

Is Machine Learning Suitable for Solving RWA Problems in Optical Networks?

Ignacio Martín⁽¹⁾, José Alberto Hernández⁽¹⁾, Sebastian Troia⁽²⁾, Francesco Musumeci⁽²⁾, Guido Maier⁽²⁾, Oscar González de Dios⁽³⁾

⁽¹⁾ Universidad Carlos III de Madrid, Spain ignmarti@it.uc3m.es

⁽²⁾ Politecnico di Milano, Italy

⁽³⁾ Telefonica Global CTO, Spain

Abstract

We show that classical supervised Machine Learning techniques, after trained with a large number of optimal RWA configurations solved via ILP, can rapidly procure the most appropriate RWA configuration to be applied for a new traffic matrix.

Introduction

The availability of large amounts of data collected in modern optical networks, such as network alarms, traffic traces and optical signal quality parameters, has substantially increased the interest in applications of Machine Learning (ML) tools to solve typical optical networking problems in the recent years.

Using such advanced data analysis techniques may enable automation and self-adaptation to the high dynamicity of optical networks at various levels⁴, e.g., to estimate optical signal quality of transmission⁵, to perform failure detection⁶, virtual topology design⁷ or, in a more general networking context, for traffic flow classification⁸.

However, to the best of our knowledge, one of the most relevant problems addressed in the optical network field, the Routing and Wavelength Assignment (RWA) for multiple demands to be accommodated in an optical network, has not been yet addressed through any ML technique.

Historically, the RWA has been tackled by using optimal numerical methods, e.g., based on Integer Linear Programming (ILP)¹. However, although ILP-based approaches may provide optimized (e.g., cost-minimized) RWA solutions, they suffer from high computational complexity, which makes the RWA problem intractable for realistic optical network scenarios. For this reason, a number of lightweight heuristics have been defined, which require much lower computation effort, but provide in turn sub-optimal solutions.

The objective of this paper is to investigate on the applicability of ML methodological area onto the RWA problem. Specifically we refer to supervised ML techniques, i.e., in which a set of *labeled* training examples are used to build a ML-

based model in charge of performing optimized RWA with very low computational effort. We refer to a *labeled* training example to as a (x, y) pair, where x consists of a given network, featured by its topology, capacity, available wavelengths, and the set of traffic demands (i.e., a traffic matrix, TM) to be accommodated, and y is the corresponding RWA config solution, such as the one provided by an ILP solver.

We speculate that, by training the supervised ML module with a proper (i.e., sufficiently large) set of examples, we are able to generalize the RWA solutions also for unobserved sets of scenarios, so as to perform RWA with limited computational effort.

As a preliminary analysis, in this paper we design supervised ML algorithms, namely Logistic Regression with Lasso and Ridge regularization, trained over a simple Spanish network topology. The datasets for training, validation and testing of the different ML models have been generated using the well-known Net2Plan open-source Java-based tool².

Our results suggest that ML techniques provide a new framework to successfully address RWA problems, since they are capable of learning from previous observations and deciding on the most appropriate RWA config (*RWC*) to be used upon a new traffic matrix.

Modeling RWA as a supervised ML problem

Consider the 5-node network topology of Fig. 1 with 5 nodes and 7 links, and let us assume a number of W wavelengths per link each one operating at C Gb/s (i.e. W lambdas @ C Gb/s).

Let us also consider a 5×4 traffic matrix collecting the traffic demands d_{ij} (in Gb/s) from source i to destination node j ($i \neq j$). The RWA algorithm

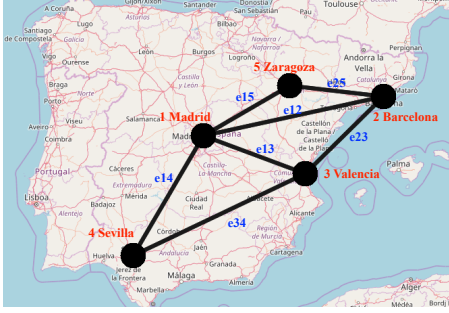


Fig. 1: Simple illustrative 5-node Spanish network

produces a list mapping each traffic demand d_{ij} with a list of sequences of links (routes) and wavelength allocations (lambdas), for instance:

$$\begin{aligned}
 d_{12} &\rightarrow (e_{12}, \lambda_1) \\
 d_{13} &\rightarrow (e_{13}, \lambda_1) \\
 &\vdots \\
 d_{24} &\rightarrow (e_{23}, e_{34}, \lambda_2) \\
 &\vdots \\
 d_{54} &\rightarrow (e_{52}, e_{23}, e_{34}, \lambda_3)
 \end{aligned}$$

This list was obtained after solving the ILP for the traffic matrix TM_1 (i.e. X_1 in the dataset); we shall call this set of optimal routes and wavelength assignments to as the target label $y_1 = RWC_1$. Very likely, TM_2 is expected to produce a different optimal RWC , namely $y_2 = RWC_2$, and so on.

If 10,000 different traffic matrices are used to feed the ILP, up to 10,000 different optimal RWC s will be produced by the ILP. However, in practical scenarios, a given RWC can be applied to multiple different traffic matrices with minimum added average link load and hop-count, generating slightly sub-optimal results. We shall allow a certain degree of extra load on attempts to reduce the number of RWC s used in the multi-class classification algorithm. This is explained in the next section.

In summary, the RWA problem has been transformed into a multi-class classification problem to be solved by ML techniques.

Experiments

Using the 5-node network of Fig. 1 as base topology, we have generated various datasets with 10,000 TM examples and their associated optimal RWC s. Each TM has been generated using the population-distance (aka gravity) model where the traffic demands between two cities depend on their size and distance³ plus additive Gaussian noise. For each pair, we draw a ran-

dom sample from a Gaussian distribution of mean population-distance value (centered at 100Gbps) and variance 0.15 times such mean, i.e. coefficient of variation of 0.15.

Each traffic matrix is then labeled by using the ILP RWA solver in Net2Plan. When solved, the amount of routing configurations in each dataset is reduced with a tolerance of 0.3 in terms of average hop count, so network configurations (in short, machine learning classes) are reduced.

Following this process, we generate TMs for 4 optical transponder configurations (i.e. lambdas and capacity), namely: 8 and 10 λ at 40 Gbps and 5 λ at both 100Gbps and 400Gbps.

Tab. 1 shows the experimental results for two versions of a well-known ML algorithm: Logistic Regression with LASSO (ℓ_1) and Ridge (ℓ_2) regularization. The table provides training, test and validation F-score values (trading-off both classification accuracy and recall), and other networking-related metrics of the resulting network setting, namely average and bottleneck Link Loads (LL), average and maximum number of hops per route and the training and validation percentage of feasible configurations.

In spite the low scores displayed in terms of ML metrics (F-score), most systems provide a moderately high amount feasible solutions regardless not being the original label. Indeed, different network configurations can satisfy the same traffic matrix requirements. Thus, feasibility, whether the ML output can be applied to the network, should be considered instead.

In this light, F_{train} and F_{val} values in the table indicate the percentage of feasible configurations in training and validation sets respectively. These amount for both variants in all datasets above 60%, which evinces a robust capacity for RWA resolution.

Whenever these classifiers procure a feasible configuration, the estimated RWC can be implemented in the network straightaway. Oppositely, if the given solution is not feasible, alternative solutions, such as the ILP, must be computed conforming a hybrid ML-ILP solution. Hence, we estimate the average completion time of this system by averaging ML and ILP classification completion times according to their occurrences (feasibility). In general terms, average completion time of a ML estimation is $10.1 \mu s$ for any model, whilst the time required by Net2Plan solver for the ILP is of $0.694 s$ on average.

Fig. 2 depicts the estimation of the completion

DATA Reg.	Train	Test	Val	avg LL	max LL	ave_hops	max_hops	F train	F val
d1: 5@400 (15 <i>RWCs</i>)									
ℓ_1	0.609	0.616	0.603	0.1286	0.51	2.05	4	99.73	99.7
ℓ_2	0.6101	0.6102	0.5907	0.1292	0.5098	2.05	4	99.61	99.9
d2: 5@100 (69 <i>RWCs</i>)									
ℓ_1	0.557	0.538	0.521	0.3554	1	1.75	3	86.31	84.6
ℓ_2	0.545	0.508	0.495	0.3509	1	1.745	3	85.52	84.16
d3: 10@40 (215 <i>RWCs</i>)									
ℓ_1	0.399	0.327	0.361	0.496	1	1.87	3	65.94	65.75
ℓ_2	0.395	0.318	0.295	0.4925	1	1.855	4	67.68	63.54
d4: 8@40 (197 <i>RWCs</i>)									
ℓ_1	0.429	0.357	0.385	0.534	1	1.64	3	65.28	64.02
ℓ_2	0.405	0.365	0.379	0.5319	1	1.64	3	64.88	64.25

Tab. 1: Traffic profiles for different functional splits

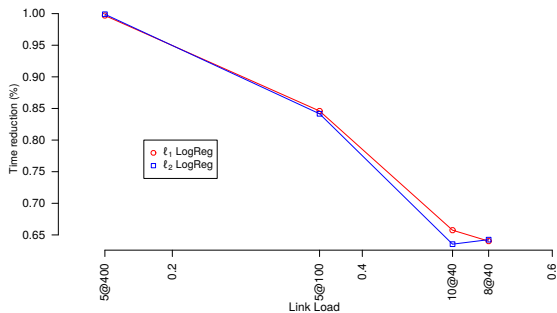


Fig. 2: Time gain of ML solution compared to link load

time cuts when using the ML-ILP system. The figure clearly illustrates how higher link load network settings have more complex configurations and, therefore, are harder to predict using ML trained with 10,000 samples. Indeed, the simplest dataset (d1) has the best performing models whereas the one with most *RWCs* (d3) shows the worst results.

In fact, such disparity of *RWCs* points to a key issue for RWA modeling: its enormous complexity. The more loaded and variable traffic in the network, the larger the space of *RWCs* and, therefore, the greater complexity and data needs to generate an accurate model.

Hence, while configurations for small static settings can be determined accurately using ML, larger dynamic networks settings increase enormously complexity and requires larger datasets.

Conclusions and future work

In summary, this work has studied how to apply supervised classification to a set of TM and their associated RWA config and whether its results may be applied to reduce completion time.

In the title, we introduced a question: *Is machine learning suitable for solving RWA prob-*

lems? and the answer is yes and no. Results do show that feasible solutions can be obtained even when misclassifying. Despite, such a complex classification requires enormous amounts of data and advanced models.

Indeed, future work shall investigate the limits of this approach by considering much larger datasets from different topologies and realistic TMs obtained from measurements.

Acknowledgements

The authors would like to acknowledge the support of the project TEXEO (TEC2016-80339-R), funded by Spanish MINECO and the EU-H2020 Metrohaul project (grant no. 761727).

References

- [1] H. Zhang *et al.*, "A review of Routing and Wavelength Assignment approaches for wavelength-routed optical WDM networks" in *Optical Networks Magazine*, Jan 2000
- [2] P. Pavon-Marino *et al.*, "Net2plan: an open source network planning tool for bridging the gap between academia and industry" in *IEEE Network*, 2015.
- [3] M. D. Vaughn *et al.*: "Metropolitan Network Traffic Demand Study" in *IEEE LEOS Conf*, 2000
- [4] J. Mata, *et al.*, "Artificial intelligence (AI) methods in optical networks: A comprehensive survey", in *OSN*, Apr 2018.
- [5] C. Rottondi, *et al.* "Machine learning method for quality of transmission prediction of unestablished lightpaths", in *IEEE/OSA JOCN*, Feb. 2018.
- [6] S. Shahkarami, *et al.* "Machine-Learning-Based Soft-Failure Detection and Identification in Optical Networks", in *OFC* 2018.
- [7] F. Morales, *et al.*, "Virtual network topology adaptability based on data analytics for traffic prediction", in *IEEE/OSA JOCN*, Jan. 2017.
- [8] T. Glennan *et al.* "Improved classification of known and unknown network traffic flows using semi-supervised machine learning", in *ACISP*, 2016