CHALMERS
UNIVERSITY OF TECHNOLOGY

Analysis and Mitigation of Unwanted Biases in ML-based QoT Classification Tasks

Analysis and Mitigation of Unwanted Biases in ML-based QoT Classification Tasks

Carlos Natalino,[1,*] Behnam Shariati,[2] Pooyan Safari,[2] Johannes Karl Fischer,[2] and Paolo Monti[1]

[1] *Department of Electrical Engineering, Chalmers University of Technology, 41296 Gothenburg, Sweden*
[2] *Fraunhofer HHI, 10587 Berlin, Germany*

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

Abstract: We address the problem of mitigating biases in models used for the quality of transmission prediction. The proposed method reduces the relative accuracy difference between samples with different feature values by up to 45%.

1. Introduction

Estimating the quality-of-transmission (QoT) of unestablished lightpaths is one of the most important tasks to enable dynamic and efficient optical networking. Artificial intelligence/machine learning (AI/ML) models have been extensively studied for QoT estimation tasks, usually showing strong performance in terms of classification accuracy and estimation error [1, 2]. Such models, when correctly applied, enable substantial benefits for the planning and operation of optical networks [3].

In the case of QoT classification tasks, AI/ML models take as input the observed network state and resource usage (i.e., existing lightpaths), and output the predicted label that represents if a lightpath configuration will work (label *true*) or not (label *false*). The performance of these models is usually assessed by computing the accuracy, which represents the rate at which the model successfully predicts the correct label.To prevent unfair accuracy evaluation, datasets are split in a balanced manner, i.e., they contain an equal number of samples for the two labels. However, average accuracy across labels falls short in uncovering potential biases stemming from other characteristics that might be critical to network operators (e.g., modulation format, number of spans in the path).

Biases may lead to misclassifications that result in substantial spectrum efficiency degradation. For instance, false negatives (when the classifier predicts that a given lightpath that would work will not) may induce the unnecessary use of lower-order modulation formats. False positives (when the classifier predicts that a lightpath will work but it does not work in reality) will lead to unnecessary reconfigurations of the network due to trial-and-failure of lightpaths, incurring delays in the lightpath setup.

Fig. 1 illustrates the number of samples pertaining to each label, modulation format, and number of spans in the path from 20,000 randomly obtained samples from the dataset 01 in [4]. Fig. 1(a) confirms that we have a balanced dataset, i.e., we have the same number of samples for each label. However, Figs. 1 (b) and (c) show that the number of samples pertaining to each modulation format and number of spans, respectively, have large imbalance. Training and/or evaluating an AI/ML model using such a dataset can lead to large differences in accuracy of specific samples depending on their modulation format and/or number of spans. A naive solution would be to strive for a completely balanced dataset, i.e., over the labels and important features, but such a dataset would be extremely difficult to obtain under real-world conditions. A more appropriate approach is to acknowledge that datasets might be inherently imbalanced, both in terms of label and important features, and devise strategies to mitigate such biases. Bias mitigation is an active area of research in the AI/ML community [5].

In this work, we raise awareness of the unwanted biases that may be present in AI/ML models if not properly analyzed. Then, we propose a simple strategy to mitigate the biases by associating each sample with a weight that assesses the importance of the samples across multiple important features. Results show that the proposed strategy reduces the maximum accuracy difference among samples with different feature value by up to 45%, and the standard deviation by up to 36%.

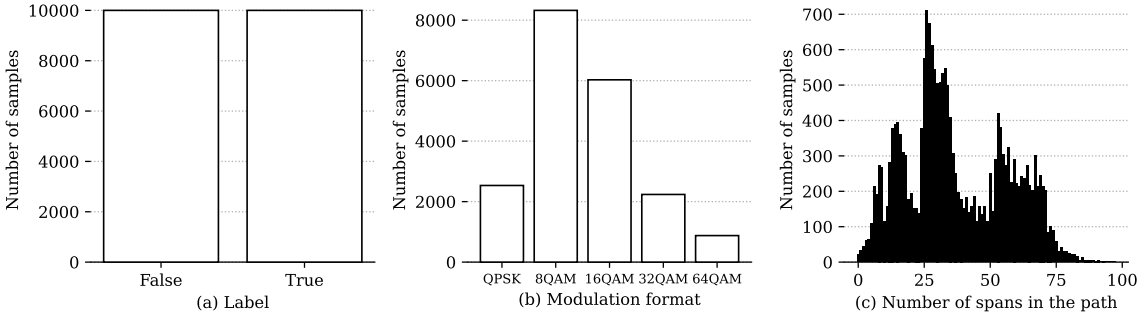

Fig. 1. Number of samples per (a) label, (b) modulation format, and (c) number of spans.

2. Mitigating Unwanted Biases with Feature-Based Sample Weights

Biases in AI/ML models are the result of a wide range of conditions. In most cases, biases result from training with biased data without proper signaling of such biases to the AI/ML training mechanism. The most well-known bias inherent in data is when the number of samples from different labels differs, and a mitigation is to define a weight to each label to counterbalance the difference.Other less-known biases may be the result of large imbalances in important features as opposed to labels, which is the focus of this work. In this case, important features are critical characteristics of the samples over which the AI/ML model would ideally achieve similar performance.

In the particular case of an artificial neural network (ANN) applied for binary classification, such as the ones adopted for QoT classification tasks, the updates of the ANN parameters are defined by the binary cross-entropy loss. Equation (1) shows how the binary cross-entropy is computed, where N represents the number of samples, and y_n and $\hat{y}_n$ are the true and predicted label of sample n, respectively. In (1) all samples have the same weight, not accounting for any (possible) imbalance or bias in the data. To mitigate bias in the data, we can adopt a variation of the binary cross-entropy loss shown in (2), where each sample n is associated with a weight w_n. When adopting (2), the challenge becomes how to compute w_n such that biases can be accounted for during training.

$$L = -\frac{1}{N}\sum_{n=1}^{N}[y_n \cdot \log \hat{y}_n + (1-y_n)\cdot \log(1-\hat{y}_n)] \tag{1}$$

$$L = -\frac{1}{N}\sum_{n=1}^{N} w_n \cdot [y_n \cdot \log \hat{y}_n + (1-y_n)\cdot \log(1-\hat{y}_n)] \tag{2}$$

In this work, we propose a simple approach that computes the sample weights based on the enumeration of important features from the dataset. Let us define X and Y as the set of samples and their associated labels, respectively, where $x_n \in X$ and $y_n \in Y$ represent the features and the associated label of sample n. The number of samples is defined as $N = |X|$. We define F as the set of features present in X, where x_n^f represents the value of features $f \in F$ of sample $x \in X$. The set of important features that should be accounted for is defined as $C \subset F$.

The intuition is to assign a weight for each sample that takes into consideration the occurrence of the feature values across all important features. We compute the weight w_n for a sample n as:

$$w_n = \frac{1}{N}\sum_{f\in C}\frac{|\bar{X}| : \bar{X} = \bigcup_{\bar{x}\in X} \bar{x}_n^f \neq x_n^f}{|C|}, n = 0,..,|X|. \tag{3}$$

In the formula, we define the set $\bar{X}$ as being the samples whose value in feature f differs from x_n^f. For example, in a dataset with 100 samples containing 20 samples for a given feature value, these samples will be given weight 0.8. When using more than one important feature, we average the weights across the features. Note that the introduction of the weights computed by (3) do not impact the sample importance if the dataset is also balanced with respect to the important features, i.e., all samples will receive the same weight in such case. To compute w_n for each sample we traverse the dataset $|X| \times |F|$ times, resulting in an overall complexity of $O = |X|^2 \times |F|$.

3. Results

We implement the proposed sample weighting strategy and test it over dataset 01 from [4, Table 3]. We randomly obtained a balanced dataset containing 100,000 samples, and divided it into three balanced datasets: training (50%), validation (30%), and testing (20%). As a baseline, we train an ANN following the same specification as in [4]: a single hidden layer with 256 neurons using *tanh* activation function, the output neuron using a sigmoid activation function, trained adopting the binary cross-entropy loss with RMSprop with a learning rate of 0.01. To assess the effectiveness of our approach, we trained an ANN with the same specification, but adopting the sample weights computed as described in the previous session. We selected two specific features due to their criticality for the operation of optical networks, and their relative importance for the output [6]: modulation format and number of spans. The distribution of these two features is shown in Fig. 1 (b) and (c), respectively. Both ANNs were trained for 200 epochs. The following results are shown over the testing dataset with 20,000 samples.

Fig. 2 shows the baseline results of the model trained without sample weights. As we can see in Fig. 2(a), the difference in performance between the labels ($\Delta_{min-max}$) is low, i.e., 1.1%, which at first does not raise any concerns with respect to bias. However, Figs. 2(b) and (c) show that the accuracy difference ($\Delta_{min-max}$) among modulation formats and number of spans can reach 10% and 53%, respectively. Moreover, the standard deviation of the accuracy (σ) across the two important features is relatively high. This evaluation clearly shows that the model shows drastically different accuracy depending on the modulation format and number of spans in the path.

Fig. 3 shows the accuracy results for the model trained with weighted samples. As we can see in Fig. 3(a), the difference of performance for the labels is slightly lower than the one obtained by the model without weights, indicating that the introduction of the weights did not degrade the balance of accuracy for the classes, and even improves it slightly, i.e., from 1.1% to 0.3%. However, when it comes to modulation format, Fig. 3(b) shows that

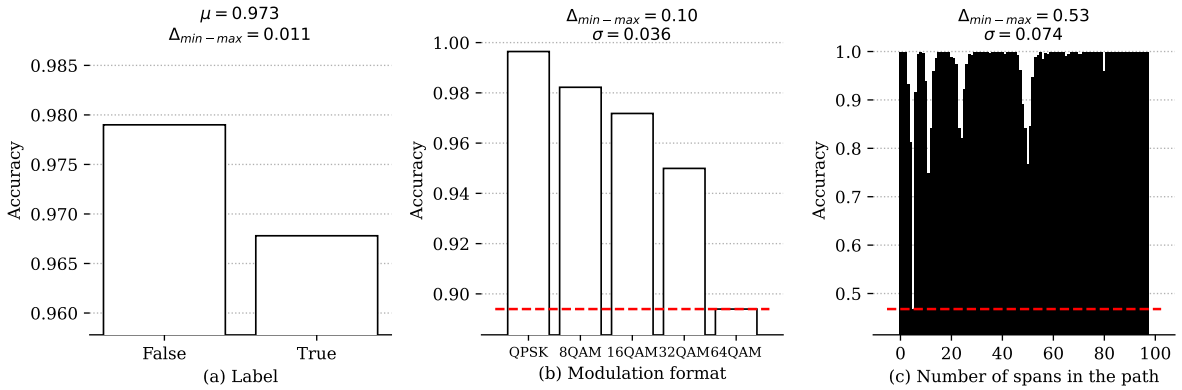

Fig. 2. Accuracy of the baseline ANN trained over a label-balanced dataset for (a) the label, (b) modulation format, and (c) number of spans. The red-dashed line shows the lowest accuracy.

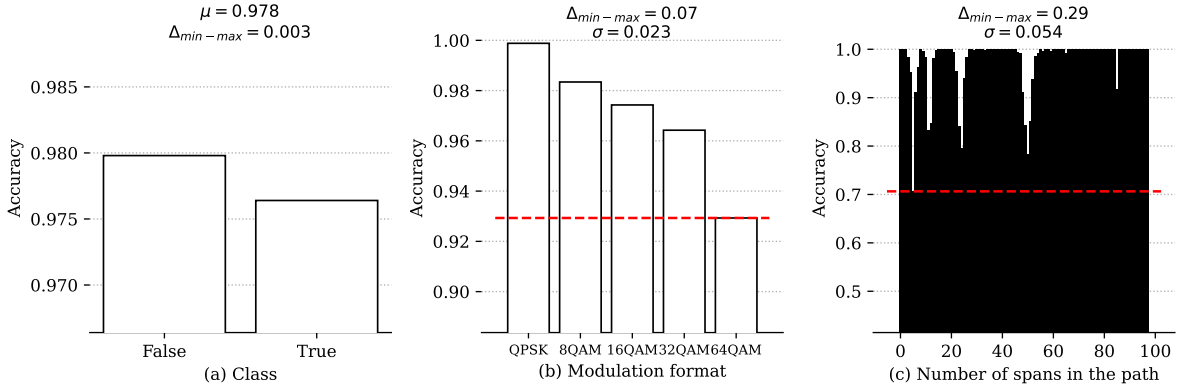

Fig. 3. Accuracy of the ANN trained with sample weights for (a) the label, (b) modulation format, and (c) number of spans. The red-dashed line shows the lowest accuracy.

the *min-max* difference is reduced by 30% (from 10% to 7%), and the standard deviation decreases by 36% (from 0.036 to 0.023). Fig. 3(c) shows that the performance difference across different number of spans also decreases from 53% to 29%, i.e., a 45% relative reduction. The standard deviation decreases by 27%, from 0.074 to 0.054.

4. Conclusions

This work brings attention to the potential unwanted biases present in AI/ML models used for QoT classification in optical networks. The proposed method reduces biases by up to 45% while reducing the standard deviation of the accuracy across different values of important features by up to 36%. Due to space constraints, we leave the analysis over other datasets from [4] for future work. Moreover, the current model is limited to features with discrete values and extensions for continuous feature values will be addressed in the future.

Acknowledgment: This work was supported by Sweden's innovation agency VINNOVA (2020-03506) and by BMBF (KIS8CEL010, FKZ 16KIS1282), within the framework of the EUREKA cluster CELTIC-NEXT project AI-NET-PROTECT.
Open source: The source code is available at: https://github.com/carlosnatalino/OFC24/.

References

1. C. Rottondi *et al.*, "Machine-learning method for quality of transmission prediction of unestablished lightpaths," J. Opt. Commun. Netw. **10**, A286–A297 (2018). DOI: 10.1364/JOCN.10.00A286.
2. S. Allogba *et al.*, "Machine-learning-based lightpath QoT estimation and forecasting," J. Light. Technol. **40**, 3115–3127 (2022). DOI: 10.1109/JLT.2022.3160379.
3. M. Lonardi *et al.*, "Machine learning for quality of transmission: a picture of the benefits fairness when planning WDM networks," J. Opt. Commun. Netw. **13**, 331–346 (2021). DOI: 10.1364/JOCN.433412.
4. 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.
5. N. Mehrabi *et al.*, "A survey on bias and fairness in machine learning," ACM Comput. Surv. **54** (2021). DOI: 10.1145/3457607.
6. O. Ayoub *et al.*, "Towards explainable artificial intelligence in optical networks: the use case of lightpath QoT estimation," J. Opt. Commun. Netw. **15**, A26–A38 (2023). DOI: 10.1364/JOCN.470812.