Towards explainable artificial intelligence in optical networks: the use case of lightpath QoT estimation

Towards Explainable Artificial Intelligence in Optical Networks: The Use Case of Lightpath QoT Estimation

OMRAN AYOUB[1], SEBASTIAN TROIA[2], DAVIDE ANDREOLETTI[1], ANDREA BIANCO[3], MASSIMO TORNATORE[2], SILVIA GIORDANO[1], AND CRISTINA ROTTONDI[3]

[1] *Scuola Universitaria Professionale della Svizzera Italiana, Lugano, Switzerland*
[2] *Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Milan, Italy*
[3] *Dipartimento di Elettronica e Telecomunicazioni, Politecnico di Torino, Turin, Italy*
[*] *Corresponding author: omran.ayoub@supsi.ch*

Compiled November 29, 2022

AI and ML continue to demonstrate substantial capabilities in solving a wide range of Optical Network (ON)-related tasks such as fault management, resource allocation and lightpath Quality of Transmission (QoT) estimation. However, the focus of the research community has been centered on ML models' predictive capabilities, neglecting aspects related to model's understanding, i.e., to interpret how the model reasons and makes its predictions. This lack of transparency hinders the understanding of a model's behavior and prevents operators from judging, and hence, trusting the model's decisions. To mitigate the lack of transparency and trust in ML, eXplainable Artificial Intelligence (XAI) frameworks can be leveraged to explain how a model correlates input features to its outputs.

In this paper, we focus on the application of XAI to lightpath QoT estimation. In particular, we exploit Shapley Additive Explanations (SHAP) as XAI framework. Before presenting our analysis, we provide a brief overview of XAI and SHAP, then discuss the benefits of the application of XAI in networking and survey studies which applied XAI to networking tasks. Then, we model the lightpath QoT estimation problem as a supervised binary classification task, to predict if the value of the Bit Error Rate (BER) associated to a lightpath is below or above a reference acceptability threshold, and train an ML eXtreme Gradient Boosting (XGB) model as classifier. Finally, we demonstrate how to apply SHAP to extract insights about the model and to inspect misclassifications.

http://dx.doi.org/10.1364/ao.XX.XXXXXX

1. INTRODUCTION

Research on the application of Artificial Intelligence (AI) and Machine Learning (ML) in Optical Networks (ON) is relentlessly growing, as AI and ML continue to demonstrate substantial capabilities in solving a wide range of ON-related tasks. ON issues, such as fault management, resource allocation and lightpath Quality of Transmission (QoT) estimation, have been successfully tackled using ML-based approaches [1, 2]. However, the majority of these applications focus on performance, disregarding other features such as ML models' reasoning and understanding. In other words, while research has mostly focused on improving ML models' predictive capabilities, the actual reasons associated with models' decisions are often unknown or disregarded.

As a matter of fact, most research studies adopt complex ML models (as ensemble and/or Deep Learning (DL) methods) that do not expose their internal decision process, motivated by their powerful prediction capabilities, in contrast to the adoption of simpler learning methods, such as Decision Trees (DTs) and linear regression, whose decisions are more interpretable, but whose performance are also less optimized.

The lack of a clear understanding of a model's decision prevents practitioners from interpreting, and hence judging, the model's reasoning prior to its deployment. This is hindering the deployment of reliable and trustworthy ML systems in a sensitive area as optical transport networks.

To mitigate the lack of trust in ML, tools from the field of eXplainable Artificial Intelligence (XAI) can be leveraged to explain

models' decision making, thus enhancing trust in AI-based systems [3, 4]. Specifically, several existing XAI frameworks can be used to provide explanations on a model's behavior, e.g., explaining how a model correlates input features to its output decisions, unveiling hidden interactions among features' values. These explanations permit to debug and verify the correctness of the model's reasoning, e.g., checking that it has correctly captured correlations and dependencies known a priori, such as those provided by analytical models of involved physical phenomena. Furthermore, explanations may help in the identification of the factors that led to wrong decisions, uncovering hidden biases in either the input data or the model itself. Explanations also permit to extract useful insights to improve the understanding of the problem at hand and, in some cases, to reverse-teach domain experts, e.g., as when XAI reveals previously unknown correlations between input features and outputs.

In this paper, we focus on the application of XAI to lightpath QoT estimation. Before demonstrating XAI application to this task, we provide a brief overview of explainability and Shapley Additive Explanations (SHAP) [5], the XAI framework considered in our study. Then, we discuss the benefits of XAI and survey studies which applied XAI to networking tasks, highlighting the advantages XAI can offer to ease the application of AI-based systems for ON management. Finally, we investigate, as a specific use case, XAI for lightpath QoT estimation. Specifically, we show how XAI frameworks can be exploited to analyze features' impact on the model, describe the model's behavior, perform feature selection, and extract knowledge about the QoT estimation problem that could guide domain experts in network design and wrong decisions identification. Though the specific insights gained from applying XAI may strongly depend on the characteristics of the adopted dataset, we have the more general aim of showing the validity of the XAI methodologies independently of the chosen dataset and of the employed learning algorithm.

We summarize the paper contributions as follows:

- We provide an overview on XAI, and particularly on SHAP, the XAI framework applied in our study;
- We survey related work on XAI in networking problems and discuss the benefits of XAI to ON management;
- We show how to apply SHAP to extract explanations that describe the reasoning of the ML model, and how to leverage these explanations to extract insights on the problem at hand;
- We show how to leverage SHAP to inspect misclassified instances, and how to exploit SHAP for feature selection.

The remainder of the paper is organized as follows. Sec. 2 provides background information on explainability and XAI frameworks. Sec. 3 discusses benefits of XAI in networking and surveys related work. Sec. 4 presents our application of SHAP to the problem of lightpath QoT estimation and offers a performance assessment. Finally, Sec. 5 concludes the paper.

2. BACKGROUND ON XAI

A. Explainability

AI explainability refers to understanding an AI-based model's behavior, i.e., how the model operates and how it makes its predictions. In this context, the key objective of XAI frameworks is to provide explanations revealing the factors that influenced the decision, thus providing understanding of the ML-based decision-making process.

Explainability is a process that starts from analyzing the training dataset, in a phase known as *pre-modeling explainability*. Pre-modeling explainability refers to applying exploratory data-analysis techniques to gain a better understanding of the dataset used for model training, as the model behavior is driven by the characteristics of the training dataset. Hence, using XAI for pre-modeling explainability, one can detect undesired biases in the training set.

As for model development, two directions can be considered to achieve an explainable ML-based model: *explainable modeling* and *post-model explainability*.

Explainable modeling refers to using a model that is intrinsically interpretable (*interpretable by-design*) such as, e.g., rule-based models and DTs, in contrast to using complex models (black boxes) that lack interpretability [6], [7], [8]. A drawback of explainable modeling is that it is limited to a number of interpretable ML models, which do not always meet the desired performance targets for the problem at hand[1]. To overcome the above-mentioned limitation, we can adopt post-model explainability, which refers to explaining an already-trained model [11] [12]. Post-model XAI frameworks are applied after the model has taken a decision, i.e., in a post-model (also referred to as *post-hoc*) manner and are discussed in the next subsection.

B. Post-model Explainability

Post-model explainability creates an interpretable ML model (referred to as *surrogate model*) that approximates the original black box model (such as Local Interpretable Model-Agnostic Explanations (LIME) [4]), in contrast to developing models that are inherently interpretable. Consequently, these models provide a description of their own behavior, which in turn explains the behavior of the black box model. For instance, to interpret an ensemble-based model, a logistic regression model can be used as an approximation model to explain decisions boundaries in a local space, i.e., in proximity to the data sample being explained, hence providing a local description of the original model's behavior. Such a description is referred to as a *local* explanation. A local explanation explains a specific observation (i.e., it explains a specific decision taken by a model for one particular data point), indicating how each input feature influenced the model's decision and how impactful it was. When aimed at explaining the model's misclassifications, one can gain insights on, for instance, the motivations behind the model's wrong reasoning.

Another class of explanations is *global* explanations. A global explanation is an aggregation of local explanations that aims to explain the general model behavior. In most applications of post-model explainability, both classes of explanations are necessary to develop a clear understanding of the model's internal mechanics. A main drawback of post-model explainability, however, is that no agreed-upon approach to quantify the quality of explanations currently exists[2]. Instead, evaluating the quality of an explanation strictly depends on the user. In other words, an explanation is judged by a human (domain expert) who examines it and quantifies up to what extent it can be trusted [18, 19].

[1] It is worth-noting that the application of physics-informed AI [9, 10], which embeds the physical knowledge into neural networks, has been proposed recently for optical networks to improve the interpretability of AI systems, however this goes beyond the focus of this work.

[2] Recent works are investigating metrics and frameworks to evaluate the quality of explanations [13–17].

Post-model XAI frameworks can either be model-agnostic, i.e., they can be applied to any ML model, or model-specific, i.e., their application is limited to a specific type of ML models such as, e.g., only to DL models. In our study, as a XAI framework, we rely on Shapley Additive Explanations (SHAP), which is discussed in more details in the next subsection.

C. Shapley Additive Explanations

SHAP is a model-agnostic XAI framework that interprets predictions based on Shapley values used in cooperative game theory. SHAP computes an explanation by calculating the importance value (also referred to as SHAP value) for each feature in a given prediction.

The SHAP value $\phi_i(f,x)$ for a feature i in a given data point x and a ML model f can be calculated as shown in Eqn. 1 [5]:

$$\phi_i(f,x) = \sum_{z' \subseteq x'} \frac{|z'|!(M-|z'|-1)!}{M!}[f_x(z') - f_x(z'/i)] \quad \textbf{(1)}$$

where $|z'|$ is the number of features in z', with $z' \subseteq x'$ represents all z' vectors that are a subset of x'. M represents the number of all input features. The equation sums over all possible coalitions of the set of features of data point x, x', considering the output of the model when feature i is included and when feature i is withheld, where:

- $\frac{|z'|!(M-|z'|-1)!}{M!}$ represents a weight given to each coalition, where $|z'|$ is the cardinality of the set of features in z';
- $f_x(z')$ is the output of model f for data point x of the coalition with feature i;
- $f_x(z'/i)$ is the output of model f for data point x of the coalition without feature i.

SHAP consists of several approaches to calculate the SHAP values, which are either model-agnostic (Shapley sampling values and KernelSHAP) or model-specific (TreeSHAP, DeepSHAP) [5]. In our work, we consider TreeSHAP, an algorithm that finds an exact computation of SHAP values with a lower complexity with respect to other versions of SHAP, by exploiting DTs structures to disaggregate the contribution of each input in a DT or DT-ensemble model [20]. This algorithm is fed with two main inputs, namely, a trained model having an ensemble or a tree structure, and a dataset composed of N samples, each represented by M features. As for the trained model, in this work we employ an eXtreme Gradient Boosting model (XGB). The output of the TreeSHAP algorithm is a matrix with N rows (i.e., one for each sample) and M columns (i.e., one for each feature). The generic ij-th entry of the matrix is the SHAP value that measures how the j-th feature of the i-th input sample impacts on the model's decisions. To perform a local explanation, the SHAP value of a feature for a particular sample is inspected. To perform a global explanation, the SHAP values of all the samples are aggregated and inspected together.

3. XAI FOR NETWORKING

Lack of interpretability may hinder the commercial deployment of AI-based solutions in many fields, including networking. As XAI promises to mitigate this issue, its application in networking has gained traction recently. In this section, we first discuss the benefits XAI can bring to the field of networking, and then we survey some existing works investigating XAI application in generic networking problems.

A. Benefits of XAI

The benefits of XAI in the field of networking are numerous [21]. A list of relevant ones is reported in the following.

Extracting Insights and Discoveries [22] (Insights) As XAI can expose the ML model's reasoning by revealing which (and how) features influence the model's decisions, it can provide insights to better understand the nature of the problem at hand. Explanations can be also used to reverse-teach domain experts, especially when ML models reveal unexpected inputs/outputs correlations.

Increase Trust in the Model (Trust [21]) As explanations can expose the model's reasoning, XAI can be used to debug such reasoning and, in turn, it allows final users to verify if the model behaves as desired, and increase their trust in the model. Here, it is vital to note that XAI can be useful to increase further the trust in ML-based decision making if put in conjunction with uncertainty quantification [23, 24]. Overall, explainability can be used to convince hesitant adopters of AI/ML models.

Decreasing potential costs caused by AI errors (Avoid Costly Errors) Taking actions based on wrong decisions can be very costly. For instance, in a automated failure-cause identification scenario, initiating a wrong repair process might lead to significant and unnecessary expenditures for the operator. In such scenario, explanations can be used by domain experts to double-check the model's decision before taking any action.

Fair Resource Allocation (Fairness [21]) XAI is gaining importance also for resource allocation problems, as the lack of transparency of ML model for resource reservation may represent a risk for reliable operations of the network, potentially leading to revenue losses. Additionally, since service or network operators must take decisions impacting different clients, XAI can be leveraged to ensure that ML-driven decisions are not biased towards any of the clients.

Improve Performance (Performance) Domain experts can analyze explanations of wrong predictions made by the model, which allows to debug reasons for misclassification and then take correcting actions accordingly. For instance, an explanation might reveal if a data point has been mislabeled in the training dataset, which can then be corrected, thus leading to an improvement in performance. Additionally, XAI frameworks such as SHAP can be exploited for feature selection, as they exhibit solid theoretical methods for estimating the features' impact on the model, allowing for the elimination of noisy features, which consequently improves the model's performance.

B. Survey on XAI for Networking

The application of XAI in the field of telecommunication networks is gaining a lot of attention from the scientific community. Recent works highlighted the need of XAI for 5G and 6G networks. In particular, being 6G "human-centric" rather than "machine-centric", in contrast to previous network generations which mainly focused on improving the network performance, 6G networks are expected to leverage massively network automation to manage the network resources, and XAI will be key to increase people's trust towards 6G network automation [25].

In the following, we survey existing studies adopting XAI for network management tasks. Table 1 summarizes these studies, highlighting aspects such as the network environment, the used XAI framework and the benefits achieved by using XAI.

QoE/QoS. We first consider works that focused on enhancing Quality of Experience (QoE) and Quality of Service (QoS) in 5G and 6G networks. Ref. [26] investigates the root-cause of

Table 1. Related work on XAI applications to telecommunications network research field

Task	Ref.	Objective	Network Environment	XAI framework	Dataset	Objectives from XAI
QoE/QoS	[26]	Improve SLA violation prediction in 5G core networks	Emulated 5G core network	SHAP, LIME, Eli5	Data from an emulated 5G core network	Performance, Trust
	[27]	Increase the quality of IoE service delivery	Simulated 5G wireless network	SHAP	5G dataset: [28]	Performance
	[29]	Improve QoE of video streaming applications	Simulated 6G wireless network	Fuzzy Decision Trees	QoS-QoE Dataset: [30]	Interpretability, Performance
Resource Allocation	[31]	Optimize short-term resource reservation for network slicing	Simulated 5G wireless network	SHAP	5G dataset: [32]	Trust, Performance
Traffic Prediction	[33]	Develop an XAI-assisted traffic prediction framework	-	LIME	QoE Dataset: [34][35]	Insights, Performance
Failure Management	[36]	Failure detection	Optical network	SHAP	Data from a real optical transmission network	Insights, Performance
	[37]	Failure localization	Optical network	SHAP	Testbed telemetry data	Insights
	[38]	Lightpath BER prediction	Optical network	SHAP	QoT dataset: [39]	Insights, Performance
	[40]	Failure-cause identification	Microwave network	SHAP, LIME	Real data collected from a microwave network	Trust, Insights
Network Security	[41]	IDSs performance	Simulated military network	Decision Trees	Data from KDD benchmark dataset	Trust, Insights, Performance
	[42]	IDSs performance	Local area network	SHAP	Data from CSE-CIC IDS2018 dataset	Trust, Insights, Performance
Miscellaneous	[43]	XAI-based micro-services framework	Generic SDN-NFV network	Generic XAI framework	-	Insights

Service Level Agreements (SLA) violation prediction for 5G network slicing while leveraging different XAI frameworks, such as SHAP, LIME, and Eli5 to enhance trust in system and improve performance. Specifically, Ref. [26] sets up an emulated 5G core network in order to collect field data, such as latency measurements, and applies an Extreme Gradient Boosting (XGBoost) model to predict latency violations. XAI frameworks are applied to validate and explain the cause of SLA violation predictions. Ref. [27] proposes an XAI-enabled framework for quality-aware Internet of Everything (IoE) service delivery, where a coefficient for each feature is estimated to evaluate their importance, such as reference signal received power (RSRP), reference signal received quality (RSRQ) and signal to interference and noise ratio (SINR). In this context, by exploiting XAI, the service provider can assess the wireless channel quality for each user by maximizing the downlink and uplink data rate of the network for IoE fulfillment and inspect features that drove the model's decisions. In Ref. [29], authors carry out an investigation on the adoption of a XAI models based on Fuzzy Decision Trees (FDTs) for the QoE classification task in video streaming applications. Authors develop a highly explainable FDT-based model as a multi-class classification problem aiming at predicting stall events.

Resource Allocation. In Ref. [31], SHAP is applied in the context of resource allocation (specifically, network slicing) for 5G wireless networks aiming at increasing transparency and enhancing trust and performance.

Traffic prediction. In Ref. [33], authors focus on the application of LIME to interpret unsupervised learning models for traffic prediction. Specifically, given a dataset, the authors use clustering results as input to train a classification model, which is then explained through the application of the LIME framework for the interpretation of results, aiming at extracting useful insights.

Failure Management. Explainability has been also exploited in failure management for optical and microwave networks. Ref. [36] applies SHAP to the problem of failure-cause detection for optical transport network (OTN) boards to identify the features relevant to the identification of failure causes. Similarly, Ref. [37] applies SHAP to investigate the reasoning of ML models in failure localization. Authors use SHAP to discover correlations between the input data and model decisions and compare explanations by considering different feature sets obtained from Optical Signal to Noise Ratio (OSNR) measurements. In our previous work [38], we exploited XAI for lightpath QoT estimation. Specifically, the work investigated the main driving factors that lead ML classification algorithms to correctly predict the Bit Error Rate (BER) associated with the transmission along a perspective lightpath. Finally, Ref. [40] applied SHAP and LIME to the problem of failure-cause identification in microwave networks aiming at extracting insights and enhancing trust in the model. Note that microwave equipment is often situated in areas not easily-reachable (e.g., on top of a hill), thus a repair action based on a wrong failure-cause identification can lead to significant and unnecessary costs for the operator.

Network security. The application of XAI has been widely used to understand and interpret decisions made by network security algorithms [44], such as Intrusion Detection Systems (IDSs). Most of these models are perceived as a black box, hence XAI has become increasingly important to interpret IDSs based on ML models to improve trust management, which role is to understand the impact of malicious data to detect any intrusion in the system. For instance, Ref. [41] exploits XAI to explain the behavior of IDSs. Specifically, Ref. [41] uses DT models to interpret the predictions of attacks made by IDSs using the KDD[3] network

[3] A dataset used for the 3rd International Knowledge Discovery and Data Mining Tools Competition, which was held in conjunction with the 5th International Conference on Knowledge Discovery and Data Mining.

intrusion dataset which includes a wide variety of intrusions simulated in a military network environment. Ref. [42] proposes a SHAP-assisted IDS framework to assess the credibility of predicted cyber-attacks through ML-based models and ensure a high level of accuracy in detecting modern cyber-threats. In particular, Ref. [42] develops an IDS based on a random forest classifier, then applies SHAP to explain the model's predictions. This methodology has been applied to different categories of cyber-attacks enclosed in the CSE-CIC-IDS2018[4] dataset, such as SSH-Brute-Force and Distributed Denial of Service.

Miscellaneous. In addition to the above-mentioned works, other studies, such as Ref. [43], envisioned XAI-based microservices frameworks mapped with the Network Functions Virtualization management and orchestration (NFV-MANO) standard, and give their long-term vision, however without applying any XAI framework.

4. XAI FOR LIGHTPATH QOT ESTIMATION

This section presents our application of XAI to lightpath QoT estimation. We first formulate the problem statement and pose some research questions. Then, we describe the datasets used in our anaylsis. Finally, we present and discuss numerical results and explanations.

A. Problem Statement and Research Questions

We consider the problem of ML-assisted lightpath QoT estimation, modeled as a binary classification. The problem consists in predicting if the value of the Bit Error Rate (BER) associated with the transmission along a lightpath will be below (class 0) or above (class 1) a reference acceptability threshold T. The input to the classification algorithm is a set of features that characterize the lightpath itself (e.g. length, modulation format, spectral width, number of spans, amount of carried traffic, number of links traversed) and its spectral proximity (e.g., the overall spectral occupation of the traversed links and the features of the spectrally-adjacent lightpaths). The ML-based QoT estimation problem has been widely investigated in literature, however without exploiting XAI. Thanks to XAI, operators can discover which features and routing and spectrum/wavelength allocation policies have more significant impact on their ML-based QoT estimation model and leverage these insights to improve the lightpath QoT estimation process (e.g., confirming if the ML-based QoT model replicates the same reasoning of existing analytical models for QoT estimation).

We first train an eXtreme Gradient Boosting (XGB) model to perform automated lightpath QoT estimation and then apply SHAP to extract global and local explanations of the trained model, aiming at demonstrating the application of explainable AI to this problem.

By exploiting XAI, we address the following research questions:

Q1) Which features are most relevant for lightpath QoT estimation across the two data sets?

Q2) How do the features contribute to estimating lightpath QoT and is the model's behavior different across different datasets?

Q3) Can we reduce the feature set by eliminating scarcely relevant features?

Q4) Can we extract insights by examining how features interact and how they impact the model's decisions?

[4]A collaborative project between the Communications Security Establishment (CSE) and the Canadian Institute for Cybersecurity (CIC)

Table 2. Overview of the distribution of some of the features among class 0 and class 1 for the different modulation formats adopted in DSA

	Class 0		Class 1		
	Mod Order		Mod Order		
	32-QAM	64-QAM	16-QAM	32-QAM	64-QAM
Number	97860	273910	66538	475231	407913
Path Len	668-1313	324-709	668-1382	324-903	84-490
Num Spans	12-20	6-12	12-21	6-14	2-7
Freq	192-195	192-195	192-195	192-195	192-195
Num Links	5-9	2-6	5-9	2-7	1-4
LP linerate	112-280	112-336	112-224	112-280	112-336

Q5) How to debug the model's reasoning by explaining its wrong decisions?

B. Dataset Description

For our analysis, we consider two of four datasets that are made publicly available to the research community in Ref. [45],[46], namely, dataset A (DSA), which refers to dataset 03 in Ref. [45] and dataset B (DSB), which refers to dataset 04 in Ref [45][5]. Both datasets comprise a set of active lightpaths over an emulated optical network with 96-channels WDM grid and 37.5 GHz spacing, transceivers operating at a symbol rate of 28 GBd, and dynamic traffic allocated with First Fit (FF) wavelength-assignment policy. DSA adopts the 14-node Telefónica Spain National Network (TSNN) topology with 21 links with online transceiver mode, while DSB adopts the 75-node CONUS topology consisting of 99 links with predefined transceiver mode. Note that, when the predefined transceiver mode is activated, transceivers' configurations in terms symbol rate, modulation format, FEC scheme, channel width, and launch power can only be chosen from a predefined set, whereas when operating in online mode any configuration can be chosen. The physical layer modeling of the simulations is based on non-linear propagation in uncompensated coherent systems. All links adopt Standard Single Mode Fiber (SSMF) spans of 80 km. The fiber power attenuation is set to 0.2 dB/km, whereas the fiber dispersion is assumed to be 17 ps/nm/km. The non-linearity coefficient is set to 1.3 1/W/km. After each span, a C-band Erbium-Doped Fiber Amplifier (EDFA) amplifies the signal to compensate for the span loss. The noise figure and gain of EDFAs are assumed to be 5 dB and 16 dB, respectively. The transmission launch power is fixed to -3 dBm. The OSNR is calculated based on the Gaussian noise model and the NLI noise is calculated using the analytical approximation of the Gaussian noise reference formula for the case of non-identical channels and the BER is computed as a fuction of SNR. Note that the datasets do not take into account features related to launch power, being it fixed for all lightpaths. Each dataset $X \in R^{D\times N}$ includes D samples with $N = 35$ scalar features, among those: lightpath modulation format (*Mod Order*), carrier frequency (*Freq*), length in km (*Path Len*) and number of hops (*Num Hops*) of the path over which the lightpath is provisioned (see Appendix A for a complete list). Unless differently stated, all the 35 features are provided as input to the classification algorithm.

Each sample is associated with a BER value and a binary label. The acceptability threshold on the BER value is set to

[5]See Ref. [46] for a complete description.

Table 3. Overview of the distribution of some of the features among class 0 and class 1 for the different modulation formats adopted in DSB

	Class 0				Class 1					
	Mod Order				Mod Order					
	8-QAM	16-QAM	32-QAM	64-QAM	BPSK	QPSK	8-QAM	16-QAM	32-QAM	64-QAM
Number	45207	26189	10027	3749	580054	335894	278540	45798	17165	5670
Path Len	3466-5359	1679-2639	747-1359	365-719	24-7797	24-7797	24-4109	24-2080	24-1070	24-554
Num Spans	48-76	24-39	12-21	6-12	1-105	1-106	1-56	1-28	1-14	1-7
Freq	192-195	192-195	192-195	192-194	192-195	192-195	192-195	192-195	192-194	192-193
Num Links	4-19	2-14	1-10	1-6	1-25	1-25	1-18	1-11	1-7	1-4
lp linerate	168-168	224-224	280-280	336-336	56-56	112-112	168-168	224-224	280-280	336-336

$T = 0.0038$: samples with BER value above T are labelled as class 0 (i.e., the class of unacceptable lightpath configurations), whereas samples with BER value below T are labelled as class 1 (i.e., the class of acceptable lightpath configurations).

DSA consists of around 1.32 million samples, 28% of them representing lightpaths with BER above T. Three modulation formats are used to provision lightpaths, however no samples for lightpaths with BER above T use 16-QAM. Table 2 provides an overview of DSA, showing the breakdown of the ranges of *Path Len*, *Num Spans*, maximum *Freq*, *Num Links*, and *Lp Linerate* (i.e., lightpath linerate), per modulation format and lightpath class. DSB consists of around 1.34 million samples, 8% of them representing lightpaths with BER above T. Table 3 provides an overview of DSB. Six modulation formats are used to provision lightpaths but no samples with BER above T adopt BPSK and QPSK. This derives from the fact that, in practical deployments, the modulation format assignment is not random: indeed, the operator will most likely assign high modulation formats to short paths and low modulation formats to long paths, such that occurrence of lightpaths with BER above threshold is reduced while spectral efficiency is maximized. As a consequence, the training dataset is biased as it lacks lightpaths characterized, e.g., by low modulation formats over short paths.

C. Results and Discussion

In this section we present numerical results and discuss our application of SHAP to the problem of lightpath QoT estimation. First, we evaluate the performance of the ML model. Then, we use SHAP summary plots to extract global explanations describing the model's general behavior. Further, we perform feature selection to reduce the set of features required for lightpath QoT estimation and use SHAP summary plots to describe the model's behavior with the reduced set of features. Considering DSB, we exploit SHAP dependency plots to inspect the impact of some selected features on the model's predictions. Finally, we focus on local explanations and show how to exploit SHAP's decision plots to explain why the model misclassified a data point of a given class, considering the set of 13 features resulting from the feature selection process.

C.1. Performance Evaluation

We use XGB as ML classification model to estimate the BER feasibility on the two QoT datasets described in section 4.B. The model predicts if the value of the BER will be below (class 0) or above (class 1) the acceptability threshold T. To train the classifier, we split the dataset into training and testing sets considering 80% of the dataset for training and 20% of the dataset for testing. We performed several splits and reported average results over the testing set. Table 4 shows the performance of the classification task by considering accuracy, precision, recall and F1-score. The results show that XGB achieves high performance in both datasets, achieving an average accuracy of 99.7% for DSA and 99.8% for DSB.

Table 4. Performance metrics of the XGB models for both datasets, DSA and DSB

	DSA		DSB	
	Class 0	Class 1	Class 0	Class 1
Accuracy	99.7%		99.8%	
Precision	99.4%	99.7%	99%	100%
Recall	99.4%	99.8%	99%	100%
F1-score	99.4%	99%	99%	100%

C.2. Global Explanations for Model's Behavior

To explain the model's global behavior, we initialize the SHAP explainer with the training data set and then calculate SHAP values of all points in our test dataset and obtain SHAP's *summary plots*. A summary plot is a global explanation of the model's behavior and correlates feature importance (SHAP value) with feature values. Figure 1(a) and 1(b) show summary plots with feature contributions towards class 0[6] considering DSA and DSB, respectively. The y-axis lists features in descending order of importance, whereas the x-axis reports the SHAP value. Each point on the plot represents the value of a feature for a given data point and is associated to a color that quantifies the feature's value in a low-to-high scale. Each point is positioned based on its SHAP value: if a point has a positive SHAP value, it means that it contributes positively towards the prediction of the class being explained; if the plot has a negative SHAP value, it means that it contributes negatively against the prediction of the class being explained. By examining summary plots of each class, we understand the relationship between the value of a feature (color of a point) and the impact on the prediction (SHAP value of a point) towards a particular class of predictions (in this case, class 0).

Let us now elaborate on how to use these explanations to understand the model's behavior (and to address the research question Q1). The plots show that predictions are mainly driven

[6]We show feature contribution (positive or negative) towards class 0 (the class with BER above the acceptability threshold) of all features for all test points considered, irrespective of their labeled class (0 or 1).

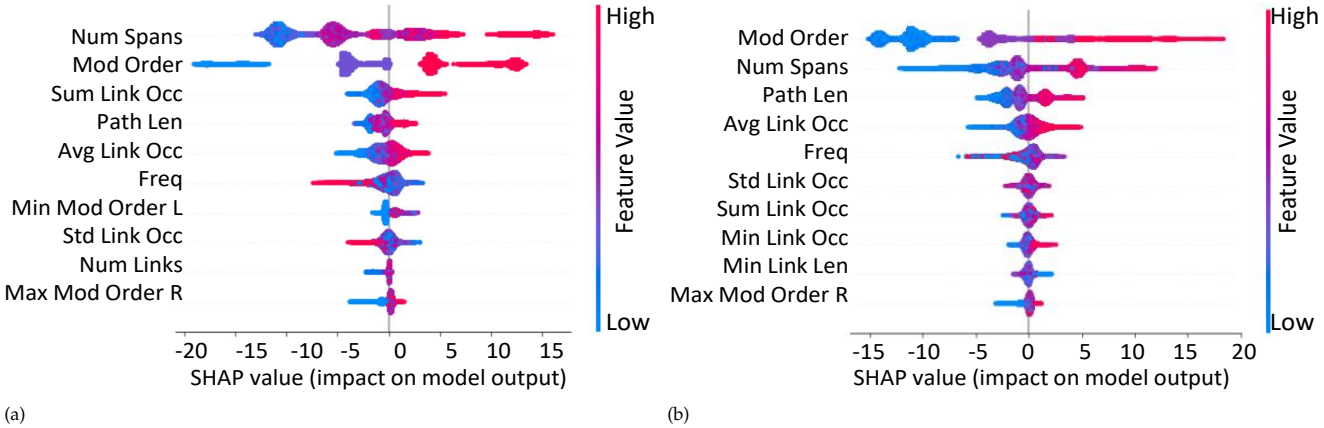


Fig. 1. SHAP summary plots generated from the models trained on (a) DSA and (b) DSB

by *Mod Order* and *Num Spans*, which exhibit significantly larger ranges of SHAP values than all the other features. The plots also show that the most relevant features are common among the two datasets, which are *Mod Order*, *Num Spans*, *Path Len*, *Sum Link Occ*, *Avg Link Occ* and *Freq*, with a slight difference in the order of importance among the two datasets. For instance, *Path Len* and *Freq* are ranked fourth and sixth for DSA, while for DSB they are ranked third and fifth. Indeed, path lengths are more diverse in DSB than in DSA, consequently a larger importance of the *Path Len* feature is expected in the model trained with DSB.

More in detail, we can examine how features impact the model's predictions based on their values, addressing research question Q2. We first consider the summary plot from DSA (Fig. 1(a)). As intuition would suggest, high values of *Num Spans* and *Mod Order* (red points) support positively the prediction towards class 0 (BER unfeasibility), while low-medium values (blue points) of *Num Spans* and *Mod Order* have a negative impact on the model's predictions towards class 0 (i.e., in other words, support BER feasibility). Note that the range of SHAP values of the other features is relatively small with respect to that of *Num Spans* and *Mod Order*, which means that combinations of values of *Num Spans* and *Mod Order* may be enough to predict the lightpath class. For instance, a data point with a high *Num Spans* and *Mod Order* will be predicted as class 0, irrespective of the values of the other features. This also means that a data point with a high *Num Spans* can only be predicted as class 1 if a low-order modulation format is assigned to it. This behavior is expected, as by referring to Tab. 2 we see that data points of class 1 of high-order modulation format (64-QAM) are characterized by very low *Num Spans* (2-7 spans) while those of class 0 have a range of *Num Spans* between 6-12. This clear distinction between the distribution of these two features among the two classes in DSA is captured, and hence relied upon, by the ML model. We remark that such behavior of the model is due to the bias in the training set discussed before, which is induced by the realistic deployment decisions that a network operator takes.

Moving to the case of DSB, which considers a much larger network topology with respect to DSA, we see that *Num Spans* and *Mod Order* show a very similar impact on the model's decisions as in the case of DSA. This happens because, also for DSB, the distribution of values of *Num Spans* and *Mod Order* is highly distinguishable between data points of class 0 and 1 (see Tab. 3), and hence, the model relies significantly on them. However, in this case *Mod Order* becomes the most relevant feature and *Num Spans* is the second most relevant feature. This is due to the fact that DSB exhibits 6 different modulation formats, whereas DSA only 3, which explains the increased importance of the *Mod Order* feature. In addition to *Num Spans* and *Mod Order*, *Path Len* (ranked third in terms of feature importance) has a clear impact on the model's decisions; high values (red points) drive the prediction towards class 0 while low values (blue points) drive the prediction towards class 1.

C.3. SHAP-based Feature Selection

An outcome of SHAP's summary plots is that both models (the one trained on DSA and the one trained on DSB) rely primarily on a small subset of features. We investigate further this outcome and address research question Q3 by performing feature selection based on SHAP values to identifying the minimal subset of features necessary for lightpath QoT estimation, without degrading any of the performance metrics reported in Table 4.

We use a ranking-based feature selection method called Recursive Feature Elimination (RFE). The feature selection process operates as follows:

1. Starting with the initial set of features, a model is trained and the SHAP values are computed, resulting in a feature rank. The accuracy of the model is also evaluated.

2. RFE is applied and the least important feature is eliminated to generate a new subset of features.

3. Steps 1-2 are iteratively repeated, i.e., a new model is trained, its accuracy is recorded, and RFE is newly applied to re-compute features' importance, produce a new rank of features and eliminate the least important one, until only one feature is left,

4. The subset of features yielding the best performance in terms of accuracy is returned.

In terms of the resulting set of features, its cardinality was reduced from 35 to 15 (13) for DSA (DSB), respectively. We note that such reduction does not impact the model's performance, since the same accuracy of 99.7% (as when considering all features) was achieved in DSA, whereas in the case of DSB, the model's accuracy improved by 0.04%. This shows that, in the two considered scenarios, only a small subset of features (one third of the features available in the dataset) is sufficient for an effective lightpath QoT classification.

Table 5. The rank of the selected subset of features for each of the datasets.

Rank	1	2	3	4	5	6	7	8
DSA	Num spans	Mod Order	Sum Link Occ	Path Len	Freq	Avg Link Occ	Num links	Min Mod Order (L)
DSB	Mod Order	Num Spans	Path Len	Freq	Avg Link Occ	Std Link Occ	Sum Link Occ	Min Link Occ

9	10	11	12	13	14	15
Std Link Occ	Max Mod Order (R)	Min Mod Order (R)	Min Link Occ	Max Link Occ	LP Linerate	Avg Link Len
Avg Link Occ	Max Link Occ	Max Mod Order (L)	Min Mod Order (L)	Max Mod Order (R)	NA	NA

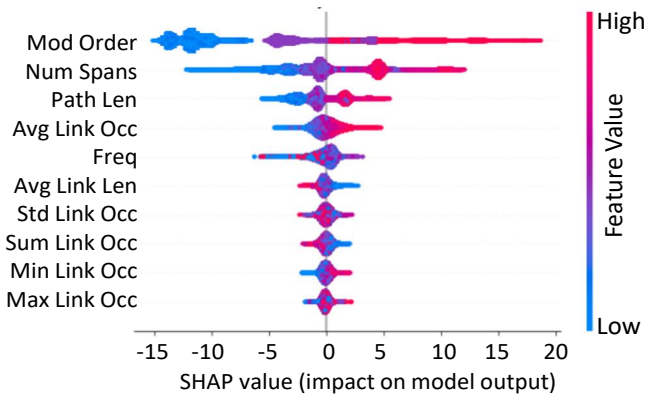


Fig. 2. Summary plot extracted from the model trained on the subset of features of DSB

Tab. 5 reports the subset of features for each of the two cases. The two lists share 13 features, i.e., all of the features selected for DSB are also selected for DSA. For DSA, the two additional features are *LP Linerate* and *Num Links*. In particular, for DSB, only one value of *LP Linerate* is present for each *Mod Order* class, and, hence, it presents no additional information, unlike for DSA, where it varies and overlaps between one *Mod Order* class and another. This is due to the fact that in DSB the transceiver choice was limited to a predefined set of configurations (predefined transceiver mode), whether in DSA there was a much wider variety of possible configurations (online transceiver mode), achievable by means of different combinations of symbol rate, modulation format, FEC scheme and channel width. As for *Num Links*, we speculate that it is more significant for DSA rather than for DSB due to the differences in the characteristics of the two topologies. Specifically, DSA corresponds to the TSNN, which has 52, 148 and 313 km as minimum, average or maximum link length, meaning that the number of links Num Links constituting a path can be informative about how long the path is (in km). On the contrary, DSB corresponds to the 75-node CONUS topology with either relatively short links (24 km, connecting nodes localized in either east or west US) or relatively very long links (crossing central US, as high as 1221 kms), which means that number of links is not informative about the overall path length. This shows that network characteristics, for some specific topologies, can have an important role in the lightpath QoT estimation task.

We now inspect in Fig. 2 SHAP's summary plot when considering DSB from the model in the case of 13 selected features. Compared to the case with all features, we see that the importance ranking of the first 5 features did not change, while it did for some other features. For example, *Avg Link Len* became the 6^{th} most important feature while it was not among the 10 most important features before feature selection. Similarly, *Max Link Occ* became the 10^{th} most impactful feature, while it was not among the 10 most important features before feature selection. As for feature impact, *Min Link Occ* and *Max Link Occ* show, as expected, similar impact on the model's output. For high values, both features show little, yet positive, contribution towards class 0 (bad QoT) while for low values, the features show little, yet negative, contribution towards class 0, meaning they contribute towards class 1 (good QoT). Overall, looking at features' impact and comparing it with that reported in Fig. 1, we can conclude that the model's behavior did not change.

C.4. Dependency Plots

In this subsection, we report and discuss SHAP's *dependency plots* of some selected features. A dependency plot is a scatter plot that shows the effect (the SHAP value) of a single feature on the model's predictions based on the feature's values and on the values of a second selected feature. As an example, consider the *dependency plot* in Fig. 3(a), which shows the impact of *Mod Order* on the model's predictions towards class 0. Each point in the scatter plot represents the SHAP value of *Mod Order* of a single prediction from the dataset. A positive SHAP value pushes the model's decision towards class 0 while a negative SHAP value pushes the model's decision towards class 1. The x-axis represents the feature value of *Mod Order*. In this case, it ranges from 2 to 64. Finally, the second y-axis represents a color scale of the second feature (in this case, *Num Spans*). Analyzing this dependency plot, we can extract information on the impact of *Mod Order* based on its value (x-axis) and based on how the value of *Num Spans* varies (color-scale). Additionally, the dependence plot permits to examine the model's reasoning at feature level, which in turn may allow to detect any bias present in model that is derived from the dataset.

For our analysis, we select and discuss four dependency plots obtained with DSB, shown in Fig. 3. In particular, each plot shows the feature's impact towards class 0, i.e., a positive (negative) SHAP value means the feature pushes model's decision towards class 0 (class 1). We start by discussing, in Fig. 3(a), the dependency plot of *Mod Order* (i.e., the impact of *Mod Order* on the model's predictions) while considering *Num Spans* on the color-scale. First, we see that low values of *Mod Order* (highlighted in box 1) have negative SHAP values (negative impact towards class 0) while high values of *Mod Order* (box 2) have positive SHAP values (positive impact towards class 0). This means that the model correlates low (high) modulation format order with good (bad) lightpath quality. Indeed, low modulation format orders are only present in samples labeled as class 1 (see Tab. 3), and hence, the model learns that bad lightpath quality can never be correlated with low modulation orders. This fact

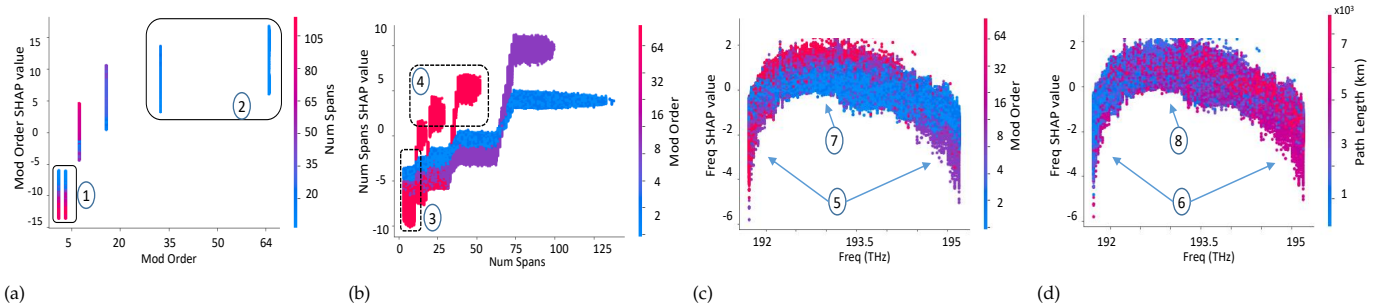


Fig. 3. SHAP dependency plots obtained with DSB of (a) *Mod Order* with its interaction with *Num Spans*, (b) *Num Spans* with its interaction with *Mod Order* and *Freq* with its interaction with (c) *Mod Order* and (d) *Path Len*

shows that the composition of the dataset in terms of modulation formats distribution heavily impacts the model's reasoning. We now discuss the interaction with *Num Spans* (represented by color scale). For low *Mod Order*, we see points of different colors, as low *Mod Order* is assigned to lightpaths routed on short or long routes (see Tab. 3) while for high *Mod Order*, only the blue color is present, as such modulation formats are only used for relatively short paths (hence low *Num Spans*). More in detail, we see that for the cases of low *Mod Order* (box 1), high *Num Spans* drives the model's decision more against class 0 than with low *Num Spans* (red points are higher in magnitude than blue points). In other words, with low *Mod Order*, high *Num Spans* is correlated with better lightpath QoT than low *Num Spans*, which is counter-intuitive, as high *Num Spans* is expected to drive the model's decision towards a worse QoT. The explanation for this behavior lies in the fact that a high *Num Spans*, in dataset DSB, is a characteristic of class 1. In Tab. 3 we see that *Num Spans* ranges up to 106 for data points of class 1 while the maximum *Num Spans* of data points in class 0 is 76. This shows that the model captured this relationship and hence, considers high *Num Spans* combined with low *Mod Order* to be correlated with worse QoT than with low *Num Spans*. Since our XAI analysis has allowed us to identify a bias in the training dataset, a possible recommendation is to carefully analyze the distribution of training samples prior to feeding them to the QoT classifier, in order to avoid such undesirable outcomes.

We further inspect this aspect by examining in Fig. 3(b) the impact of *Num Spans* on the model's decisions while considering its interaction with *Mod Order*, since flipping the two features in the plot allows us to inspect the SHAP value associated to the *Num Spans* feature. The plot shows that relatively low values of *Num Spans* (highlighted in box 3), as expected, drive the model's decisions towards good lightpath QoT (negative SHAP values for class 0). For slightly higher *Num Spans* (15 to 25 spans), and especially when correlated with high *Mod Order* (box 4), we see that *Num Spans* drives the model's decisions towards bad lightpath QoT (class 0). Indeed, such combinations of *Mod Order* and *Num Spans* are present only in class 0, while in class 1 data points with high *Mod Order* have lower *Num Spans*. This explanation shows that the model has captured salient characteristics of the dataset in consideration to distinguish between the two classes.

Based on such outcomes, we distill a fundamental take-home message: when inspecting the reasoning of a trained ML model, we should be aware that its behavior does not necessarily show general validity, nor it always reflects expected trends and correlations. Indeed, as discussed above, in certain circumstances, high-order modulation formats show weaker correlation with class 0 than low-order modulation formats, contrary to what common sense would suggest. Therefore, careful consideration should be adopted when generalizing conclusions obtained through XAI, to avoid dependencies on the composition of training datasets rather than on the nature of the problem at hand.

We now discuss, in Figs. 3(c)-(d), two dependency plots for *Freq*, i.e., the lightpath's wavelength in the WDM grid considering the modulation format order (Fig. 3(c)) and the lightpath length (Fig. 3(d)). The two figures permit to gain insights on the effects of the spectrum allocation policy (the dataset was generated using a First Fit (FF) strategy and contains instances for 6 different traffic loads). We observe that at the two extremes of the spectrum band (pointed at by arrows 5 and 6 in Figs.3(c)-(d) respectively), samples exhibit negative SHAP values (i.e., in favor of lightpath feasibility), thanks to the fact that in those spectrum regions lightpaths suffer less interference from adjacent channels. Conversely, in the center-left frequency range (pointed at by arrows 7 and 8 in Figs.3(c)-(d), respectively), most points have positive SHAP values (i.e., against lightpath feasibility). Indeed, since the FF policy starts scanning for availability of the spectrum from left to right: i) the left side of the spectrum has, on average, higher occupation than the right side; ii) due to fragmentation effects, short lightpaths with high-order modulation formats are typically allocated on the left spectrum side. Indeed, red-colored points indicating high-order modulation formats are mostly confined in the left part of Fig. 3(c), whereas long lightpaths with low-order modulation formats are allocated on the right side (indeed, the majority of blue points, which represent low-order modulation formats, are located in the right part of Fig. 3(c), and in Fig. 3(d) the same portion of the plot contains most of the red points, which indicate long lighpaths). This way, operators may visually inspect how the combined effect of path length and choice of wavelength/modulation format assignment influence the likelihood that the perspective lightpath will exhibit unacceptable QoT. It is worth noting that the above mentioned trends can only be discovered by direct inspection, since none of the 35 features included in the dataset carries explicit information about the adopted spectrum allocation and modulation format assignment criteria.

C.5. Explaining Misclassifications

Finally, we focus on local explanations aiming at demonstrating how to debug the model's behavior for individual instances. We show how to exploit SHAP's *decision plots* to explain why the model misclassified a data point of a given class, considering as set of features the set of 13 features resulting from the feature selection process.

Figures 4(a)-(b) show two instances of model's misclassifi-

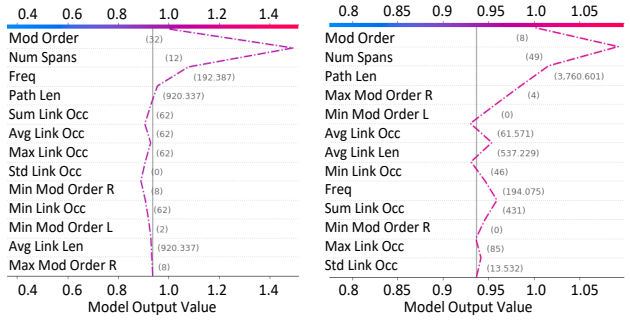
cations where the true label is class 0 but the model wrongly predicted class 1. Decision plots can be read as follows. The x-axis, either at the top or at the bottom, represents the output of the model (in our case, it is either 0 or 1, corresponding to each of the classes of the binary classification problem). The y-axis lists the features in decreasing order of their impact on the model's decision. The plot is centered on the x-axis at the *expected value*[7] of the explainer. In our case, the expected value of the explainer is 0.9, as around 90% of the data points are of class 1 while 10% are of class 0. The curve represents if a feature (considering its value in parenthesis) drove the decision towards class 1 (i.e., pushed the curve to the right) or class 0 (i.e., pushed the curve to the left). In Fig. 4(a), obtained for one misclassified sample of DSB, we see that most of the features (except for *Freq*, *Num Spans* and *Mod Order*) had none or negligible impact on the model's decision (none of the features caused the curve to skid to the left or the right of the vertical line that represents the expected value). *Mod Order*, having a value of 32 (corresponding to 32-QAM), suggested class 0. On the contrary, *Freq* and *Num Spans*, with values of 192.387 and 12, respectively, both drove the model's decision towards class 1, with *Num Spans* showing a significant impact. This means that the relatively low value of *Num Spans* was the reason behind this misclassification. In fact, data points (lightpaths) of class 0 with low *Num Spans* (12 or less) are underrepresented in the dataset, while data points of class 1 with *Num Spans* of such values are abundant in the dataset (see Tab. 3). This shows that low value of *Num Spans*, when combined with high order modulation formats, may cause the model to misclassify data points of class 0 as class 1.

In Fig. 4(b) we see a different example where *Mod Order*, *Num Spans* and *Path Len* are the most impacting features. *Mod Order* (8-QAM) suggested class 0 while *Num Spans* and *Path Len* drove the model's decision towards class 1. It is interesting to see that the impact of *Mod Order* towards class 0 and the impact of *Num Spans* and *Path Len* towards class 1 are equal. In other words, the influence of these features balanced each other. In this case, other features emerge to be more decisive. In particular, *Max Mod Order R* (equal to 4, corresponding to QPSK) and *Min Mod Order L* (equal to 0, meaning no left lighpath exists), had a significant influence towards class 1. The explanation shows that the fact that *Max Mod Order R* is QPSK (which heavily corresponds to class 1), contributed to classifying the lightpath in consideration as class 1. In other words, the model reasoned as follows: the fact that neighboring lightpaths enjoy good QoT, having QPSK as modulation format, which we have seen to be correlated by the model to good QoT, influenced the model's decision regarding the lightpath in consideration, classifying it as a lightpath of class 1 (with QoT lower than T). This observation highlights a specific case of misclassification. Domain experts can analyze several other misclassifications to see if such reasoning dominates the model's behavior or not.

5. CONCLUSIONS

We investigated the use of eXplainable Artificial Intelligence (XAI) for Machine Learning (ML)-based lightpath Quality-of-Transmission (QoT) estimation. We modeled the problem as a supervised binary classification problem and developed an ML eXtreme Gradient Boosting (XGB) model to solve it. We considered two datasets which correspond to different network topologies. After solving the supervised classification problem, we demonstrated our application of XAI relying on Shapley Additive exPlanations (SHAP) as a framework.

We first extracted global explanations of the models trained on each of the data set using SHAP summary plots. The explanations show that only few features are relevant to the QoT estimation problem. Based on these outcomes, we then performed SHAP-based feature selection aiming at identifying a subset of features that is sufficient for the lightpath QoT estimation task. Results show that set of features can be reduced to 13-15 features (instead of 35 features originally adopted) without compromising on performance. Moreover, SHAP dependency plots show that the model's behavior suffers from inconsistencies due to biases in the dataset. Thus, dataset composition should be analyzed carefully, as biased training datasets may cause not-easy-to-capture inconsistencies. Finally, we demonstrated how to exploit SHAP's decision plots to identify reasons behind model's misclassifications.



(a) (b)

Fig. 4. Two example local explanations of model's misclassification, drawn from DSB

REFERENCES

1. F. Musumeci, C. Rottondi, A. Nag, I. Macaluso, D. Zibar, M. Ruffini, and M. Tornatore, "An overview on application of machine learning techniques in optical networks," IEEE Commun. Surv. & Tutorials **21**, 1383–1408 (2018).
2. R. Gu, Z. Yang, and Y. Ji, "Machine learning for intelligent optical networks: A comprehensive survey," J. Netw. Comput. Appl. **157**, 102576 (2020).
3. A. Adadi and M. Berrada, "Peeking inside the black-box: a survey on explainable artificial intelligence (xai)," IEEE access **6**, 52138–52160 (2018).
4. M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why should I trust you?" Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, (2016), pp. 1135–1144.
5. S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," Adv. neural information processing systems **30** (2017).
6. T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," Artif. intelligence **267**, 1–38 (2019).
7. M. Du, N. Liu, and X. Hu, "Techniques for interpretable machine learning," Commun. ACM **63**, 68–77 (2019).
8. S. Chakraborty, R. Tomsett, R. Raghavendra, D. Harborne, M. Alzantot, F. Cerutti, M. Srivastava, A. Preece, S. Julier, R. M. Rao *et al.*, "Interpretability of deep learning models: A survey of results," in *2017 IEEE smartworld, ubiquitous intelligence & computing, advanced & trusted computed, scalable computing & communications, cloud & big*

[7]The expected value of a SHAP explainer is a value that represents the average prediction of the classifier, or, in other words, the value that would be predicted by always predicting class 1, irrespective of any feature value [5].

data computing, Internet of people and smart city innovation (smartworld/SCALCOM/UIC/ATC/CBDcom/IOP/SCI), (IEEE, 2017), pp. 1–6.
9. D. Wang, X. Jiang, Y. Song, M. Fu, Z. Zhang, X. Chen, and M. Zhang, "Applications of physics-informed neural network for optical fiber communications," IEEE Commun. Mag. **60**, 32–37 (2022).
10. X. Jiang, D. Wang, Q. Fan, M. Zhang, C. Lu, and A. P. T. Lau, "Physics-informed neural network for nonlinear dynamics in fiber optics," arXiv preprint arXiv:2109.00526 (2021).
11. D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, and K.-R. Müller, "How to explain individual classification decisions," The J. Mach. Learn. Res. **11**, 1803–1831 (2010).
12. W. Samek, T. Wiegand, and K.-R. Müller, "Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models," arXiv preprint arXiv:1708.08296 (2017).
13. S. Mohseni, N. Zarei, and E. D. Ragan, "A multidisciplinary survey and framework for design and evaluation of explainable ai systems," ACM Transactions on Interact. Intell. Syst. (TiiS) **11**, 1–45 (2021).
14. J. Zhou, A. H. Gandomi, F. Chen, and A. Holzinger, "Evaluating the quality of machine learning explanations: A survey on methods and metrics," Electronics **10**, 593 (2021).
15. G. Vilone and L. Longo, "Notions of explainability and evaluation approaches for explainable artificial intelligence," Inf. Fusion **76**, 89–106 (2021).
16. S. R. Islam, W. Eberle, and S. K. Ghafoor, "Towards quantification of explainability in explainable artificial intelligence methods," in *The thirty-third international flairs conference,* (2020).
17. A. Rosenfeld, "Better metrics for evaluating explainable artificial intelligence," in *Proceedings of the 20th international conference on autonomous agents and multiagent systems,* (2021), pp. 45–50.
18. J. van der Waa, E. Nieuwburg, A. Cremers, and M. Neerincx, "Evaluating xai: A comparison of rule-based and example-based explanations," Artif. Intell. **291**, 103404 (2021).
19. S. Mohseni, J. E. Block, and E. D. Ragan, "A human-grounded evaluation benchmark for local explanations of machine learning," arXiv preprint arXiv:1801.05075 (2018).
20. S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "From local explanations to global understanding with explainable ai for trees," Nat. machine intelligence **2**, 56–67 (2020).
21. A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins *et al.*, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," Inf. fusion **58**, 82–115 (2020).
22. R. Roscher, B. Bohn, M. F. Duarte, and J. Garcke, "Explainable machine learning for scientific insights and discoveries," Ieee Access **8**, 42200–42216 (2020).
23. M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya *et al.*, "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," Inf. Fusion **76**, 243–297 (2021).
24. H. Maryam, T. Panayiotou, and G. Ellinas, "Representing uncertainty in deep qot models," in *2022 20th Mediterranean Communication and Computer Networking Conference (MedComNet),* (IEEE, 2022), pp. 113–121.
25. S. Wang, M. A. Qureshi, L. Miralles-Pechuaán, T. Huynh-The, T. R. Gadekallu, and M. Liyanage, "Explainable ai for b5g/6g: Technical aspects, use cases, and research challenges," arXiv preprint arXiv:2112.04698 (2021).
26. A. Terra, R. Inam, S. Baskaran, P. Batista, I. Burdick, and E. Fersman, "Explainability methods for identifying root-cause of sla violation prediction in 5g network," in *GLOBECOM 2020-2020 IEEE Global Communications Conference,* (IEEE, 2020), pp. 1–7.
27. M. Munir, S.-B. Park, C. S. Hong *et al.*, "An explainable artificial intelligence framework for quality-aware ioe service delivery," arXiv preprint arXiv:2201.10822 (2022).
28. D. Raca, D. Leahy, C. J. Sreenan, and J. J. Quinlan, "Beyond throughput, the next generation: a 5g dataset with channel and context metrics," in *Proceedings of the 11th ACM multimedia systems conference,* (2020), pp. 303–308.
29. A. Renda, P. Ducange, G. Gallo, and F. Marcelloni, "Xai models for quality of experience prediction in wireless networks," in *2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE),* (IEEE, 2021), pp. 1–6.
30. V. Vasilev, J. Leguay, S. Paris, L. Maggi, and M. Debbah, "Predicting qoe factors with machine learning," in *2018 IEEE International Conference on Communications (ICC),* (IEEE, 2018), pp. 1–6.
31. P. Barnard, I. Macaluso, N. Marchetti, and L. A. DaSilva, "Resource reservation in sliced networks: An explainable artificial intelligence (xai) approach," (TechRxiv, 2021).
32. F. A. Silva, A. C. Domingues, and T. R. B. Silva, "Discovering mobile application usage patterns from a large-scale dataset," ACM Transactions on Knowl. Discov. from Data (TKDD) **12**, 1–36 (2018).
33. A. Morichetta, P. Casas, and M. Mellia, "Explain-it: Towards explainable ai for unsupervised network traffic analysis," in *Proceedings of the 3rd ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks,* (2019), pp. 22–28.
34. M. Seufert, P. Casas, N. Wehner, L. Gang, and K. Li, "Features that matter: Feature selection for on-line stalling prediction in encrypted video streaming," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS),* (IEEE, 2019), pp. 688–695.
35. M. Seufert, P. Casas, N. Wehner, L. Gang, and K. Li, "Stream-based machine learning for real-time qoe analysis of encrypted video streaming traffic," in *2019 22nd Conference on innovation in clouds, internet and networks and workshops (ICIN),* (IEEE, 2019), pp. 76–81.
36. C. Zhang, D. Wang, L. Wang, L. Guan, H. Yang, Z. Zhang, X. Chen, and M. Zhang, "Cause-aware failure detection using an interpretable xgboost for optical networks," Opt. Express **29**, 31974–31992 (2021).
37. O. Karandin, O. Ayoub, F. Musumeci, Y. Hirota, Y. Awaji, and M. Tornatore, "If not here, there. explaining machine learning models for fault localization in optical networks," in *2022 International Conference on Optical Network Design and Modeling (ONDM),* (2022), pp. 1–3.
38. O. Ayoub, A. Bianco, D. Andreoletti, S. Troia, S. Giordano, and C. Rottondi, "On the application of explainable artificial intelligence to lightpath qot estimation," in *2022 Optical Fiber Communications Conference and Exhibition (OFC),* (2022), pp. 1–3.
39. P. Safari, B. Shariati, G. Bergk, and J. K. Fischer, "Deep convolutional neural network for network-wide qot estimation," in *Optical Fiber Communication Conference,* (Optical Society of America, 2021), pp. Th4J–3.
40. O. Ayoub, F. Musumeci, F. Ezzeddine, C. Passera, and M. Tornatore, "On using explainable artificial intelligence for failure identification in microwave networks," in *2022 25th Conference on Innovation in Clouds, Internet and Networks (ICIN),* (2022), pp. 48–55.
41. B. Mahbooba, M. Timilsina, R. Sahal, and M. Serrano, "Explainable artificial intelligence (xai) to enhance trust management in intrusion detection systems using decision tree model," Complexity **2021** (2021).
42. S. Wali and I. Khan, "Explainable ai and random forest based reliable intrusion detection system," (TechRxiv, 2021).
43. S. Sharma, A. Nag, L. Cordeiro, O. Ayoub, M. Tornatore, and M. Nekovee, "Towards explainable artificial intelligence for network function virtualization," in *Proceedings of the 16th International Conference on Emerging Networking EXperiments and Technologies,* (2020), pp. 558–559.
44. T. Senevirathna, Z. Salazar, V. H. La, S. Marchal, B. Siniarski, M. Liyanage, and S. Wang, "A survey on xai for beyond 5g security: Technical aspects, use cases, challenges and research directions," arXiv preprint arXiv:2204.12822 (2022).
45. "https://www.hhi.fraunhofer.de/en/departments/pn/products-and-solutions/qot-dataset-collection.html," .
46. G. Bergk, B. Shariati, P. Safari, and J. K. Fischer, "Ml-assisted qot estimation: a dataset collection and data visualization for dataset quality evaluation," J. Opt. Commun. Netw. **14**, 43–55 (2022).

A. APPENDIX A

Table . List of features used to represent the lightpath within the considered datasets, along with their units of measure and relative descriptions

Feature	Description
Path Len (km)	Length of the lightpath
Avg Link Len (km)	Average length of the links that compose the lightpath
Min Link Len (km)	Minimum length of the links that compose the lightpath
Max Link Len (km)	Maximum length of the links that compose the lightpath
Num Links	Number of links that compose the lightpath
Num Spans	Number of spans the lightpath traverses
Freq (THz)	Central carrier frequency of the lightpath
Mod Order	Cardinality of the modulation Format
Min Mod Order L	Minimum cardinality of the modulation format among the left spectrally-adjacent lightpaths, along all traversed links
Min Mod Order R	Minimum cardinality of the modulation format among the right spectrally-adjacent lightpaths, along all traversed links
Max Mod Order L	Maximum cardinality of the modulation format among the left spectrally-adjacent lightpaths, along all traversed links
Max Mod Order R	Maximum cardinality of the modulation format among the right spectrally-adjacent lightpaths, along all traversed links
LP Linerate (Gb/s)	Line rate of the lightpath
Min LP Linerate L (Gb/s)	Minimum line rate among the left spectrally-adjacent lightpaths, along all traversed links
Max LP Linerate L (Gb/s)	Maximum line rate among the left spectrally-adjacent lightpaths, along all traversed links
Min LP Linerate R (Gb/s)	Minimum line rate among the right spectrally-adjacent lightpaths, along all traversed links
Max LP Linerate R (Gb/s)	Maximum line rate among the right spectrally-adjacent lightpaths, along all traversed links
Conn Linerate (Gb/s)	Line rate of the connection
Min Link Occ	Minimum spectral occupation of the links the lightpath traverses
Max Link Occ	Maximum spectral occupation of the links the lightpath traverses
Avg Link Occ	Average spectral occupation of the links the lightpath traverses
Std Link Occ	Standard deviation of the spectral occupation of the links the lightpath traverses
Sum Link Occ	Sum of the spectral occupation of the links the lightpath traverses
Num Channels	Total number of active channels along the links the lightpath traverses
Min Inter BER	Minimum BER of interfering lightpaths
Max Inter BER	Maximum BER of interfering lightpaths
Avg Inter BER	Average BER of interfering lightpaths
Min BER L	Minimum BER among the left spectrally-adjacent lightpaths, along all traversed links
Max BER L	Maximum BER among the left spectrally-adjacent lightpaths, along all traversed links
Min BER R	Minimum BER among the right spectrally-adjacent lightpaths, along all traversed links
Max BER R	Maximum BER among the right spectrally-adjacent lightpaths, along all traversed links