to its deficient performance), we can see two groups, one consisting of the classifiers KNN, L-SVM, DT and NB; and the second with LR, ANN and LDA. While the F1-score achieved by the former does not go higher than 70% in our experiments, the latter group show an F1-score performance of almost 90%. By our last sample data (75.6K), all of the classifiers seem to begin convergence. Intrinsically, each of these are mathematically-defined in different forms, however, we require further exploration that could justify a correlation with the manipulation of the input data to each of them and their performance when attempting to predict unestablished lightpaths as in our use case.

The visual representation of the results of the ensemble classifiers are depicted in Fig. 4(a) and 4(b). As for the normal classifiers, we can also see a trade-off between training computational time and the classification performance. While

the results from the AdaBoost and the Bagging models were expected, the low performance of the Random Forest algorithm was not such. However, these results claim to be implementation-dependent, and due to the (*almost*) reasonable performance of the Decision Tree classifier in the previous analysis, we would further explore the implementation of Random Forest.

We have summarized these results in Table 3. In general, the common trade-off between training computational time and accuracy performance suggests that these algorithms may be suitable for enhancing the speed of metro-access transport networks. Further research is required in the categorization of the physical layer features, so that we could build models performing with higher accuracy. In addition, more complex techniques such as deep-/reinforcement-/transfer-learning, seem promising candidates to tackle the weaknesses of the algorithms presented in this study.

Table 2. Parameters used for training the algorithms.

Algorithm	Model Parameters
K-Nearest Neighbours	K:1
Linear SVM	kernel:linear, gamma:100, C:0.0001
RBF SVM	kernel:rbf, gamma:100, C:0.0001
Logistic Regression	solver:lbfgs, multiclass:multinomial, random state:1
Decision Tree	max depth:5
Artificial Neural Network	alpha:1, max iter:10000
Naive Bayes	default
Random Forest	max depth:5, estimators no.:10, max features:1
AdaBoost	estimators no.:10
Bagging	estimators no.:100, max samples:0.8, max features:0.8

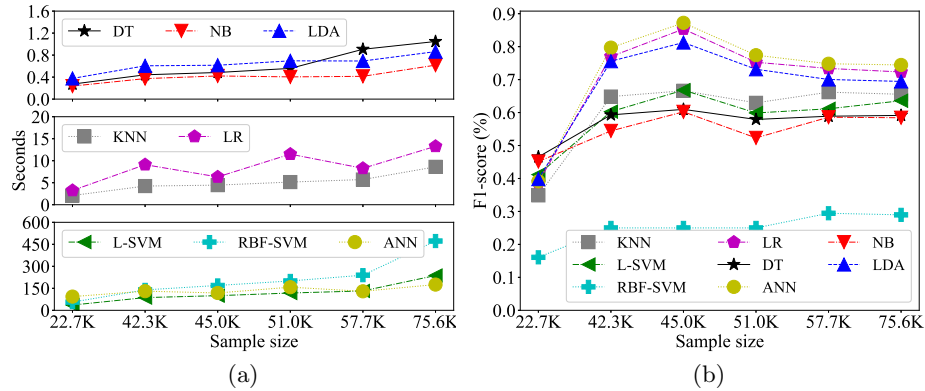


Fig. 3. (a) Training time of each classifier. (b) F1-score of each classifier.

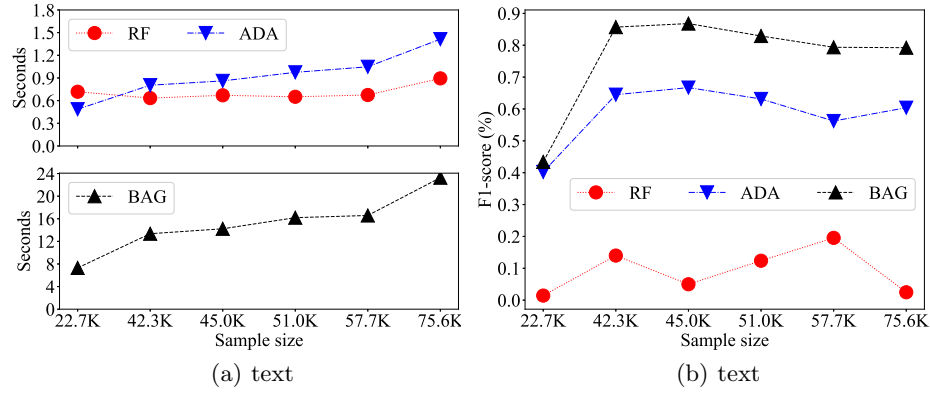


Fig. 4. (a) Training time of each ensemble classifier. (b) F1-score of each ensemble classifier.

Table 3. Results of the F1-score and training computation time for each classifier against the sample sizes.

KNN		L-SVM		RBF-SVM		LR		
<i>*SS</i>	<i>F1-S (%)</i>	<i>TT</i>	<i>F1-S (%)</i>	<i>TT</i>	<i>F1-S (%)</i>	<i>TT</i>	<i>F1-S (%)</i>	<i>TT</i>
22,723	0.34	2.09s	0.41	36.03s	0.16	56.05s	0.38	3.24s
42,320	0.64	4.24s	0.6	1.44m	0.25	2.32m	0.76	9.1s
44,952	0.66	4.45s	0.66	1.66m	0.25	2.82m	0.85	6.31s
50,984	0.62	5.14s	0.59	1.96m	0.25	3.33m	0.75	11.49s
57,740	0.66	5.72s	0.61	2.20m	0.29	3.99m	0.73	8.26s
75,624	0.65	8.6s	0.63	3.95m	0.29	7.87m	0.72	13.27s

DT		ANN		NB		LDA		
<i>*SS</i>	<i>F1-S (%)</i>	<i>TT</i>	<i>F1-S (%)</i>	<i>TT</i>	<i>F1-S (%)</i>	<i>TT</i>	<i>F1-S (%)</i>	<i>TT</i>
22,723	0.46	0.27s	0.39	1.55m	0.45	0.23s	0.39	0.37s
42,320	0.59	0.44s	0.79	2.17m	0.54	0.37s	0.75	0.60s
44,952	0.6	0.48s	0.87	1.97m	0.6	0.42s	0.81	0.61s
50,984	0.57	0.55s	0.77	2.62m	0.52	0.40s	0.73	0.69s
57,740	0.58	0.90s	0.74	2.16m	0.58	0.41s	0.7	0.69s
75,624	0.59	1.04s	0.74	2.92m	0.58	0.61s	0.69	0.86s

* Note: *SS*: sample size; *F1-S*: F1-score; *TT*: training time.

5 Conclusions

In this paper, we have introduced a performance analysis of classification algorithms to address the prediction of OSNR levels of newly established lightpaths in optical networks with different configuration settings and dimensions. Our results suggest that the implementation of such algorithms to complement the decision-making processes of control systems is promising, but requires further analysis of the physical layer features in the optical domain.

Furthermore, we have found an interesting behavioural correlation between some of the classifiers that we will further explore in the future. Also, while in our implementation none of the classifiers achieved a F1-score performance above 90%, we believe that through finer tuning and with the implementation of more complex techniques, such as deep-/reinforcement-/transfer-learning, we could achieve better performance and find a better suitability for the metro-access transport networks scenario.

Acknowledgements

Financial support from Science Foundation Ireland (SFI) 15/US-C2C/I3132 and 13/RC/2077 (CONNECT) are gratefully acknowledged.

References

1. Ruffini, M.: Multidimensional Convergence in Future 5G Networks. *Journal of Lightwave Technology* **35**(3), 535549 (Feb 2017). <https://doi.org/10.1109/JLT.2016.2617896>
2. Ruffini, M., Slyne, F.: Moving the Network to the Cloud: the Cloud Central Office Revolution and Its Implications for the Optical Layer. *Journal of Lightwave Technology* pp. 11 (2019). <https://doi.org/10.1109/JLT.2019.2891990>
3. OpenROADM Project. <http://www.openroadm.org/>
4. Telecom Infra Project. <https://telecominfraproject.com/>
5. Recommendation ITU-T G.7702. <http://handle.itu.int/11.1002/1000/13540>
6. Mata, J., de Miguel, I., Durán, R.J., Merayo, N., Singh, S.K., Jukan, A., Chamania, M.: Artificial intelligence (AI) methods in optical networks: A comprehensive survey. *Optical Switching and Networking* 28, 43–57 (2018). <https://doi.org/https://doi.org/10.1016/j.osn.2017.12.006>, <http://www.sciencedirect.com/science/article/pii/S157342771730231X>
7. Li, Y., Mo, W., Zhu, S., Shen, Y., Yu, J., Samadi, P., Bergman, K., Kilper, D.C.: tSDX: Enabling Impairment-Aware All-Optical Inter-Domain Exchange. *Journal of Lightwave Technology* **36**(1), 142154 (2018)
8. Junio, J., Kilper, D.C., Chan, V.W.S.: Channel power excursions from single-step channel provisioning. *J. Opt. Commun. Netw.* 4(9), A1A7 (Sep 2012). <https://doi.org/10.1364/JOCN.4.0000A1>, <http://jocn.osa.org/abstract.cfm?URI=jocn-4-9-A1>
9. Morais, R.M., Pedro, J.: Machine learning models for estimating quality of transmission in DWDM networks. *IEEE/OSA Journal of Optical Communications and Networking* 10(10), D84D99 (Oct 2018). <https://doi.org/10.1364/JOCN.10.000D84>

10. Mata, J., d. Miguel, I., Durán, R.J., Aguado, J.C., Merayo, N., Ruiz, L., Fernández, P., Lorenzo, R.M., Abril, E.J., Tomkos, I.: Supervised Machine Learning Techniques for Quality of Transmission Assessment in Optical Networks. In: 2018 20th International Conference on Transparent Optical Networks (ICTON). pp. 14 (July 2018). <https://doi.org/10.1109/ICTON.2018.8473819>
11. Díaz-Montiel, A.A., Aladin, S., Tremblay, C., Ruffini, M.: Active Wavelength Load as a Feature for QoT Estimation Based on Support Vector Machine. In: 2019 IEEE International Conference on Communications (ICC) (May 2019 - *Accepted*)
12. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12, 28252830 (2011)
13. Díaz-Montiel, A.A., Yu, J., Mo, W., Li, Y., Kilper, D.C., Ruffini, M.: Performance Analysis of QoT Estimator in SDN-controlled ROADM networks. In: 2018 International Conference on Optical Network Design and Modeling (ONDM). pp. 142147 (May 2018). <https://doi.org/10.23919/ONDM.2018.8396121>
14. Sartzetakis, I., Christodouloupoulos, K., Varvarigos, E.: Formulating QoT Estimation with Machine Learning. In: 2018 European Conference on Optical Communication (ECOC). pp. 13 (Sep 2018). <https://doi.org/10.1109/ECOC.2018.8535429>
15. Tremblay, C., Aladin, S.: Machine Learning Techniques for Estimating the Quality of Transmission of Lightpaths. In: 2018 IEEE Photonics Society Summer Topical Meeting Series (SUM). pp. 237238 (July 2018). <https://doi.org/10.1109/PHOSST.2018.8456791>
16. Barletta, L., Giusti, A., Rottondi, C., Tornatore, M.: QoT estimation for unestablished lighpaths using machine learning. In: 2017 Optical Fiber Communications Conference and Exhibition (OFC). pp. 13 (March 2017)