Using Active Learning to Decrease Probes for QoT Estimation in Optical Networks

Dario Azzimonti[1], **Cristina Rottondi**[1] **and Massimo Tornatore**[2]

[1] *Dalle Molle Institute for Artificial Intelligence, Lugano, Switzerland*
[2] *Dept. of Electronics, Information, and Bioengineering, Politecnico di Milano, Italy*
{dario.azzimonti,cristina.rottondi}@supsi.ch, massimo.tornatore@polimi.it

Abstract: We use active learning to reduce the number of probes needed for machine-learning-based QoT estimation. When building an estimation model based on Gaussian processes, only QoT instances that minimize estimation uncertainty are iteratively requested.

OCIS codes: 060.4250 Fiber optics and optical communications, Networks; 060.4510 Fiber optics and optical communications, Optical communications

1. Introduction

Forecasting the Quality of Transmission (QoT) of a lightpath before its establishment is a critical procedure for design and management of optical networks. Recently, Machine Learning (ML) techniques for QoT estimation [1,2] have been proposed as effective alternative to approximated mathematical formulas, as the Gaussian Noise Model, which adopt conservative approaches to compensate for model simplifications and inaccuracies. Such ML methods are typically based on supervised learning techniques that require a large amount of data (e.g., Bit Error Rate - BER - or Optical Signal to Noise Ratio - OSNR - samples) collected at the receivers of already-deployed ligthpaths [3] to build "training sets" of data, from which ML can extract the knowledge to estimate the QoT of future lightpaths (i.e., predicting whether their BER will exceed a given system threshold). Unfortunately, to ensure high estimation accuracy, the training-set size of supervised learning methods proposed so far shall contain data for a very large number of lightpaths (up to 10's of thousands) [2], but, in real network deployments, the number of monitored/monitorable ligthpaths is limited by practical considerations (insufficient telemetry, old legacy equipment, etc.). In a general sense, monitored data might be expensive to acquire and shall be extracted/queried only when necessary. This is especially true for lightpaths with above-threshold BER (i.e., malfunctioning lightpaths), which are unlikely to be observed during network operation due to the conservative system-design strategies adopted to guarantee transmission quality. Samples with above-threshold BERs can be acquired by deploying probe lightpaths [2], but at the cost of wasting equipment and spectral resources.

How to provide accurate ML-based QoT predictions in presence of small training sets is an important and almost-unexplored research issue. In this paper, we propose an *Active Learning* (AL) method that works on top of a ML predictor based on Gaussian Processes (GP). After an initial training with a limited number of instances, the proposed AL algorithm iteratively asks to collect only few selected training samples with specific characteristics, with the intent of minimizing the number of required samples. In particular, samples that minimize a specifically tailored acquisition function will be sought for.

In the rest of the paper, we describe the proposed AL solution (Section 2) and we numerically assess its accuracy, showing that we can obtain higher accuracy with much less training instances (Section 3).

2. Proposed Active Learning Solution for QoT Estimation

2.1. *Gaussian processes*

Gaussian processes (GP) are a state-of-the-art approach for regression and classification, which can be considered as a Bayesian implementation of kernel methods. A GP is completely characterized by a mean function m and a positive definite covariance kernel k defined on the input space, which are chosen before observing the data and thus encode our prior knowledge. The kernel k is often chosen from a parametric family, such as the squared exponential or Matérn family and depends on few hyper-parameters θ that encode the scale of the output and the characteristic length scale of each input (see [6], chapter 1). Given the data, we can compute analytically the posterior mean and covariance kernel, thus providing a trained fully probabilistic model.

2.2. *GP-based QoT estimator*

Our GP is trained using historical BER values associated to 5 features: modulation format, traffic volume, lightpath length, number of links traversed by the lightpath and length of the longest traversed link. We normalize the

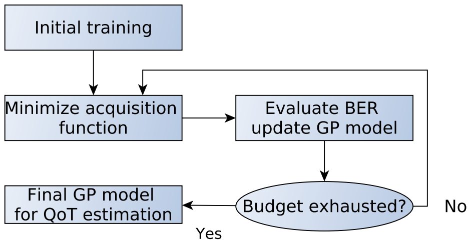
Initial training
Minimize acquisition function
Evaluate BER update GP model
Budget exhausted?
No
Yes
Final GP model for QoT estimation

Fig. 1: The active learning framework

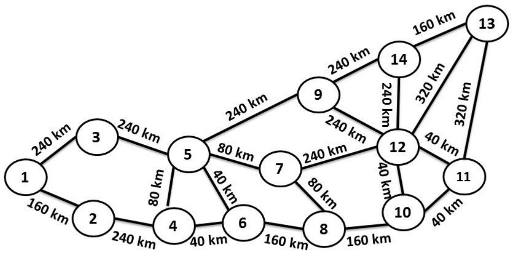

Fig. 2: Japan network topology

5 input features to the unit hypercube $\mathbb{X} = [0,1]^5$ and we consider the function $f_{BER} : \mathbf{x} \in \mathbb{X} \subset \mathbb{R}^5 \to \mathbb{R}$. We assume that observed BER measurements y are not a deterministic function of the features, but are perturbed by a normally-distributed noise, i.e., $y = f_{BER}(\mathbf{x}) + \varepsilon$, where $\varepsilon \sim N(0, \sigma_N^2)$, to account for time-varying penalties affecting transmission and for inter-channel crosstalk caused by adjacent lightpaths.

We assume that f_{BER} is a realization of a GP with a prior constant mean function $m(\mathbf{x})$, estimated from the data, and prior covariance kernel $k_\theta(\mathbf{x}, \mathbf{x}')$, according to [4]. We fix an initial training set $\mathbf{X}_{\ell_0} = \{\mathbf{x}_1, \ldots, \mathbf{x}_{\ell_0}\} \subset \mathbb{X}$ and we evaluate f_{BER} at $\mathbf{X}_{\ell_0}$ obtaining the vector $\mathbf{f}_{\ell_0}$. This training set is first used to estimate the covariance hyper-parameters θ, by maximizing the likelihood of the GP model (a step automatically done by most GP toolboxes). We then obtain the posterior mean and covariance with the GP update formulae (see, e.g. [6]).

The trained model provides an estimator for f_{BER} with the posterior GP mean and an estimator for the *probability of excursion*. This quantity, defined as $p(\mathbf{x}) = P(f_{BER}(\mathbf{x}) > T)$, indicates the probability that $f_{BER}(\mathbf{x})$ exceeds a BER threshold T fixed in advance and it can be considered as the output probability of a binary classifier. Given the ℓ instances of the initial training set, the excursion probability at any $\mathbf{x} \in \mathbb{X}$ can be computed with $p_\ell(\mathbf{x}) = \Phi((m_\ell(\mathbf{x}) - T)/\sqrt{k_\ell(\mathbf{x}, \mathbf{x})})$, where Φ is the c.d.f. of a standard Gaussian variable, m_ℓ and k_ℓ are the GP mean and covariance trained on ℓ samples. Given a discrimination threshold γ, we can classify an instance $\mathbf{x}$ as above T if $p_\ell(\mathbf{x}) > \gamma$ or below T otherwise. There are several techniques to choose γ, such as fixing a reference value (e.g. $\gamma = 0.5$) or by looking at the ROC curve. Moreover, $p_\ell(\mathbf{x})$ will be used during the AL phase to explore the feature space in order to find instances that lead to near-to-threshold BER values.

2.3. *Active Learning*

The aim of the AL phase is to expand the initial training set with additional instances that exhibit near-to-threshold BER values. By following [5], we can select the next training instance $X_{\ell+1}$ by minimizing the *acquisition function* $J_\ell(\mathbf{x})$ defined as $J_\ell(\mathbf{x}) := \mathbb{E}_\ell \left[\int_{\mathbb{X}} p_{\ell+1}(z)(1 - p_{\ell+1}(z)) dz \mid X_{\ell+1} = \mathbf{x} \right]$, where $\mathbb{E}_\ell$ denotes the conditional expectation given the values $\mathbf{f}_\ell$ at $\mathbf{X}_\ell$ and we are further conditioning the next unknown instance to be $X_{\ell+1} = \mathbf{x}$. The integrand in J_ℓ is the variance of the indicator $\mathbf{1}_{f_{BER}(\mathbf{x}) > T}$ and it is linked to the probability of misclassification $\tau_\ell(x) = \min(p_\ell(\mathbf{x}), 1 - p_\ell(\mathbf{x}))$, [5]. By minimizing J_ℓ we find the instance $\mathbf{x}_{\ell+1}$ that minimizes the integrated probability of misclassification at the next step $(\ell + 1)$. The acquisition function J_ℓ has a semi-analytical expression which allows for a fast optimization, see [7]. Once the next training instance $\mathbf{x}_{\ell+1}$ is selected, we assume that a probe lightpath characterized by the features of $\mathbf{x}_{\ell+1}$ is deployed and that its BER is measured, so that we can evaluate f_{BER} at point $\mathbf{x}_{\ell+1}$ and update the GP model. Note that, since the objective function J_ℓ is constrained to return only lightpath lengths and link lengths within the values allowed by the considered network topology, we can always associate a probe lightpath to a set of features $x_{\ell+1}$. This operation is repeated until a predefined limit on the number of iterations is reached or until the acquisition function value drops below a certain tolerance. Fig. 1 represents the adopted AL solution in a block diagram.

3. Results

We now apply the proposed AL solution for QoT estimation of future lightpaths to be deployed within the Japan network topology depicted in Fig. 2 and evaluate its accuracy in terms of AUC. Note that the GP model is a regression model: to compute the AUC, classification is achieved by binarizing the regression output, i.e., by thresholding the probability of excursion p_ℓ.

Dataset generation. To generate synthetic data we use the QTool described in [2]: on input of a candidate lightpath and modulation format, it calculates the BER as a function of the signal-to-noise ratio (SNR) measured at the input of the channel decoder. The Qtool implements the approximated AWGN model of dispersion uncompensated transmission over single mode fibers, assuming the usage of a flexi-grid with 12.5 GHz slice width and elastic transceivers operating at 28 Gbaud with optical bandwidth of 37.5 GHz, using one modulation format among dual polarization (DP)-BPSK, DP-QPSK and DP-n-QAM, with $n = 8, 16, 32, 64$. Traffic demands exceeding the capacity of a single transceiver are accommodated in superchannels containing multiple adjacent transceivers. The

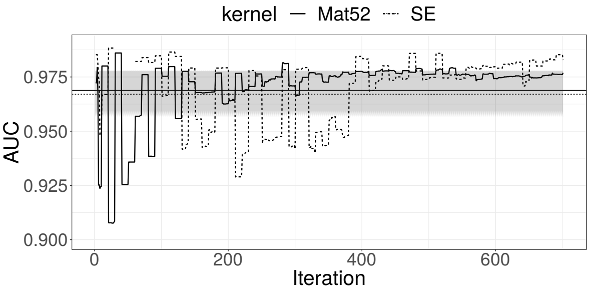

Fig. 3: Training setting $g1$: AUC values versus iteration, offline supervised learning as horizontal line.

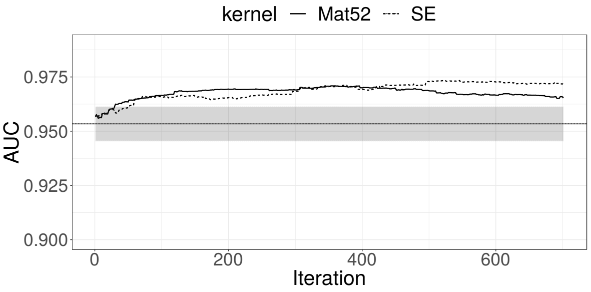

Fig. 4: Training setting $g2$: AUC values versus iteration, offline supervised learning as horizontal line.

Qtool also simulates time-varying impairments by adding randomly-distributed penalties. For the construction of the initial training set, we consider ℓ_0 instances selected in two settings: $g1$) we generated an instance by randomly choosing a source-destination node pair, associating a traffic demand uniformly selected in the range $[50-500]$ Gbps with 50 Gbps granularity and evaluating the BER with the Qtool. We set the BER threshold to $T = 4 \cdot 10^{-3}$ and discarded instances with above-threshold BER. The procedure was repeated until ℓ_0 samples with BER $\leq T$ were obtained; $g2$) the same procedure as in setting $g1$ was iterated ℓ_0 times, without discarding any instance. The setting $g1$ models a realistic historical dataset in which only ligthpaths with below-threshold BER were observed during network operation, the setting $g2$ models an ideal dataset that also includes instances with BER above T. The test set was constructed by generating a separate set of $E = 2000$ instances, as in setting $g2$.

Numerical assessment. We use an initial training set of $\ell_0 = 200$ samples (generated either with setting $g1$ or $g2$) and repeat up to 700 AL iterations, adding one instance at a time. We compare the AL approach to a standard non-active (non-AL) supervised ML approach where training is performed over a dataset of $\ell' = 900$ samples generated by taking 200 samples either with setting $g1$ or $g2$ and 700 samples uniformly as in $g2$. We consider two kernel functions k: a Matérn covariance with smoothness parameter $\nu = 5/2$ (Mat52) and a squared exponential kernel (SE) (see [6], chapter 4). Figure 3 shows AUC values for the training setting $g1$. The performance averaged over 20 runs of the standard offline ML is shown with an horizontal line, and the 90% confidence interval is reported as shaded area. With the AL approach, the AUC obtained with the initial training (i.e. with $\ell_0 = 200$ samples) provides initially AUC=0.5 (not shown in the figure) because no training instance with BER above T was included in the training set. However, when the AL phase starts, the AUC immediately increases, quickly surpassing the AUC of the non-AL approach after 200 iterations with the Mat52 (resp 400 iterations with the SE kernel), thus saving 500 samples (resp 300) w.r.t. the non-AL approach. Figure 4 shows the results for the training setting $g2$. After few tens of iterations the AL approach exceeds the upper side of the 90% confidence interval of the AUC obtained with the non-AL approach, thus leading to a significant reduction in the training set size (around 250 samples instead of 900) even in the case when the initial training set contains both above- and below-threshold BER samples.

4. Conclusions

This paper explores how active learning can be used for QoT estimation when the number of lightpaths available to collect BER training instances is limited. We obtained reductions of at least 33% in the training set size needed to achieve satisfactory classification performance, w.r.t. a standard offline supervised learning approach. As future work, the use of kernels especially developed for BER function regression could bring substantial improvements, whereas more recent, safe learning objective functions [8] could improve the stability of the results.

References

1. T. Panayiotou et al., "Performance analysis of a data-driven quality-of-transmission decision approach on a dynamic multicast-capable metro optical network," *IEEE/OSA J. of Optical Comm. and Netw.*, vol. 9, no. 1, pp. 98–108, Jan. 2017.
2. C. Rottondi et al., "Machine-learning method for quality of transmission prediction of unestablished lightpaths," *IEEE/OSA J. of Optical Comm. and Netw.*, vol. 10, no. 2, pp. A286–A297, Feb 2018.
3. K. Christodoulopoulos et al., "ORCHESTRA-Optical performance monitoring enabling flexible networking," Proc. ICTON 2015.
4. T. Santner et al., "The Design and Analysis of Computer Experiments," *Springer-Verlag New York*, 2003.
5. J. Bect et al. "Sequential design of computer experiments for the estimation of a probability of failure," *Statist. Comput.*, vol. 22, no. 3, pp. 773–793, 2012.
6. C.E. Rasmussen and C. K. I. Williams "Gaussian Processes for Machine Learning," *The MIT Press*, 2006.
7. C. Chevalier et al. "Fast Parallel Kriging-Based Stepwise Uncertainty Reduction With Application to the Identification of an Excursion Set," *Technometrics*, vol. 56, no. 4, pp. 455–465, 2014.
8. D. Azzimonti et al. "Adaptive design of experiments for conservative estimation of excursion sets," *arXiv:1611.07256*, 2018+.