Link Failure prediction and Localization in Cloud Scale Networks using Supervised Learning

Zahra Bakhtiari

zahra22@stanford.edu

Abstract: We propose and demonstrate propose a proactive technique using supervised learning classification methods to pro-actively detect links failures. The method has the capability to localize flapping or failing links to enable cloud service providers to be proactive about service failures and customers outages.

I. Introduction

Cloud service providers attracts customers all over the world by providing diverse types of services via millions of servers. The key to CSPs' success is the ability to provide services to customers with fast response and all-time availability. The physical infrastructure which supports cloud services is based on IP and optical networks which can fail and go out of service for variety of reasons.

Proactive link health diagnosis is a game changer for cloud scale service providers. When a link failure happens, there are many risk scenarios including customer outage, long maintenance windows and complexity in terms of failure localization. In this paper we aim at enabling intelligent, pro-active optical network management and operation tool using supervised learning classification methods to pro-actively detect links failures.

By implementing this operational tool service providers can perform consequent remedial actions that can prevent network downtime and enable scheduled preventive maintenance.

II. Related Work

Cloud networks are complex systems, involving hardware, firmware, software defined network (SDN) control plane and advanced operational tools. Due to the complex nature of these systems any network change takes a long time including, upgrades or service deployments to take effect. Machine Learning (ML) is a perfect candidate for problems with complex nature [1]. Machine learning has many use cases in optical/IP networking including root-cause analysis [2], [3] and failure localization [4], [5], as well as CAPEX and OPEX related use cases. Machine learning research has a long history in optical/IP networking but due to recent innovations in intelligent computational hardware, Software-defined networking (SDN) and Network Function Virtualization (NFV) platforms machine learning has been deployed significantly by cloud service providers as well.

In this paper, we build linear classifiers based on supervised learning with proper pre-processed features that has the capability to be implemented at cloud scale.

III. Concept

Figure 1(a) is a conceptual diagram of Cloud Optical Infrastructure. As can be seen an end to end (E2E) link is

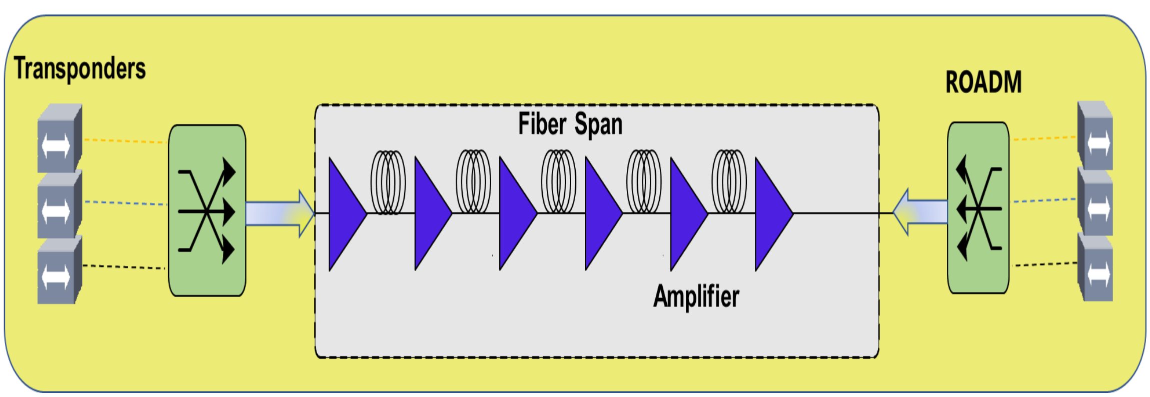

Fig. 1. Cloud Optical Infrastructure

composed of multiple spans. Each span includes fiber links, ROADM and amplifiers. Each fiber span has a baseline/target span loss. The diff between this target value and real time span loss is an important feature in link health diagnosis. Each amplifier is also associated to a baseline/target gain value. The diff between this target value and real time gain value is another important feature in link health diagnosis. As can be seen in Fig. 2.
the aggregated end-to-end span loss and amplifier gain diffs are used here to develop supervised learning models to proactively diagnose marginal or failing links from healthy ones and react to it in order to avoid customers outages.

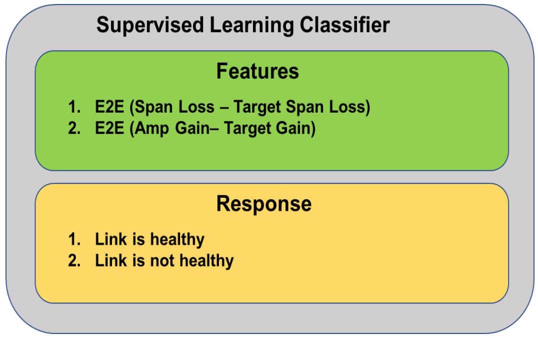

Fig. 2. Linear Classifier Schematic

III. DATASET AND FEATURES

Live network data has been used and retrieved from time-series data base to train this model which differentiates it from some few previous works. This step includes data wrangling, data labeling, preparing the features in the diff format for an end-to-end link. It also includes managing empty data cells as well as differentiating data per traffic direction. We filled in missing features raw values by linearly interpolating between neighbor cells. If the missing data was due to link failure root cause, we filled the cells according to the expected raw values when fiber cut happens.
This dataset includes 4 raw axis data each including 19 spans in the bin equal to 2 minutes. The metrics are the following: span Loss, target span loss, amplifier gain and target gain.
Eq.1 shows the feature engineering that were done to reduce the number of features to two and we use OSNR for data labeling.

$$x_1 = (Amp_{gain}) - (Target_{gain}),\ x_2 = (Target\_span\_loss) - (span\_loss) \tag{1}$$

Each aggregated feature is generated by choosing proper bin to find summation of the diffs between initial features as explained in Eq.2.

$$X1 = \sum_{n=0}^{18}(x_{1n}),\ \ X2 = \sum_{n=0}^{18}(x_{2n}) \tag{2}$$

Note that X1 and X2 are finalized features that are used for training. In total there are 6000 data points of two features each that were used for classifiers training and model performance evaluation. SVM with linear kernel, GDA and logistic regression classifiers have been trained with 75% of dataset and their performances were measured on the test dataset which includes 25% of initial dataset.

IV. Supervised Learning Models

Supervised learning is used in a variety of applications, including networking. The goal is to predict one or more target values given the input features. This section provides an overview of three linear classifiers that we used to model optical link failures. Those three classifiers are as following: logistic regression, GDA and SVM with linear kernel.

1. Logistic Regression

As a baseline model, we trained a standard logistic regression model with the average empirical loss in Eq.3.

$$J(\theta) = -\frac{1}{n}\sum_{i=1}^{n}\left(y^{(i)}\log(h_\theta(x^{(i)})) + (1-y^{(i)})\log(1-h_\theta(x^{(i)}))\right),$$

(3)

where $\left(y^{(i)}\right) \in \{0,1\}$, $h_\theta(\mathrm{x}) = \mathrm{g}(\theta^T\ \mathrm{x})$ and $g(z) = 1/(1+e^{-z})$. Note that logistic regression can perform well as a linear classifier for labeled data where the two classes are not fully separated.

For training step, 4500 data points of two aggregated features each were used. Note that, 1500 data points of two aggregated features each were used for performance evaluation. Note that we trained the logistic regression model using Newton's Method.

2. Gaussian Discriminant Analysis (GDA)

GDA were picked as a second classifier for link failure prediction with the joint distribution of (x, y) in Eq.4 as a model that relies on the gaussian nature of the data,

$$p(y) = \begin{cases} \phi & \text{if } y = 1 \\ 1-\phi & \text{if } y = 0 \end{cases}$$

$$p(x|y=0) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}}\exp\left(-\frac{1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0)\right)$$

$$p(x|y=1) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}}\exp\left(-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\right),$$

(4)

where φ, μ0, μ1, and Σ are the parameters of our model. Note that GDA can perform well as a linear classifier when the dataset has a normal distribution and even can beat logistic regression in case the two classes of data are fully separated.

3. SVM with Linear Kernel

SVM is a supervised machine learning algorithm which can be used as a linear or nonlinear classifier depending on the use case and data. The data base can be transformed to a higher dimensional space via a method called "kernel trick" by choosing a kernel in the following form:

$$K(x, y) = \phi(x)^T\phi(y)$$

(5)

We decided to train the model based on SVM as one of the classifier because in case the test accuracy is not good enough with linear decision boundary via logistic regression and GDA we can extend the method by picking a nonlinear kernel and see if nonlinear classifier performs any better.

V. Results and Discussions

As discussed previously, we tried training three different linear classifiers including logistic regression, GDA and SVM with linear kernel. For training step, 4500 data points of two aggregated features each were used. Note that, 1500 data points of two aggregated features each were used for performance evaluation. We evaluate the performance of these models by measuring test accuracy which is the count of correct classified test data over the size of test dataset. The test accuracy can be shown as following:

$$\text{Test } accuracy = \frac{1}{n}\sum_{i=1}^{n} 1\{y_i = y_{i'}\}$$

(6)

where y is the predicted label and y' is the true label. As can be seen in Fig. 3, we found that logistic regression and SVM with linear kernel perform with high accuracy of 99% due to the linear nature of the decision boundary. GDA performance comes second with the accuracy value of 97% due to the nature of the dataset. The cost value for logistic regression were reported as 0.02503. Note due to the linear nature of the decision boundary and high accuracy of linear classifiers there was no need to implement SVM with nonlinear kernel.

Classifier	Test Accuracy
Logistic Regression	**99%**
GDA	**97%**
Linear SVM	**99%**

Fig. 3. Test accuracy measures for logistic regression, GDA and linear SVM

To visualize the two logistic regression and GDA linear classifiers performances, we used different symbols for distinct labels. Figure 4 demonstrates the two classes separated by logistic regression classifier including a linear decision boundary defined by p(y|x) = 0.5. The linear decision boundary could classify the data successfully.

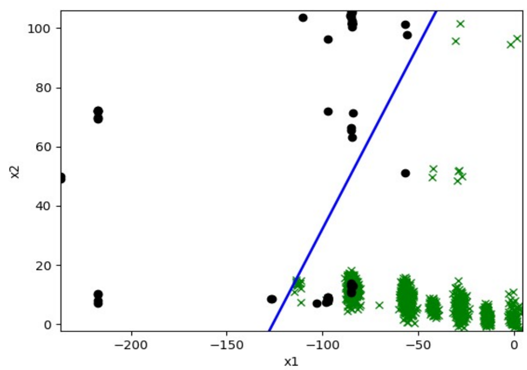

Fig. 4. Logistic regression classifier demonstration including the linear decision boundary

Figure 5 demonstrates the two classes separated by GDA classifier including a linear decision boundary defined by p(y|x) = 0.5. The linear decision boundary could classify the data successfully.

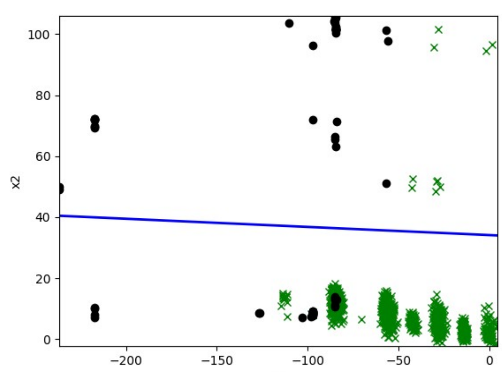

Fig. 5. Gaussian discriminant analysis classifier demonstration including the linear decision boundary

Note that model coefficients values for logistic regression and GDA are demonstrated in Table.1

Classifier	θ_0	θ_1	θ_2
GDA	34.37	-0.02.69661661	-1.01
Logistic Regression	22.01	0.174	-0.14

Table.1 Model coefficients for logistic regression and GDA

VI. CONCLUSION AND FUTURE WORK

We found that logistic regression and SVM with linear kernel perform with high accuracy due to the linear nature of the decision boundary and GDA performance comes second. Due to the linear nature of the decision boundary and high accuracy of linear classifiers there was no need to implement SVM with nonlinear kernel. Training the global network using these three classifiers will be the next step for this project. It will also be beneficial to add more intelligence to the model by gathering more data and adding more features and potentially testing a nonlinear classifier if the data makes it justified.

VII. References

[1] D. Rafique and L. Velasco, "Machine Learning for Optical Network Automation: Overview, Architecture and Applications," IEEE/OSA JOCN, 2018.
[2] D. Rafique et al., "TSDN-enabled network assurance: A cognitive fault detection architecture," in ECOC, 2017.
[3] D. Rafique et al., "Analytics driven fault discovery and diagnosis for cognitive root cause analysis," in OFC, 2018. [4] A. P. Vela et al., "Soft failure localization during commissioning testing and lightpath operation," IEEE/OSA JOCN 2018.
[5] B. Shariati et al., "Learning from the Optical Spectrum: Failure Detection and Identification [Invited]," IEEE/OSA JLT, 2018.
[6] Ll. Gifre et al., "Autonomic Disaggregated Multilayer Networking," IEEE/OSA JOCN, 2018.

APPENDICES

Here is the link to the zip file for this project:
https://github.com/mehrnaz22/ML_project/blob/master/ML_Project.zip