Machine-Learning-as-a-Service for Optical Network Automation

Citation for the original published paper (version of record):

Natalino Da Silva, C., Mohammadiha, N., Panahi, A. (2023). Machine-Learning-as-a-Service for Optical Network Automation. Optical Fiber Communication Conference and Exhibition.
http://dx.doi.org/10.23919/OFC49934.2023.10116794

Machine-Learning-as-a-Service for Optical Network Automation

Carlos Natalino,[1,*] Nasser Mohammadiha,[2,3] Ashkan Panahi[3]

[1] *Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden*
[2] *Department of Computer Science and Engineering, Chalmers University of Technology, Gothenburg, Sweden*
[3] *Ericsson Research, Gothenburg, Sweden*

[*] *carlos.natalino@chalmers.se*

Abstract: MLaaS is introduced in the context of optical networks, and an architecture to take advantage of its potential is proposed. A use case of QoT classification using MLaaS techniques is benchmarked against state-of-the-art methods.

1. Introduction

Machine Learning (ML) is one of the most leveraged methods to realize autonomous operations in optical networks. Quality-of-Transmission (QoT) regression and classification are among the most frequent use cases when ML is applied to optical networks. For instance, QoT regression is widely employed to estimate the Bit Error Rate (BER), Optical Signal-to-Noise Ratio (OSNR), or Error Vector Magnitude (EVM) of an optical channel based on monitoring features [1, 2]. The same features are often used to obtain a QoT classifier, where an ML model verifies whether or not an optical channel configuration will meet given requirements [3]. The detection of physical layer attacks is another application, where ML models play a significant role [4]. The use of ML models for anomaly detection, fault prediction, and localization has also been reported in the literature.

As the use of ML broadens within the context of optical network automation, the challenges in building and maintaining the multitude of ML models increase. In this regard, two issues are of great importance. Firstly, current approaches require ML models to be manually built by engineers with knowledge (even if superficial) of *(i)* ML techniques and *(ii)* optical networks. Several steps, e.g., feature selection/scaling, are highly dataset-dependent, as demonstrated in [4], which translates into ML models being labor-intensive and requiring specialized professionals. Secondly, to take full advantage of ML, a multitude of models need to be built. In the best-case scenario, a single ML model can be general enough to be used over an entire network (e.g., QoT-related models), but in a common application, an individual model per optical channel is required. In practice, this may lead to tens, hundreds, if not thousands of ML models being used in a medium-scale network deployment. In summary, to fully realize the vision of autonomous network operations, the current approach to handling ML models leads to massive demands for highly-specialized practitioners and resources.

In response, this work introduces the concept of Machine-Learning-as-a-Service (MLaaS) in the context of optical network automation. MLaaS aims at providing a framework that enables *customers* (here, the optical network automation entities) to request and quickly get access to trained ML models with as little human intervention as possible. The MLaaS framework is responsible for automating the steps necessary to build, train and re-train ML models, drastically reducing or eliminating human intervention for the most frequent activities related to ML. We propose an architecture that enables an optical network automation framework to automatically trigger MLaaS functionalities in charge of automatic model creation and validation. We implement the envisioned architecture and demonstrate its suitability, analyzing the convergence time and accuracy of the resulting model in a QoT classification use case over the dataset reported in [3,5]. The results show convergence time and accuracy performance on par with the state-of-the-art Artificial Neural Network (ANN) [3] while requiring minimal human intervention. The latter translates into considerable savings in terms of engineering time.

2. Machine-Learning-as-a-Service for Optical Network Automation

MLaaS aims at automating the repetitive and time-consuming steps necessary to create, deploy, monitor, and update ML models. Fig. 1 illustrates the main steps involved in the MLaaS pipeline, divided into two stages. The pipeline is triggered upon the need for a new or updated ML model. The main inputs of the pipeline are the type of ML task required (e.g., classification, regression, prediction), the dataset to be used to train and evaluate the resulting model, the resource constraints to take into account while creating the model (e.g., maximum computing time, maximum number of threads), and the method to retrieve the final result (e.g., file or web service).

The *model creation* stage is associated with building (or updating existing) ML models. It requires the cooperation of several steps aimed at building the best model possible. Model definition selects, out of a portfolio of

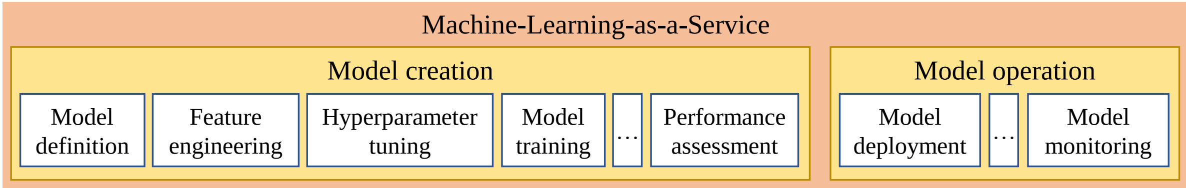

Fig. 1. MLaaS pipeline including ML lifecycle, and operational aspects of the ML model (MLOps).

potential models, the most suitable models for the task at hand (e.g., considering the dataset and the time/resource budget). Then, feature engineering, hyperparameter tuning, model training, and performance assessment iteratively improve the candidate model(s) until the time/resource limits are reached. Once the model is built, the *model operation* stage is responsible for model deployment (including means to perform inference in the model) and monitoring its performance. Fig. 2(a) shows the proposed architecture that integrates the MLaaS framework proposed in [6] with an optical network automation framework. Upon the need for a new or updated ML model, the optical network automation framework triggers the following workflow. First, the optical network automation framework sends an ML model request (i.e., via a YAML file) to the MLaaS framework. Fig. 2(b) illustrates the YAML content of the request, which specifies the task, the dataset, and the time constraint to be considered by the MLaaS framework. Then, the MLaaS framework obtains the dataset to be used for training, validation, and testing using the provider and identifier information as per the ML model request. In this context, the provider refers to the method through which the MLaaS framework should obtain the dataset (e.g., FTP, HTTP, OpenML). The identifier refers to the specific resource within the selected provider (e.g., file URL when using HTTP).

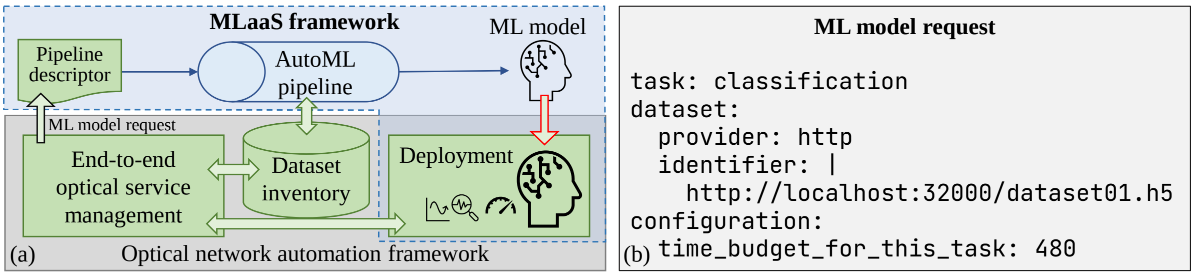

Fig. 2. (a) Architecture with MLaaS and optical network automation frameworks cooperating and (b) a hypothetical example of the ML model request.

Once the dataset is obtained, the model creation part starts. For building the ML model, the framework may leverage existing Automated ML (AutoML) libraries, e.g., [7, 8], or others that might have been developed ad hoc for a given use case. Finally, once the model is built, the model operation stage deploys the model within a compatible environment, making it ready to be used by the optical network automation framework. The response from the MLaaS framework (not illustrated in the figure due to space constraints) includes the location of the deployed model (similar to the dataset identifier in Fig. 2(b)).

3. Use Case: QoT Classifier

We implement the architecture proposed in Fig. 2(a), focusing on the model creation stage presented in Fig. 1. We leave the model operation stage for future work. The implemented framework uses the Auto-Sklearn [8] library for the model creation stage. The use case consists of building a QoT binary classifier, responsible for estimating whether or not a particular optical channel configuration would work if deployed, based on past established optical channel performance data. We use the dataset reported in [3, 5] for this assessment.

Auto-Sklearn leverages a portfolio comprised of Scikit-Learn supervised learning algorithms to build models. Based on the time/resource budget and the dataset at hand, Auto-Sklearn uses successive halving to make the training process by progressively pruning models that are not performing well. It also combines techniques of meta-learning to assist in self-adjusting the learning parameters so as to obtain the best possible ML model. In the end, the resulting Auto-Sklearn model contains a single or an ensemble of ML models that influence the final output based on their training performance.

We use *dataset 01* from [3] (Table 3), which contains 35 input features, 4 target features, and 1,524,755 samples. Before using the dataset in our experiment, we remove some unnecessary input features (i.e., the connection ID) and some unnecessary target features (i.e., we only use the “class” target feature). The target feature is a binary variable whose value represents whether or not the optical channel configuration given as input is satisfactory. We randomly select a balanced dataset with 100,000 samples out of the 1,524,755 samples of the dataset.

The performance of the training is analyzed by evaluating, over time, the accuracy over the training and validation datasets. The performance of the resulting model is assessed by comparing its accuracy to the accuracy of the ANN reported in [3]. To ensure a fair comparison, we build an ANN using the same balanced dataset used as input to our MLaaS implementation. For the ANN, we follow the specifications described in [3] (i.e., an ANN comprising 34 input neurons and one hidden layer with 256 neurons using tanh as the activation function, and pre-processing features by removing the mean and scaling to unit variance).

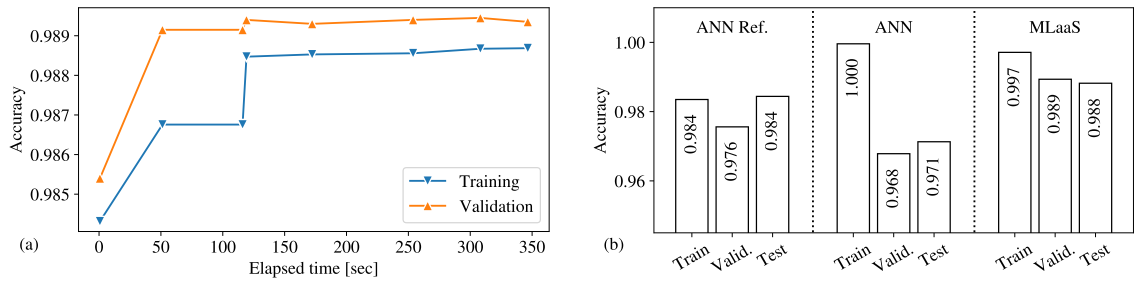

Fig. 3. Results: (a) training and validation accuracy over time; (b) accuracy on the training, validation, and test datasets reported in [3] and obtained by our ANN and MLaaS framework.

Fig. 3 shows the results of the training, validation, and testing of the model derived using MLaaS. Fig. 3(a) shows that the model built by the MLaaS framework takes approximately 150 seconds to converge. Comparatively, the ANN built for the same task takes more than 5 minutes to train over 400 epochs with mini-batches of 64 samples using the same CPU. The difference in training time comes from the fact that the AutoML library used here leverages simpler supervised learning algorithms in its portfolio, while ANNs adopt more complex operations during training. However, adopting GPUs for training and inference may substantially speed up the ANN.

Fig. 3(b) shows a comparison among the accuracy reported in [3], the one obtained by our ANN, and the one obtained by the MLaaS framework. The difference between the reference and our ANN comes from having been trained with different samples from the main dataset. On the other hand, our ANN and MLaaS-based models have been trained over the same subset of data. We observe that all the approaches lead to fairly accurate models. Our ANN does not reach the same validation and testing accuracy scores as the ones reported in [3], possibly due to being trained and evaluated under different subsets of data. However, the model built by our MLaaS implementation presents a slightly higher accuracy across all the sets of data. This means that the MLaaS framework was able to build a model that is as good at the task as a hand-crafted one while reducing human intervention significantly, and requiring no manual data pre-processing steps.

4. Conclusions

This work introduced the concept of MLaaS applied to optical network automation. We defined the main stages and steps involved in building and maintaining ML models that can be used to assist in the automation of several tasks across an optical network. The performance assessment results showed that a model generated by our implementation has comparable performance to a hand-crafted model reported in the literature, while drastically reducing the human intervention (i.e., time of a physical person) needed during its creation. As future work, we plan to expand the range of tasks provided by our MLaaS framework and improve its interface and performance.

Acknowledgment: This work was supported by Chalmers' ICT-AoA seed project "Auto5G" and by Sweden's innovation agency VINNOVA, within the framework of the EUREKA cluster CELTIC-NEXT project AI-NET-PROTECT (2020-03506).

References

1. L. Velasco *et al.*, "Learning life cycle to speed up autonomic optical transmission and networking adoption," J. Opt. Commun. Netw. **11**, 226–237 (2019). DOI: 10.1364/JOCN.11.000226.
2. Y. Fan *et al.*, "Feedforward neural network-based evm estimation: Impairment tolerance in coherent optical systems," IEEE J. Sel. Top. Quantum Electron. **28**, 1–10 (2022). DOI: 10.1109/JSTQE.2022.3177004.
3. G. Bergk *et al.*, "ML-assisted QoT estimation: a dataset collection and data visualization for dataset quality evaluation," J. Opt. Commun. Netw. **14**, 43–55 (2022). DOI: 10.1364/JOCN.442733.
4. C. Natalino *et al.*, "Experimental study of machine-learning-based detection and identification of physical-layer attacks in optical networks," J. Light. Technol. **37**, 4173–4182 (2019). DOI: 10.1109/JLT.2019.2923558.
5. G. Bergk *et al.*, "QoT dataset collection," `https://www.hhi.fraunhofer.de/networkdata` (2022).
6. C. Natalino *et al.*, "Machine-learning-as-a-service for network automation," to appear (2023).
7. H. Jin *et al.*, "Auto-Keras: An efficient neural architecture search system," in *Proc. of ACM SIGKDD,* (ACM, 2019), pp. 1946–1956. DOI: 10.1145/3292500.3330648.
8. M. Feurer *et al.*, "Auto-Sklearn 2.0: Hands-free AutoML via meta-learning," J. Mach. Learn. Res. **23**, 1–61 (2022).