POLITECNICO DI TORINO
Repository ISTITUZIONALE

Assessment of Cross-train Machine Learning Techniques for QoT-Estimation in agnostic Optical Networks

(Article begins on next page)

25 April 2024

Assessment of cross-train machine learning techniques for QoT-estimation in agnostic optical networks

IHTESHAM KHAN,* MUHAMMAD BILAL, AND VITTORIO CURRI

Politecnico di Torino, Corso Duca degli Abruzzi, 24, 10129, Torino, Italy
*ihtesham.khan@polito.it

Abstract: With the evolution of 5G technology, high definition video, virtual reality, and the internet of things (IoT), the demand for high capacity optical networks has been increasing dramatically. To support the capacity demand, low-margin optical networks engage operator interest. To engross this techno-economic interest, planning tools with higher accuracy and accurate models for the quality of transmission estimation (QoT-E) are needed. However, considering the state-of-the-art optical network's heterogeneity, it is challenging to develop such an accurate planning tool and low-margin QoT-E models using the traditional analytical approach. Fortunately, data-driven machine-learning (ML) cognition provides a promising path. This paper reports the use of cross-trained ML-based learning methods to predict the QoT of an un-established lightpath (LP) in an *agnostic* network based on the retrieved data from already established LPs of an *in-service* network. This advanced prediction of the QoT of un-established LP in an *agnostic* network is a key enabler not only for the optimal planning of this network but it also provides the opportunity to automatically deploy the LPs with a minimum margin in a reliable manner. The QoT metric of the LPs are defined by the generalized signal-to-noise ratio (GSNR), which includes the effect of both amplified spontaneous emission (ASE) noise and non-linear interference (NLI) accumulation. The real field data is mimicked by using a well reliable and tested network simulation tool GNPy. Using the generated synthetic data set, supervised ML techniques such as wide deep neural network, deep neural network, multi-layer perceptron regressor, boasted tree regressor, decision tree regressor, and random forest regressor are applied, demonstrating the GSNR prediction of an un-established LP in an *agnostic* network with a maximum error of 0.40 dB.

1. Introduction

The global data traffic demand will experience a dramatic increase over the next few years [1], driven by the implementation of 5G technology and the expansion of bandwidth-hungry applications; such as high definition video, virtual and augmented reality contents. To sustain this remarkable rise in IP traffic the network operator demands the full exploitation of the residual capacity of already deployed network infrastructure. In order to fully exploit the residual capacity of any network infrastructure, the data transport layer needs to be driven to reach the maximum available capacity. The primary key enabler for optimal data transport exploitation is the dense wavelength division multiplexed (DWDM) transmission technique and network disaggregation. These features kick-off a road map for state-of-the-art technologies such as elastic optical networks (EONs) and software-defined networking (SDN) paradigm in optical networks. The striking features of EON and SDN in optical networks are dynamic and adaptive provisioning of network resources both in control and data plane. In the data plane, the EON paradigm [2,3] has explored a unique optical network architecture able to provide LPs based on the actual traffic demands. This flexibility makes the LP provisioning problem much more challenging than traditional fixed-grid wavelength division multiplexing (WDM) networks. Apart

from this in the control plane, the SDN controller can manage the working points of the various network elements separately, enabling customized management.

Nowadays, optical networks are moving towards partial disaggregation, with a final goal of full disaggregation. The network disaggregation's primary step is to consider the optical line systems (OLSs) that connect the network nodes. In the present scenario, the QoT degradation depends on OLSs controllers' capability to operate at the optimal working point [4,5]. The more precisely this working point is achieved, the lower the margin for traffic deployment and, thus, the larger the deployed traffic rate. Therefore, to reduce the margin, it is essential to rely on a QoT estimator that enables the reliable prediction of the performance of LP before its actual deployment, i.e., the generalized SNR (GSNR), that includes the effect of both ASE noise and NLI accumulation [6] in an *agnostic* network scenarios, i.e., the networks where the network operator does not have an exact knowledge of the working point of network elements (gain and noise figure ripples in amplifiers, insertion losses, etc.).

The main purpose of this work is to explore the relaxation of uncertainty in the GSNR prediction and enable the *agnostic* network controller to reliably deploy the LP with minimum margin. In the present study, we suppose a completely blind scenario by relying only on available data of *in-service* network, i.e., the network where the key operator does have an exact knowledge of the working point of network elements. Typically, analytical models are used for the estimation of QoT in the context of *in-services* network as the analytical approach requires the *exact* description of system parameters that are not possible to achieve in the current context of *agnostic* network. The current frame of work related to *agnostic* network concludes that the analytic approach is not feasible to achieve prior provisioning of QoT of LP in such an agnostic scenario. Furthermore, the uncertainty on the amplifiers' working point, typically induced by a mixed effect of physical phenomena [7] and implementation issues, marking that the analytic strategy failed to accomplish in an open environment.

To contravene this, we optate for an alternative way to use a data-driven ML paradigm, which has already been effectively used in the various context of managing optical networks; see [8–11] for performance monitoring applications. A comprehensive survey of ML applied in optical networks can be found in [12]. Explicitly, coming towards a particular interest of this study, i.e., QoT-E of LP before its deployment, some useful ML methods such as cognitive case-based reasoning (CBR) technique are proposed [13]. Experimental results corresponding to [13] achieved with real field data are discussed in [14]. In [15], the authors used ML techniques for controlling OLS. A technique such as random forests (RF) is proposed to exploit the stored database in [16] to reduce uncertainties on network parameters and design margins. In the context of multicast transmission in an optical network, a neural network (NN) is trained to predict the Q-factor in [17–19]. Several ML techniques for QoT-E of LP before its deployment are also presented in [20,21]. An RF-based binary classifier is presented in [22] to predict the bit-error-ratio (BER) of un-deployed LPs. In [23], an RF classifier, along with the potentiality of two other techniques, i.e., k-nearest neighbor (KNN) and support vector machine (SVM) is suggested. The authors exploit QoT as a label and also made a detailed comparison of the proposed ML techniques. Finally, the analysis reported in [23], showed that the SVM is more refined in performance but worst in computational time.

As compared to the previous literature, we study QoT-E of LP before its actual deployment in an *agnostic* network. We supposed a more realistic approach to planning optical network architecture by targeting the GSNR response to specific traffic configurations of a particular LP of completely *agnostic* network in an open environment. A reliable estimation of QoT against this particular LP of *agnostic* network is acquired by exploiting the data related to GSNRs response of various traffic realizations of deployed LPs of *in-service* network. The exploited data related to GSNRs response of deployed LPs of *in-service* network is perturbated by varying the most delicate parameters of EDFA, noise figure, and amplifier ripple gain with insertion losses.

This prior provisioning of QoT can be used in *agnostic* network planning and the wavelength assignment in the online scenario.

The remainder of the paper is organized as follows. In Section 2, we briefly describe the abstraction of the physical layer, along with the argument that an accurate QoT-E has a key role in minimizing the margin. In Section 3, the simulation model, along with synthetic data generation and analysis, are presented. In Section 4, the orchestration of ML engine is presented. In Section 5, we illustrate the proposed ML methods in terms of QoT-E. Then, in Section 6, we describe the results in detail. Finally, the conclusion and future research direction are drawn in Section 7.

2.   Abstraction of optical transport network

Typically, an optical transport network is a structure of connecting nodes with mesh topology, where traffic request is added/dropped or routed, as shown in Fig. 1(a). The topology links are bidirectional fiber connections from node-to-node, deployed as one or more fiber pairs with single fiber for each particular direction that are generally amplified by specific span length using combine and/or distributed amplification techniques: erbium-doped fiber amplifiers (EDFAs) optionally assisted by some distributed Raman amplification. Furthermore, topology links are commonly expressed as an OLS and a specific controller that has the characteristic to set the working point of each in-line amplifier (ILA) and the spectral load fed at the input of each fiber span.The current state-of-the-art optical network exploits DWDM for spectral usage of fiber propagation and coherent technology for optical transmission. Further to this, the transport layer routing operations are performed using reconfigurable optical add/drop multiplexers (ROADM) technology. The spectral grid considered in DWDM technology can either be fixed or flexible, according to the ITU-T recommendations [24], that define the spectral slots for both of the grid architectures. Using any grid architecture, LPs are deployed, where LPs are the set of possible connections between nodes according to the traffic requirements. Over each deployed LP, a polarization-division-multiplexed (PDM) multi-level modulation format is used for propagation from each source to the destination pair. During the transmission, LP suffers from several propagation impairments such as amplifiers noise added as an ASE, fiber propagation, and ROADM filtering effects. Additionally, the fiber propagation has been widely illustrated that the fiber propagation on an uncompensated optical coherent transmission system impairs the QoT of operated LPs by introducing amplitude and phase noise [4,25–27]. This introduced phase noise is effectively compensated by the DSP module at the receiver, using a carrier phase estimator (CPE) algorithm. This particular kind of noise can only be considered for very high symbol rate transmission designed for short-reach [27]. In contrast to this, the amplitude noise, typically defined as the NLI, always impairs the performance. It is a Gaussian disturbance that accumulates with the receiver's ASE noise. Finally, the ROADMs filtering effects also apply some degradation level to QoT, generally considered an extra loss.

2.1.   GSNR as quality of transmission estimation metric

It is well accepted that the metric of QoT for any candidate LP routed through a particular OLSs is given by the GSNR, including both the effects of the accumulated ASE noise and NLI disturbance, defined as Eq. (1), where OSNR$= P_{Rx}/P_{ASE}$, $SNR_{NL} = P_{Rx}/P_{NLI}$, P_{Rx} is the power of the particular channel at the receiver, P_{ASE} is the power of the ASE noise and P_{NLI} is the power of the NLI. Considering a particular BER vs. OSNR back-to-back characterization of the transceiver, the GSNR precisely gives the BER, as it has been extensively shown in multi-vendor experiments using commercial products [6]. The non-linear effects during fiber propagation generate P_{NLI}, which depends on the power of the particular channel and the spectral load with a cubic law [4]. In the above context, it is quite clear that for each particular OLS, there exists an optimal spectral load that maximizes the GSNR [5]. Given the particular source and destination
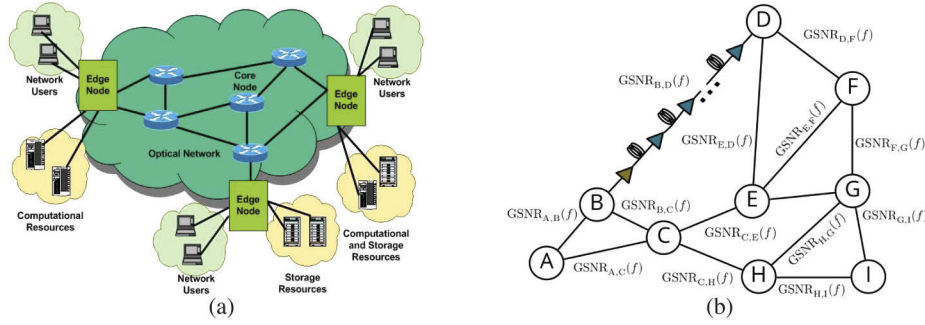
Fig. 1. (a) Architecture of optical network; (b) Abstraction of optical network.

separated by N optical domains, each one is characterized by a GSNR$_i$, where $i = 1, \ldots, N$, the following Eq. (2) clearly demonstrate the overall QoT. Analyzing the propagation effects on a given LP over a particular source and destination, we abstracted the behavior as a cascading effect of each optical domain that introduces QoT impairments. Along with ROADM effects, given LP over a particular source and destination experiences the accumulation of the impairments of all previously traversed OLSs, where each traversed OLS introduces some amount of ASE noise and NLI. Therefore, besides the effects of ROADMs, each LP experiences the cumulative

$$\text{GSNR} = \frac{P_{\text{Rx}}}{P_{\text{ASE}} + P_{\text{NLI}}} = \left(\text{OSNR}^{-1} + \text{SNR}_{\text{NL}}^{-1} \right)^{-1} \tag{1}$$

$$\text{GSNR} = \left(\sum_{i=1}^{N} \frac{1}{\text{GSNR}_i} \right)^{-1} \tag{2}$$

impairments of all previously passed OLSs, where each introduces some amount of ASE noise and NLI. For QoT purposes, the OLS can be abstracted by a unique parameter defined as SNR degradation that, in general, is frequency-dependent (GSNR$_i(f)$), if the OLS controllers can keep the OLS operating at the optimal working point. Hence, considering the above scenario, a network can be abstracted as a weighted graph (G), where G = (V, E) corresponding to the particular topology. The G vertices (V) are ROADM network nodes, while the edges (E) are the OLSs having GSNR$_i(f)$ degradation as weights on the corresponding edges, shown in Fig. 1(b). In particular, for an LP routed from source node A to destination node F that traverses through intermediate nodes C and E, the QoT is:

$$\text{GSNR}_{\text{AF}}^{-1}(f) = \text{GSNR}_{\text{AC}}^{-1}(f) + \text{GSNR}_{\text{CE}}^{-1}(f)) + \text{GSNR}_{\text{EF}}^{-1}(f) \,. \tag{3}$$

After the availability of network abstraction, LPs provisioning can be possible through a particular source and destination with the minimum margin, which depends upon the *GSNR* of a particular source to the destination route.

2.2. Approaches for QoT estimation

In this section, we listed different possible approaches to getting knowledge of OLS characteristics, with each allowing a different reduction in the GSNR uncertainty.

In the first approach, the data available from the system and network components such as static characterization of devices (e.g., amplifier gain and noise figure in the frequency domain, connector loss, etc.) is used to implement an accurate QoT-E in vendor-specific systems. Regarding this particular approach, a considerable number of analytical models are available for calculating the GSNR by using the data and characterizing the OLS components. However, this

static data-based approach may not be accurate as the components experience gradual degradation due to the aging factor, leading to progressively unreliable QoT-E after a certain period.

A second approach is based on the telemetry data considering only the present network status. Deducing an OLS agnostic functioning in an open environment, the OLS controller mainly depends upon telemetry data actualizing from the optical channel monitor (OCM) and the EDFAs. In this particular approach, it is workable to utilize the telemetry of the network's current state to estimate an accurate QoT. In contrast to the previous approach, this circumstantial approach does not rely on the static characterization of devices' parameters and thus removes the uncertainty in the QoT-E accuracy due to the device aging factor discussed in the former approach. Nevertheless, this particular method's problem is that the GSNR response, mainly its OSNR component, is highly dependent upon the spectral load configuration, which deduces a considerable uncertainty in the QoT margin [15].

The third approach considers a data set that collects the QoT responses against random spectral loads of *in-service* network. This data is generated during the operative phase of the *in-service* network by measuring the OLS response in terms of GSNR for various spectral load configurations. This particular case provides an ideal playground to apply ML. An ML method using a training data set composed of spectral load realizations of an *in-service* network yields an accurate QoT-E for each newly generated spectral load realization of *agnostic* network scenario. In contrast to the second approach, where only telemetry data is considered, this approach utilizes the QoT-E based on the GSNR response to specific spectral load configurations of *in-service* network, decreasing the uncertainty in GSNR predictions of *agnostic* network. Additionally, this method does not need any knowledge of the OLS's physical parameters in opposition to the first approach. Thus, this method provides an ideal platform to apply ML methods in an *agnostic* network scenario.

In this work, we focus on the third approach, the ML model's training on the GSNR response to specific spectral load configurations of the already deployed *in-service* network, and consider its realization to predict the QoT of an *agnostic* network shown in Fig. 2(a).
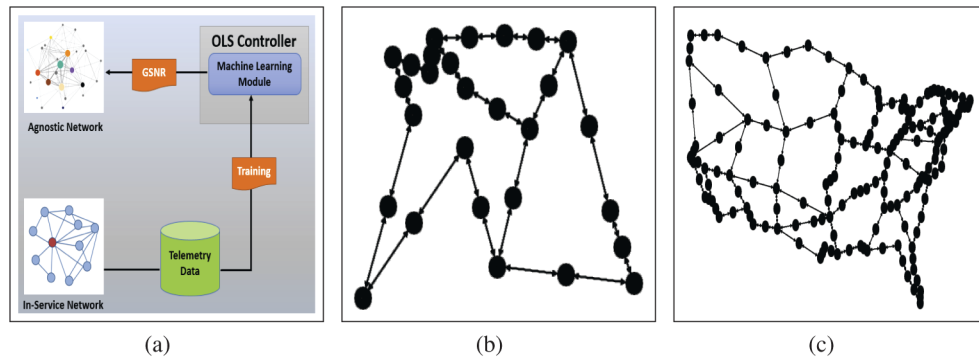
Fig. 2. (a) Model orchestration; (b) EU network topology; (c) USA network topology.

3. Simulation model and synthetic data generation & analysis

A typical SDN empowered backbone optical network is considered, in which edges are modeled by OLSs, while nodes are defined as ROADM sites. The given OLSs comprise of fibers and amplifiers and are managed by a controller supposed to operate at the optimal nonlinear-propagation working point. The random behavior of the physical layer is considered through the amplifier gain ripple. To these fiber impairments such as fiber attenuation (α), dispersion (D), and insertion losses are also considered. In order to make the simulation more realistic, the statistics

of insertion losses are determined by an exponential distribution with $\lambda = 4$, as described in study [28]. Because of the limitation of computational resources, the considered OLSs carry only 76 channels over the standard 50 GHz grid on the C-band, having total bandwidth close to 4THz. We do not expect a substantial difference in results when considering standard 96 channels on the entire C-band. We are supposed to rely on transceivers at 32 GBaud, shaped with a root-raised-cosine (RRC) filter. The ILA, particularly EDFAs in the optical line, is configured to work at a constant output power of 0 dBm per channel. All network links are supposed to operate on standard single-mode fiber (SMF) with a span length of 80 km. The simulation parameters are given in Table 1.

Table 1. Simulation Parameters

Launch Power/ Channel	[0-0.006] *dBm*
Dispersion (*D*)	16.0 *ps/nm/km*
Attenuation coefficient (α)	0.3 *dB/km*
Channel Spacing	50 *GHz*
Average Span Length	$\approx$ 80 *km*
WDM Comb (C-Band)	76
Baud Rate	32 *Gbaud*
Amplifier Noise Figure	[6 - 11] *dB*
Amplifier Gain Ripple	Variation of 1 *dB*
Fiber Type	Standard SMF
Insertion losses	Exponential Distribution ($\lambda = 4$) [28]

To retrieve a data set, in the absence of real field data, a reliable and well-tested open-source network simulation tool GNPy [29,30] is used to generate the synthetic data against the proposed scenario. Generally, this library outlines an end-to-end simulation environment that develops the network models for the physical layer [31]. The QoT-E engine of the GNPy library is spectrally resolved and is based on generalized Gaussian noise (GGN) model [30,32]. Characterizing this particular capability, the GNPy is configured to mimic the spectral load configuration of a network. The mimicked parameters are signal power at receiver, ASE noise accumulation, and NLI generation during the propagation of LP, GSNR against each propagating LP from a particular source to a destination. Finally, the number of spans traversed by candidate LP from source to destination. Considering the ASE noise and NLI augmentation, the ASE is more prominent, because it is twice the NLI when the system operates at optimal power [4,33]. Remarkably, it is also the most strenuous to measure. The ASE noise power depends on the working point of EDFAs [34], which eventually depends on the spectral load [7].In the above frame of reference, the generated data set is perturbated by varying the most delicate parameters of EDFA; noise figure and amplifier ripple gain. The selection of noise figure is made by uniform distribution varying between 6 dB to 11 dB while the amplifier ripple gain is varied uniformly with 1 dB of variation. Additionally, the generated data set is further perturbed by insertion losses figured by an exponential distribution [28]. The mimicked data set consists of two different sub-set of data; one set refers to an *in-service* network while the other refers to an *agnostic* network. The configuration used for an *agnostic* network scenario is the same as for the former *in-service* network case except for perturbed EDFA noise figure, ripple gain, and insertion losses in the OLS already described.

After concluding the basic network configuration, the most elegant part is the spectral load configuration of simulated links. The proposed model is the subset of 2^{76}, where 76 is the total number of channels. In the considered subset of spectral load realization, each source-to-destination (s $\rightarrow$ d) pair has 1024 realizations of random traffic ranging from 34% to 100%

of total bandwidth utilization. The first data set generated against the European Union (EU) Network Topology, used as an *in-service* network and for an *agnostic* network, the required data set is generated against the USA Network Topology shown in Fig. 2(b) and Fig. 2(c). The raw statistics of both network topologies are shown in Table 2. In order to mimic the proposed behaviour, GNPy is configured to generate 4096 realizations of spectral load configuration for 4 (s –> d) pairs of *in-service* EU Network and 35,840 realizations of spectral load against 35 (s –> d) pairs of an *agnostic* USA Network shown in Table 3 and Table 4.

Table 2. Topologies Details

	EU: Training	USA : Testing
Average distance (Km)	2014.06	2541.75
Maximum distance (Km)	3051.10	5481.07
Minimum distance (Km)	669.30	568.33
Average number of spans	19.75	27.49

Table 3. EU: Training

Source	Destination	Spans	Source	Destination	Spans
Vienna	Warsaw	7	Brussels	Bucharest	30
Amsterdam	Berlin	8	Frankfurt	Istanbul	34

Table 4. USA : Testing

Source	Destination	Spans	Source	Destination	Spans
Milwaukee	Minneapolis	6	Denver	Detroit	29
Louisville	Memphis	7	Kansas City	Las-Vegas	30
Boston	Buffalo	8	Las-Vegas	Little Rock	30
Charleston	Charlotte	8	Albuquerque	Atlanta	33
Chicago	Cincinnati	12	Birmingham	Bismarck	33
Greensboro	Hartford	14	Austin	Baltimore	34
Atlanta	Austin	20	Miami	Milwaukee	35
Charlotte	Chicago	20	Abilene	Albany	36
Columbus	Dallas	20	Bismarck	Boston	36
Dallas	Denver	20	Detroit	El Paso	38
El Paso	Fresno	21	Hartford	Houston1	38
Minneapolis	Nashville	21	Baton Rouge	Billings	41
Buffalo	Charleston	22	Billings	Birmingham	41
Houston	Jacksonville	23	Albany	Albuquerque	45
Memphis	Miami	24	Los Angeles	Louisville	46
Baltimore	Baton Rouge	25	Fresno	Greensboro	54
Little Rock	Long Island	25	Long Island	Los Angeles	61
Jacksonville	Kansas City	26			

After retrieving the data set, the statistical analysis of the GSNR against different spectral load configurations for particular (s –> d) is made to estimate the uncertainty in GSNR calculations. We considered a single path {Milwaukee-Minneapolis} in the test data set for the statistical analysis of the GSNR. We selected only one path for analysis as we do not expect a substantial

difference in the statistical characteristics of GSNR for the other paths.Firstly, the distribution of *GSNR* for this particular path is depicted in Fig. 3. In this regard, we selected channel under test (channel number 10) for all the test realization in which this candidate channel is always *ON*. Along with this analysis, few primary considerations arise by computing the average of the GSNRs for every channel over all the test realizations of this peculiar path, presented in Fig. 4. The average values of GSNRs present a characteristic figure of the OLSs component, which comes to pass between 13.27 dB, and 13.90 dB, along with standard deviations from 0.233 dB to 0.33 dB respectively.
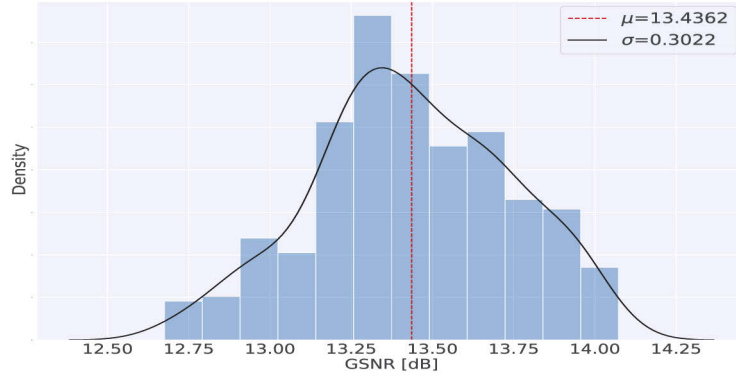
Fig. 3. GSNR distribution of acquired data set.
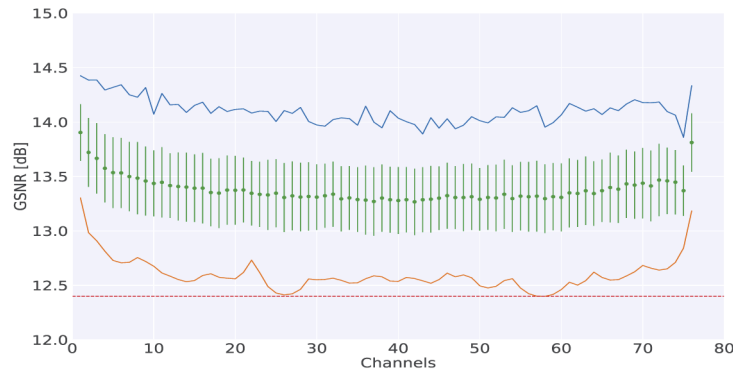
Fig. 4. Overall, GSNR measurements for a single path in the frequency domain. The green dots are the mean values over each channel's entire sample; the error bars are equal to the standard deviations. In blue and orange, the maximum and the minimum for each channel are outlined, respectively. The dashed red line indicates the overall GSNR minimum of 12.4 dB.

Considering the current scenario, if nothing is known about the GSNR dependency upon frequency, the same GSNR threshold must be enforced for all channels with magnitude lower than a global expected minimum. In this case, the GSNR_i^{p} are fixed to the constant GSNR threshold of 12.4 dB creating an average margin of up to 1.70 dB over a considered set of realizations for this particular path. Moving towards the next scheme, consider the availability of stored data that particularizes the frequency-resolved GSNR response, one can truncate the margin by fixing a minimum value for each channel that must lie under the respective minimum measurement (continuous orange line in Fig. 4). However, this solution is sub-optimal; it is the most acceptable reachable result, conservative and agnostic with respect to the definite spectral

load configuration against this peculiar path. This definitive solution yields a finite improvement, compared to the first case having a value of 1.70 dB, as the average margin would be reduced to 1.51 dB. However, this second approach is strikingly dependent upon the sample space of GSNR realization; acquiring a minimum reliable value for each channel requires a sufficiently large number of instances. In the above context, both the considered scenarios are not feasible given *agnostic* network, where the statistics of the GSNR against different spectral load configuration is not available.

In contrast to this, an ML approach appears to be a promising candidate not only to predict the GSNR against any particular path of a network but also aided a decrease in the uncertainty in GSNR predictions, even if the dimension of the sample of *in-service* network is limited.

4.   Machine learning model orchestration

The proposed models cognate the features and labels of the candidate LP of the *in-service* network using ML. The manipulated parameters used to define the features for ML models include received signal power, NLI, ASE, channel frequency, and distance between source to the destination node (integral number of fiber spans traversed). In contrast, the exploit label is manipulated by the *GSNR* parameter of the candidate LP depicted in Fig. 5. The total number of input features for proposed ML models consist of *380* entries, as we have 76 entries against each manipulated parameters *(76 X 5 = 380)*. The considered models utilize ML's functionality, which is a powerful tool to find the relationship between the provided features and the desired label [35].Typically almost all ML-based models perform better when they are trained on standardized data sets [36]. Generally, the standardized data set has zero mean and unit variance.

$$Z = \frac{X - \mu}{\sigma} \tag{4}$$

$$\text{MSE} = \frac{\sum_{i=0}^{n} \left(\text{GSNR}_i^p - \text{GSNR}_i^a \right)^2}{n} \, , \tag{5}$$

In the present work, standardization of data set is done using z-score normalization, expressed in Eq. (4), where μ and σ is the mean and standard deviation of each particular feature vector of data set. Along with this, model evaluation is done by *mean square error (MSE)* as a loss function expressed in Eq. (5), where GSNR_i^a and GSNR_i^p are the actual and predicted values of any candidate channel for the *i*th spectral load, respectively, and n is the total number of realizations in the test data set.
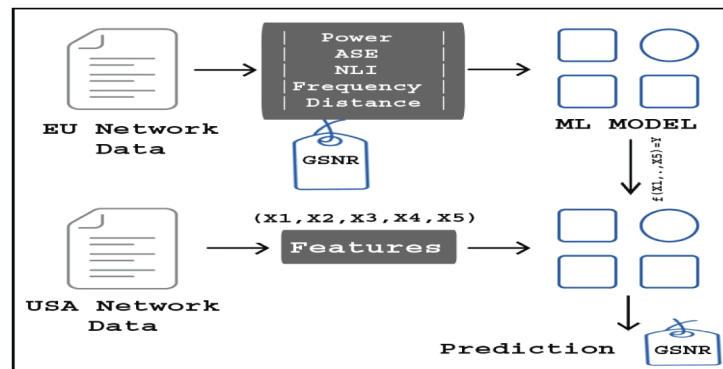

Fig. 5. Machine learning module.

5.   Machine learning analysis

The paradigm of ML and especially deep learning concept allows the inference of useful network characteristics that cannot be easily or directly measured. Generally, ML models' cognition ability is provided by a series of intelligent algorithms that learn the inherent information of the training data. The inherent information is then abstracted into the decision models that guide the testing phase. These trained decision models offer operational advantages by allowing the network to *draw conclusions* and react autonomously.

In the present work, six ML models are proposed for QoT-E. Each proposed model consists of three basic modules; pre-processing, training, and testing. The pre-processing module is used to standardize the data set before applying it to the training module. The training module uses the standardized data set of EU Network, which is considered as *in-service* network shown in Table 3 for the training of the proposed models. After training, the testing module explicitly starts testing on the USA Network data set subset, considered as *agnostic* network shown in Table 4.

The proposed ML-based models are developed by using high-level python application program interface (API) of two open-source ML libraries, *scikit-learn*[©] (SKL) [36] and *TensorFlow*[©] (TF) [37], which provide a variety of learning algorithms as well as appropriate functions to refine the data set before using as an input to ML model. Generally, SKL is a general-purpose or a traditional ML library, while TF is considered a deep learning library. Unlike TF, SKL provides a powerful feature engineering tool kit for data refining processes such as dimensional reduction, standardization, transformation, etc. while TF automatically extracts useful features from the data and does not need to this manually.

Using python API of *SKL's*, three possible ML models; Decision Tree Regressor, Random Forest Regressor, and Multi-Layer Perceptron Regressor, while using *TF's*; Boosted Tree Regressor, Deep Neural Network and Wide Deep Neural Network are developed.

5.1.   Decision tree regressor

To estimate QoT, the Decision tree regressor (DTR) model is proposed, which provides direct relationships between the input and response variables [38]. DTR constructs a tree based on several decisions made by analyzing different aspects of data set features and eventually leads to a response variable. The DTR has two main basic tuning parameters; *min_samples_leaf* and *max_depth*. To get the optimum values of these parameters, we tuned the number of *min_samples_leaf* = 3 and *max_depth* = 100 in order to achieve the best trade-off between precision and computational time in the particular simulation scenario.

5.2.   Random forest regressor

The Random forest regressor (RFR) is a type of ML algorithm that creates an *ensemble* of regression trees using a *bagging* technique [39]. *Bagging* creates various subsets of data from the training data set chosen randomly with replacement. The RFR is a step extension over *Bagging*. It not only takes a random subset of data but also a random selection of features rather than using all the features to train several decision trees. The final prediction of the RFR is made by merely averaging the predictions of each individual decision tree. Similar to DTR parameter tuning, we also do the same for RTR and decide *min_samples_leaf* = 3 and *max_depth* = 100 in the particular simulation scenario.

5.3.   Multi-layer perceptron regressor

The Multi-layer perceptron regressor (MLPR) is one of the most commonly used AI networks. Generally, MLPR is not a single perceptron with multiple layers but multiple artificial neurons (perceptrons) with multiple layers. Typically, MLPR has three or more layers of perceptrons. The layers of the MLPR form a directed, acyclic graph where each layer of MLPR is fully connected

to the subsequent layer [40]. The proposed MLPR is configured by several parametric values such as *training steps* = 1000, loaded with back-propagation (BP) algorithm along with default *Stochastic Gradient Descent Algorithm (SGD)* optimizer having default *learning rate* = 0.01 and L_1 regularization = 0.001 [41]. The basic function of L_1 regularization is to prevent MLPR from over-fitting. In addition to this, several non-linear activation functions such as *Relu, tanh, sigmoid* are tested during model building. After testing all of these non-linear activation functions, *Relu* is selected to empowered MLPR, as it outperforms others in terms of prediction and computational load [42]. Finally, the most important is the *hidden-layer*, MLPR is tuned on several numbers of *hidden-layer* and neurons to achieve the best trade-off between precision and computational time. These two parameters are linked to the complexity of the MLPR, which is tied to the complexity of the problem. Although the increase in the number of layers and neurons improves the accuracy of the MLPR up to a certain extent, a further increase in these values has an adverse effect that causes over-fitting and increases in the computational time. In this trade-off, MLPR for QoT-E uses *3 hidden-layers*, containing *20 neurons* each.

5.4. Boosted tree regressor

The Boosted tree regressor (BTR) is also a type of ML algorithm that creates an *ensemble* of regression trees using the *Gradient-Boosting* technique. BRT works by combining various regression trees' models, particularly decision trees, using *Gradient-Boosting* [43]. More formally, this class of models can be written as in Eq. (6), where the final regressor f is the sum of

$$f(x) = r_0 + r_1 + r_2 + \cdots \cdots .. + r_i \tag{6}$$

simple base regressor r_i. Similar to other tree regressors parameter tuning, we also do the same for BTR and decide *min_samples_leaf* = 3 and *max_depth* = 100. Further more, BTR is also configured by several other parameters such as default *learning rate* = 0.01 along with L_1 regularization = 0.001 for absolute weights of the tree leafs for the current simulation scenario.

5.5. Deep neural network

The Deep neural network (DNN) is a type of Artificial Neural Network (ANN) with multiple layers between the input and output layers [44]. Each particular layer of DNN consists of multiple neurons, the computational and learning unit of neural networks shown in Fig. 6. For the QoT-E, the proposed DNN is configured by several parametric values such as *training steps* = 1000, loaded with default *Adaptive Gradient Algorithm (ADAGRAD)* keras optimizer with default *learning rate* = 0.01 and L_1 regularization = 0.001 [45]. The basic function of L1 regularization is to prevent DNN from over-fitting. In addition to this, during model building non-linear activation function, *Relu* is selected that allows translation of the given input features into the prediction label of our point of interest with less complexity [42]. Finally the most important is the number of *hidden-layers*, DNN is tuned on several number of *hidden-layers* and neurons in order to achieve the best trade-off between precision and computational time. In this trade-off, DNN for QoT-E uses *3 hidden-layers*, containing *20 nodes* each.

5.6. Wide deep neural network

The Wide deep neural network (W-DNN) is a type of DNN architecture that combines the strength of *memorization* with *generalization* [46]. Generally, W-DNN is synergically trained on *wide* linear model such as *Linear Regression* (LR) for memorization and for generalization on *DNN*. The output of wide LR and generalized DNN are combined at the output layer to get the final prediction shown in Fig. 6(b). For the QoT-E, the proposed W-DNN is configured with 1000 *training steps*, loaded with default *Follow the Regularized Leader* (FTRL) keras optimizer with default *learning rate* = 0.01 and L_1 regularization = 0.001 against wide *LR* model [47], while generalize *DNN* is loaded with default *ADAGRAD* keras optimizer with default *learning*
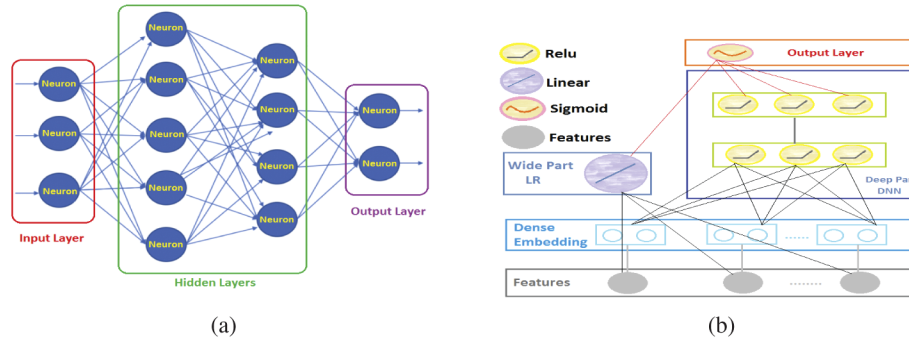
Fig. 6. (a) Deep neural network architecture; (b) Wide deep neural network architecture.

rate = 0.01 and L_1 regularization = 0.001 [45]. The *Relu*, nonlinear activation functions is selected to empowered deep part, as it outperform others non-linear activation function in terms of prediction [42]. The output layer is fed with *sigmoid* function, as *sigmoid* produces activation values in a particular range so that the output layer will always be activated. During the training of W-DNN, the wide and deep parts are jointly trained at the same time. The loss function over training steps for W-DNN is observed in Fig. 12. The synergic training of both wide and deep architectures optimizes all parameters and the related weights of their sum, taken into account during the training time.

6. Results and discussion

The numerical assessment of various ML models developed by using python APIs of *SKL* and *TF* libraries have been performed by considering 4 different paths of EU Network for training and 35 different paths of USA Network for testing. The prediction power of the each proposed model is estimated by calculating the $\Delta GSNR$, where $\Delta GSNR = GSNR_{Predicted}$ - $GSNR_{Actual}$. The proposed models are simulated on a workstation having specifications, 32 GB of 2133 MHz RAM and an Intel Core i7 6700 3.4 GHz CPU.

6.1. Comparison of SKL based learning methods for QoT-E

In this section, we exploit the proposed ML models' prediction performance based on *SKL's* API: DTR, RFR, and MLPR. In Fig. 7 distribution of $\Delta GSNR$, a prediction error metric is plotted for DTR, RFR, and MLPR for the test samples of *on channels realization* only. For the given simulation scenario, the DTR is unable to find the underlying relationship and irregularities. On the other hand, the RFR took advantage of averaging various decision trees instead of a single decision tree, trained on randomly selected subsets of the training samples. Therefore, the overall performance of the RFR is much better than the DTR. Furthermore, the MLPR performed very well due to its cognitional potentiality provided by internally configured neurons compared to the DTR and RFR. The above descriptive results are verified by observing the mean (μ) and standard deviation (σ) of $\Delta GSNR$ distribution against each proposed model in Fig. 7.

Furthermore, observing Fig. 9(a), the box plot of $\Delta GSNR$ also depicts that the prediction performance of MLPR outperforms DTR and RFR. In Fig. 9(a), the central rectangle span specifies the first quartile (Q_1) to the third quartile (Q_3). A segment inside the rectangular box shows the *median* of $\Delta GSNR$ and *"whiskers"* around the rectangular box show the minimum and maximum values of $\Delta GSNR$. Focusing on MLPR, after observing Fig. 7 and Fig. 9(a), we further analyze and elaborate the results related to the MLPR by showing the bean plot of $\Delta GSNR$ distribution against all the test paths of *agnostic* USA Network in Fig. 11. In Fig. 11, the *"whiskers"* above and below the bean show the out bound of $\Delta GSNR$, the black segments show
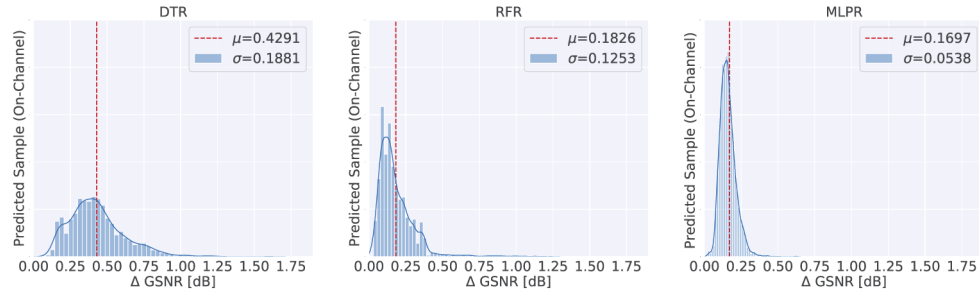
Fig. 7. *SKL* based learning methods.

individual observation along with the *μ* value (*red segment in each bean*) of Δ*GSNR* for each test path. After demonstrating prediction performance, we further analyze the training time for each *SKL* based learning method shown in Fig. 10(a). The Fig. 10(a), depicts that the proposed MLPR takes longer time duration during training as compared to RFR and DTR due to its internal fully connected *hidden perceptrons*. The RFR takes a slighter longer duration than the DTR because of its dependency over the *bagging technique*.

6.2. Comparison of TF based learning methods for QoT-E

In this section, we are doing the same analysis for *TF* based learning methods as we have done for *SKL* based learning methods. Initially, we exploit the proposed ML models' prediction performance based on *TF's* API: BTR, DNN, W-DNN shown in Fig. 8. Figure 8 demonstrates the distribution of Δ*GSNR* for W-DNN, BTR and DNN for the test samples of *on channels*

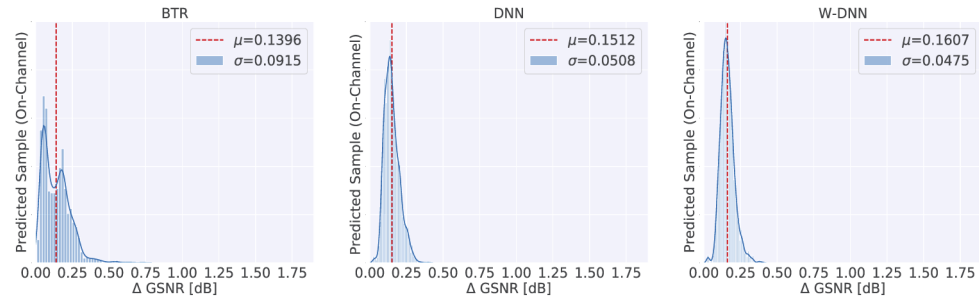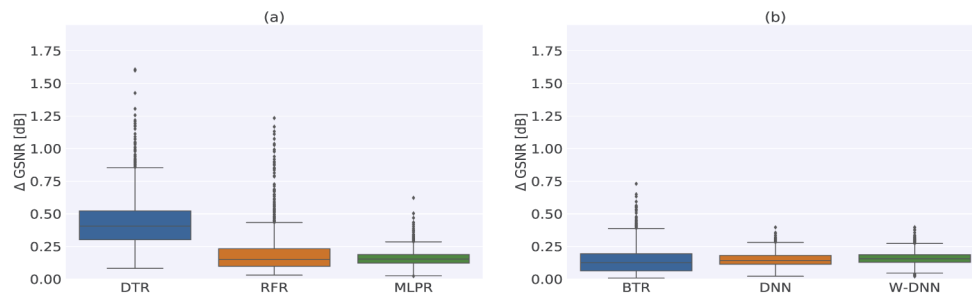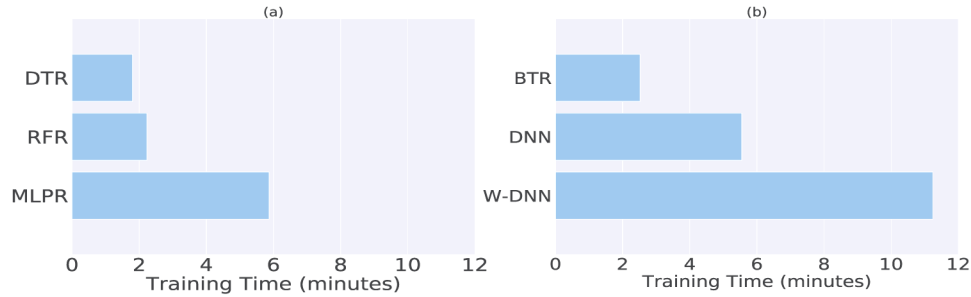**Fig. 8.** *TF* based learning methods.

Fig. 9. (a) Comparison of *SKL* based learning methods; (b) Comparison of *TF* based learning methods.

realization only. For the given simulation scenario, the BTR took advantage of the *boasting technique*, combining various models of the regression trees and select the new tree that best reduces the *loss function* instead of choosing randomly. On the other hand, the DNN outperforms BTR due to its cognitional potentiality provided by internally configured neurons as compared to BTR. Finally, the W-DNN gives more refined tuning to DNN as it combines the LR and DNN. The wider LR gives more refined tuning to the DNN by characterizing features at the output layer. The above-described results are verified by observing the μ and σ of $\Delta GSNR$ distribution for each proposed model in Fig. 8.

In order to better visualize the comparison among W-DNN, BTR and DNN, Fig. 9(b) shows the box plot of $\Delta GSNR$ distribution for each *TF* based learning method. Focusing on W-DNN after observing Fig. 8 and Fig. 9(b), it is quite obvious that W-DNN is the best *TF* based learning method in the present simulation scenario. We further analyze and elaborate the results related to W-DNN by showing the bean plot of $\Delta GSNR$ distribution of all the test paths of *agnostic* USA Network in Fig. 13. Furthermore, we also analyze the training time for each *TF* based learning method shown in Fig. 10(b). Figure 10(b) depicts that the proposed W-DNN takes a long time duration during training compared to the DNN and BTR due to the synergic training of LR

Fig. 10. (a) Training time of *SKL* based learning methods; (b) training time of *TF* based learning methods.
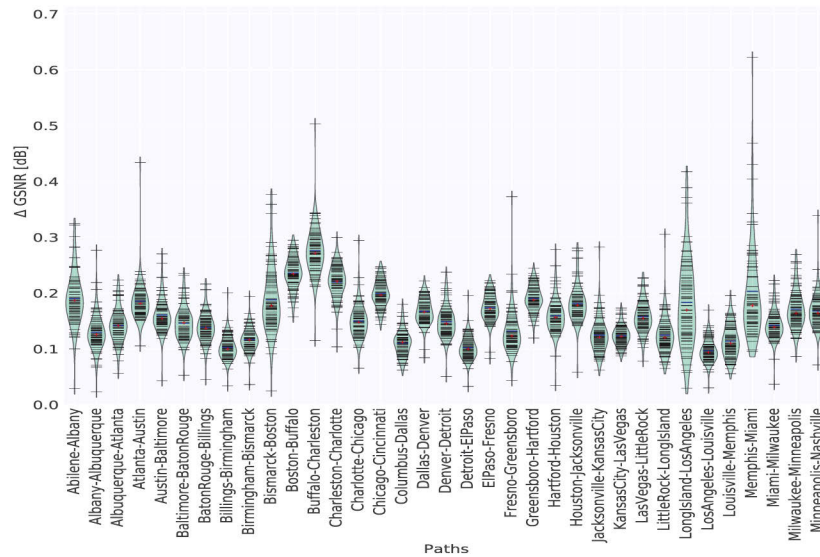
Fig. 11. Distribution of $\Delta GSNR$ for the simulated links of USA network using *MLPR*.

and DNN. The DNN takes a longer duration than the BTR because of its internal *hidden-layers* containing several neuron units.

Finally, we analyze the prediction ability of the finest models of the two libraries; MLPR and W-DNN against all the test lightpaths. Observing Fig. 13 and Fig. 11 the W-DNN shows maximum Δ*GSNR* = 0.40 dB against against *Buffalo-Charleston* lightpath while the MLPR shows maximum Δ*GSNR* = 0.62 dB against *Memphis-Miami* lightpath. The remarkable performance of W-DNN is because of the synergic training of LR and DNN, which enables it to outperform the traditional MLPR and DNN. Furthermore, the classical MLPR shows a large deviation of Δ*GSNR* because it is based on the gradient-based local search technique, which at some point during training gets stuck in an unwanted local minimum.
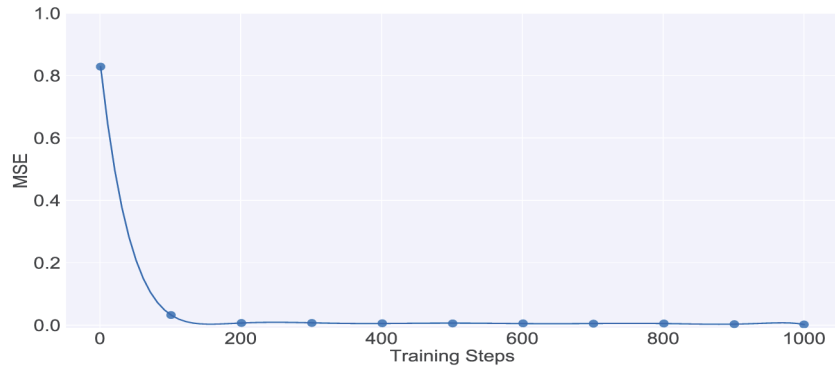

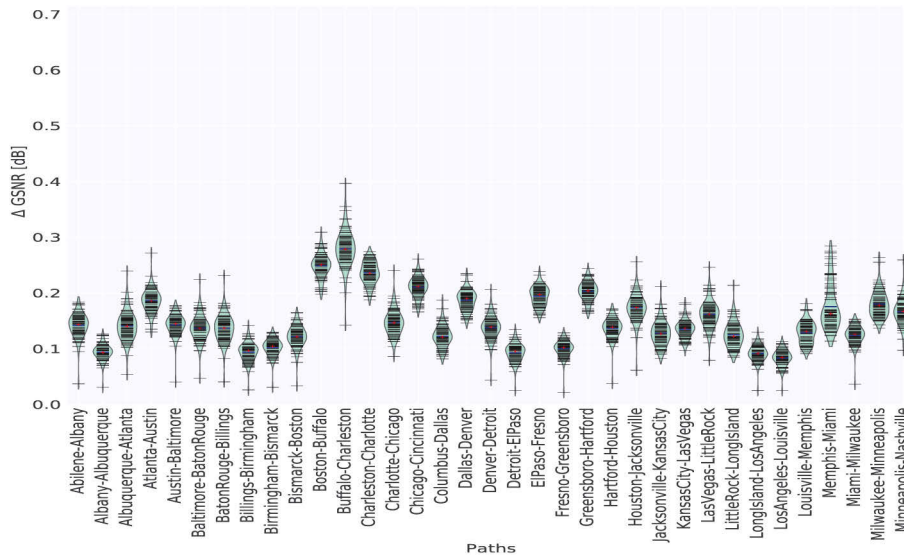**Fig. 12.** W-DNN loss function over the training steps.

Fig. 13. Distribution of Δ*GSNR* for the simulated links of USA network using *W-DNN*.

7. Conclusion

In summary, we proposed and exploit the ability of several different ML models for QoT-E, considering the scenario of cross-training of these ML techniques on *in-service* EU Network and

tested on completely *agnostic* USA network. The proposed ML models are developed by using higher-level APIs of *SKL* and *TF* libraries. The developed models are cross-trained and tested on the synthetic data generated by the GNPy library.

Exploiting the ability of ML, W-DNN performs barely better than DNN due to the enhancement provided by synergic training of generalize (DNN) and wide (LR) architectures. The prediction performance of the applied DNN is marginally more refined than the MLPR since the back-propagation algorithm used by the MLPR is based on the gradient-based local search technique, which at some point during training gets stuck in an unwanted local minimum [48].

In addition to this, the BTR and RFR prediction is better than DTR due to the use of ensembling techniques; *boosting and bagging*. Finally, results demonstrate that ML techniques are an undeniable alternative for faster and precise provisioning QoT of an LP in an *agnostic* optical network scenario. Remarkably, the W-DNN proved to be the model achieving the best generalization with maximum prediction error smaller than 0.40 dB.

Disclosures

The authors declare no conflicts of interest.

References

1. Cisco, "Cisco Visual Networking Index: Forecast and Trends, 2017–2022," Tech. rep., Cisco (2017).
2. O. Gerstel, M. Jinno, A. Lord, and S. B. Yoo, "Elastic optical networking: A new dawn for the optical layer?" IEEE Commun. Mag. **50**(2), s12–s20 (2012).
3. G. Zhang, M. De Leenheer, A. Morea, and B. Mukherjee, "A survey on ofdm-based elastic core optical networking," IEEE Commun. Surv. Tutorials **15**(1), 65–87 (2013).
4. V. Curri, A. Carena, A. Arduino, G. Bosco, P. Poggiolini, A. Nespola, and F. Forghieri, "Design strategies and merit of system parameters for uniform uncompensated links supporting nyquist-WDM transmission," J. Lightwave Technol. **33**(18), 3921–3932 (2015).
5. R. Pastorelli, "Network optimization strategies and control plane impacts," in *OFC*, (OSA, 2015).
6. M. Filer, M. Cantono, A. Ferrari, G. Grammel, G. Galimberti, and V. Curri, "Multi-Vendor Experimental Validation of an Open Source QoT Estimator for Optical Networks," J. Lightwave Technol. **36**(15), 3073–3082 (2018).
7. M. Bolshtyansky, "Spectral hole burning in erbium-doped fiber amplifiers," J. Lightwave Technol. **21**(4), 1032–1038 (2003).
8. M. Freire, S. Mansfeld, D. Amar, F. Gillet, A. Lavignotte, and C. Lepers, "Predicting optical power excursions in erbium doped fiber amplifiers using neural networks," in *2018 (ACP)*, (IEEE, 2018), pp. 1–3.
9. J. Thrane, J. Wass, M. Piels, J. C. Diniz, R. Jones, and D. Zibar, "Machine learning techniques for optical performance monitoring from directly detected pdm-qam signals," J. Lightwave Technol. **35**(4), 868–875 (2017).
10. F. N. Khan, C. Lu, and A. P. T. Lau, "Optical performance monitoring in fiber-optic networks enabled by machine learning techniques," in *2018 (OFC)*, (IEEE, 2018), pp. 1–3.
11. L. Barletta, A. Giusti, C. Rottondi, and M. Tornatore, "Qot estimation for unestablished lighpaths using machine learning," in *OFC*, (OSA, 2017).
12. J. Mata, I. de Miguel, R. J. Duran, N. Merayo, S. K. Singh, A. Jukan, and M. Chamania, "Artificial intelligence (ai) methods in optical networks: A comprehensive survey," Opt. Switching Netw. **28**, 43–57 (2018).
13. T. Jiménez, J. C. Aguado, I. de Miguel, R. J. Durán, M. Angelou, N. Merayo, P. Fernández, R. M. Lorenzo, I. Tomkos, and E. J. Abril, "A cognitive quality of transmission estimator for core optical networks," J. Lightwave Technol. **31**(6), 942–951 (2013).
14. A. Caballero, J. C. Aguado, R. Borkowski, S. Salda na, T. Jiménez, I. de Miguel, V. Arlunno, R. J. Durán, D. Zibar, J. B. Jensen, R. M. Lorenzo, E. J. Abril, and I. T. Monroy, "Experimental demonstration of a cognitive quality of transmission estimator for optical communication systems," Opt. Express **20**(26), B64–B70 (2012).
15. A. D'Amico, S. Straullu, A. Nespola, I. Khan, E. London, E. Virgillito, S. Piciaccia, A. Tanzi, G. Galimberti, and V. Curri, "Using machine learning in an open optical line system controller," J. Opt. Commun. Netw. **12**(6), C1–C11 (2020).
16. E. Seve, J. Pesic, C. Delezoide, S. Bigo, and Y. Pointurier, "Learning process for reducing uncertainties on network parameters and design margins," J. Opt. Commun. Netw. **10**(2), A298–A306 (2018).
17. T. Panayiotou, S. P. Chatzis, and G. Ellinas, "Performance analysis of a data-driven quality-of-transmission decision approach on a dynamic multicast-capable metro optical network," J. Opt. Commun. Netw. **9**(1), 98–108 (2017).
18. W. Mo, Y.-K. Huang, S. Zhang, E. Ip, D. C. Kilper, Y. Aono, and T. Tajima, "Ann-based transfer learning for qot prediction in real-time mixed line-rate systems," in *2018 (OFC)*, (IEEE, 2018), pp. 1–3.

19. R. Proietti, X. Chen, A. Castro, G. Liu, H. Lu, K. Zhang, J. Guo, Z. Zhu, L. Velasco, and S. B. Yoo, "Experimental demonstration of cognitive provisioning and alien wavelength monitoring in multi-domain eon," in *Optical Fiber Communication Conference*, (Optical Society of America, 2018), pp. W4F–7.

20. I. Khan, M. Bilal, M. Siddiqui, M. Khan, A. Ahmad, M. Shahzad, and V. Curri, "Qot estimation for light-path provisioning in un-seen optical networks using machine learning," in *ICTON*, (IEEE, 2020).

21. I. Khan, M. Bilal, and V. Curri, "Advanced formulation of qot-estimation for un-established lightpaths using cross-train machine learning methods," in *ICTON*, (IEEE, 2020).

22. C. Rottondi, L. Barletta, A. Giusti, and M. Tornatore, "Machine-learning method for quality of transmission prediction of unestablished lightpaths," J. Opt. Commun. Netw. **10**(2), A286–A297 (2018).

23. S. Aladin and C. Tremblay, "Cognitive tool for estimating the qot of new lightpaths," in *OFC*, (OSA, 2018).

24. https://www.itu.int/rec/T-REC-G.694.1/en.

25. D. J. Elson, G. Saavedra, K. Shi, D. Semrau, L. Galdino, R. Killey, B. C. Thomsen, and P. Bayvel, "Investigation of bandwidth loading in optical fibre transmission using amplified spontaneous emission noise," Opt. Express **25**(16), 19529–19537 (2017).

26. A. Nespola, S. Straullu, A. Carena, G. Bosco, R. Cigliutti, V. Curri, P. Poggiolini, M. Hirano, Y. Yamamoto, T. Sasaki, J. Bauwelinck, K. Verheyen, and F. Forghieri, "Gn-model validation over seven fiber types in uncompensated pm-16qam nyquist-wdm links," IEEE Photonics Technol. Lett. **26**(2), 206–209 (2014).

27. D. Pilori, F. Forghieri, and G. Bosco, "Residual non-linear phase noise in probabilistically shaped 64-qam optical links," in *OFC*, (2018).

28. Y. Ando, "Statistical analysis of insertion-loss improvement for optical connectors using the orientation method for fiber-core offset," IEEE Photonics Technol. Lett. **3**(10), 939–941 (1991).

29. Telecominfraproject, "Telecominfraproject/oopt-gnpy," (2019).

30. A. Ferrari, M. Filer, K. Balasubramanian, Y. Yin, E. Le Rouzic, J. Kundrát, G. Grammel, G. Galimberti, and V. Curri, "Gnpy: an open source application for physical layer aware open optical networks," J. Opt. Commun. Netw. **12**, C31–C40 (2020).

31. G. Grammel, V. Curri, and J.-L. Auge, "Physical simulation environment of the telecommunications infrastructure project (tip)," in *OFC*, (OSA, 2018), pp. M1D–3.

32. M. Cantono, D. Pilori, A. Ferrari, C. Catanese, J. Thouras, J.-L. Augé, and V. Curri, "On the Interplay of Nonlinear Interference Generation with Stimulated Raman Scattering for QoT Estimation," J. Lightwave Technol. **36**(15), 3131–3141 (2018).

33. A. Ferrari, G. Borraccini, and V. Curri, "Observing the generalized snr statistics induced by gain/loss uncertainties," in *ECOC*, (IEEE, 2019).

34. B. D. Taylor, G. Goldfarb, S. Bandyopadhyay, V. Curri, and H.-J. Schmidtke, "Towards a route planning tool for open optical networks in the telecom infrastructure project," in *OFC/NFOEC*, (2018).

35. C. M. Bishop, *Pattern recognition and machine learning* (springer, 2006).

36. G. Hackeling, *Mastering Machine Learning with scikit-learn* (Packt Publishing Ltd, 2017).

37. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} ({OSDI} 16)*, (2016), pp. 265–283.

38. L. Rokach and O. Z. Maimon, *Data mining with decision trees: theory and applications*, vol. 69 (WS, 2008).

39. L. Breiman, "Random forests," Mach. Learn. **45**(1), 5–32 (2001).

40. X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *AISTATS*, (2010), pp. 249–256.

41. S. Ruder, "An overview of gradient descent optimization algorithms," arXiv preprint arXiv:1609.04747 (2016).

42. C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation functions: Comparison of trends in practice and research for deep learning," arXiv preprint arXiv:1811.03378 (2018).

43. J. Elith, J. R. Leathwick, and T. Hastie, "A working guide to boosted regression trees," J. Animal Ecol. **77**(4), 802–813 (2008).

44. Y. Bengio, "Learning deep architectures for ai," Foundations and Trends Mach. Learn. **2**(1), 1–127 (2009).

45. J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," J. Mach. Learn. Res. **12**, 2121–2159 (2011).

46. H.-T. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir, R. Anil, Z. Haque, L. Hong, V. Jain, X. Liu, and H. Shah, "Wide & deep learning for recommender systems," CoRR **abs/1606.07792** (2016).

47. H. B. McMahan, "Follow-the-regularized-leader and mirror descent: Equivalence theorems and l1 regularization," Tech. rep., Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (2011).

48. J. Kacprzyk and W. Pedrycz, *Springer handbook of computational intelligence* (Springer, 2015).