On Real-time and Self-taught Anomaly Detection in Optical Networks Using Hybrid Unsupervised/Supervised Learning

X. Chen[(1)], B. Li[(2)], M. Shamsabardeh[(1)], R. Proietti[(1)], Z. Zhu[(2)], S. J. B. Yoo[(1)]

[(1)] University of California, Davis, Davis, CA 95616, USA. Email: xlichen@ucdavis.edu, sbyoo@ucdavis.edu
[(2)] University of Science and Technology of China, Hefei, Anhui 230027, China. Email: zqzhu@ieee.org

Abstract *This paper proposes a real-time and self-taught anomaly detection scheme for optical networks using hybrid unsupervised/supervised learning. Evaluations with an experimental dataset demonstrate that the proposed scheme can successfully identify* 100% *of the anomalies without any prior knowledge of abnormal network behaviors while restricting the false positive rate to be only* 6.5%.

Introduction

While hard failures (e.g., fiber cuts) can cause immediate service disruptions, anomalies in optical networks may result from equipment malfunctions, management software faults and malicious attacks etc., which gradually degrade the network operations[1]. Therefore, powerful anomaly detection and localization schemes are essential for enhancing the availability of optical networks. Previous solutions mostly adopted threshold-based schemes and could not detect anomalies unless they had induced significant deviations of network parameters. On the other hand, thanks to the rapid advances in machine learning (ML) technology, recent studies have reported a few cognitive anomaly detection algorithms based on the learning of network behaviors with ML[1,2]. Nevertheless, these algorithms were trained with manually featured abnormal network behaviors and were unable to detect unknown anomalies. Note that, anomalies usually occur infrequently (therefore difficult to collect) but exhibit unique patterns compared with normal network behaviors[3]. An unsupervised clustering algorithm that directly learns patterns of data by exploiting the similarities among data instances would become a promising solution to distinguish anomalies from normal behaviors.

In this paper, we propose to realize real-time and self-taught anomaly detection in optical networks using a hybrid unsupervised/supervised learning scheme. The proposed scheme first employs an unsupervised self-learning data clustering module (DCM) to extract the patterns of the performance monitoring data. Then, to facilitate real-time anomaly detection, we develop a self-taught mechanism that trains a supervised learning deep neural network (DNN) based classifier & regressor with the learned knowledge by the DCM. Evaluations with an experimental dataset show that our proposal can identify 100% of the anomalies without any prior knowledge of their patterns while the false positive rate is only 6.5%.

Proposed Framework

Fig. 1 shows the proposed framework for real-time and self-taught anomaly detection in optical networks. The network manager deploys optical spectrum analyzers (OSAs) at certain locations in the data plane to monitor the signal power of each channel as well as the noise level. More advanced optical performance monitoring techniques, e.g., coherent detection, can be used to measure the quality-of-transmission of signals. By leveraging the software-defined networking (SDN) scheme, the network manager is able to collect the monitoring data remotely and in real time. The monitoring data are stored in a database and then sent to the data preprocessing module (DPM) for feature engineering. Specifically, the DPM may transform, clip or combine the raw data to generate new data instances suitable for different anomaly detection purposes, e.g., single-point or end-to-end lightpath inspection.

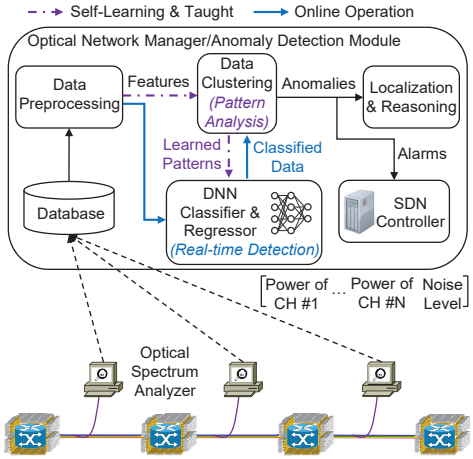

Fig. 1: Proposed anomaly detection framework.

In the learning phase, the features extracted by the DPM are fed to the unsupervised *self-learning* data clustering module (DCM) for pattern analysis. Basically, the DCM exploits the similarities among data instances and divides the monitoring dataset into multiple clusters and outliers, i.e., extracting the inherent rules lying in the big data. Here, outliers refer to isolated or sparse data instances that cannot form clusters. The DCM alarms the outliers as anomalies based on a consensual assumption that network anomalies occur rarely compared with normal behaviors. Note that, executing the DCM every time a new data instance arrives is time-consuming as the DCM has to revisit all the instances already in the database. To facilitate real-time anomaly detection, we train a supervised learning DNN classifier & regressor with the learned patterns by the

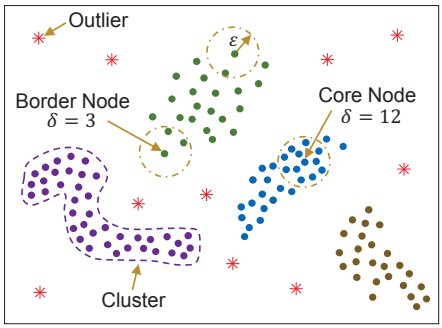

Fig. 2: An example for the principle of DBSCAN.

DCM (i.e., a *self-taught* mechanism). Then, during *online network operations*, the DNN is used for predicting whether new monitoring data are abnormal and classifying normal instances into existing clusters. Classified data instances are returned to the DCM for results verification and for periodical knowledge refreshing. We can see that the DCM does not require any prior knowledge of abnormal network behaviors as input, and therefore, can potentially detect arbitrary and unknown types of anomalies. Finally, upon detecting an abnormal instance, the DNN/DCM raises an alarm to the SDN controller and also sent the instance for further anomaly localization and reasoning. The alarms enable the SDN controller proactively adjusting the service provisioning schemes to mitigate the risk of severe future service disruptions.

Algorithm Design

We designed the DCM with a density-based clustering algorithm dubbed DBSCAN[4]. Let S denote the monitoring dataset and $d_{i,j}$ represent the distance (e.g., Euclidean distance) between data instances s_i and s_j ($s_i, s_j \in S$). DBSCAN first defines the ε-neighborhood of each s_i as $\{s_j \in S | d_{i,j} \leq \varepsilon\}$, whose size is known as the density of s_i (denoted as δ_i). A data instance with $\delta_i \geq MinPts$ is marked as a core node. Then, DBSCAN defines s_j as a density-reachable node from a core node s_i if $d_{i,j} \leq \varepsilon$ or there exists a sequence of core nodes $s_p, ..., s_q$ such that $d_{i,p} \leq \varepsilon, ..., d_{q,j} \leq \varepsilon$. DBSCAN constructs each cluster $S_c \subset S$ by starting from a random unvisited core node and iteratively including density-reachable nodes from it. The acquired clusters may afterward be merged if the distances among them are small. Finally, DBSCAN categorizes data instances that do not belong to any cluster as outliers. Fig. 2 shows an illustrative example for DBSCAN. We can see that DBSCAN can detect arbitrary shapes of clusters, making it a promising scheme for exploring the patterns of normal and abnormal behaviors in optical networks.

The example in Fig. 2 also indicates that the border nodes of clusters normally have much lower densities than the core nodes, while the outliers are with the lowest densities. Inspired by this observation, we designed a DNN classifier & regressor (Fig. 3) to assist fast and real-time anomaly detection. Specifically, based on the output of the DCM, we obtain the training and testing datasets for the DNN by assigning each s_i a label consisting of the corresponding cluster ID and density value. The DNN is trained with the backpropagation algorithm which aims to minimize the overall difference between the outputs of the DNN and the labels of the training data. We used the testing dataset to verify the prediction accuracy of the DNN. During online network operations, each newly collected data instance $\hat{s}_i$ is first evaluated by the DNN instead of going to the DCM directly. We claim $\hat{s}_i$ with high predicted density (i.e., $\hat{\delta}_i > \delta_0$) to be normal and expand the database according to the cluster $\hat{s}_i$ has been classified into. Otherwise, we mark $\hat{s}_i$ as a suspected point and invoke the DCM for verification. This way, we significantly reduce the number of executions of the time-consuming DCM. Note that, both the DCM and DNN classifier & regressor will be periodically refined in the background with the state-of-art database for refreshing the attained knowledge repository.

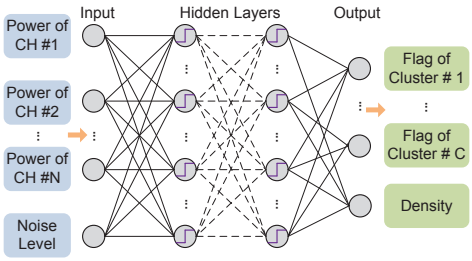

Fig. 3: Structure of the DNN classifier & regressor for single-point anomaly detection.

The proposed scheme also allows the localization and reasoning of the detected anomalies. Basically, we can calculate the distances in each dimension between an abnormal data instance s_i and those normal instances belonging to the neighboring clusters of s_i. By analyzing the distance vectors, we can discern from s_i either (1) notable deviations in certain dimensions (e.g., a sudden increase of the noise level at an intermediate node of a lightpath due to the amplifier failure) or (2) seemingly regular fluctuation of each dimension but abnormal overall patterns (e.g., a gradual decrease of the channel power gain at the early stage of the amplifier malfunctioning).

Results

We evaluated the performance of the proposed self-taught anomaly detection scheme with the performance monitoring data collected from a seven-node testbed[5]. We modified network parameters and configurations to emulate various network anomalies such as EDFA malfunctioning and channel misconfigurations. We focus on the single-point anomaly detection and preprocess the experimental dataset to contain $8,249$ instances, each of which has 22 dimensions (power measurements from 21 channels and the noise level). All the data instances are normalized before evaluations.

Table 1 summarizes the results of the false negative (f_n) and false positive (f_p) rates from the DCM with different setup of ε and $MinPts$. We can observe a clear

Tab. 1: Results of the false negative (f_n) and false positive (f_p) rates from the DCM ((f_n, f_p)%).

$MinPts$ \ ε	1	2	3
3	22.0, 4.6	54.0, 0.5	92.0, 0.0
5	0.0, 8.8	24.0, 0.6	76.0, 0.0
8	0.0, 11.2	0.0, 0.9	50.0, 0.0
10	0.0, 11.8	0.0, 1.5	14.0, 0.0
12	0.0, 12.4	0.0, 2.1	14.0, 0.0
15	0.0, 13.2	0.0, 3.9	14.0, 0.0

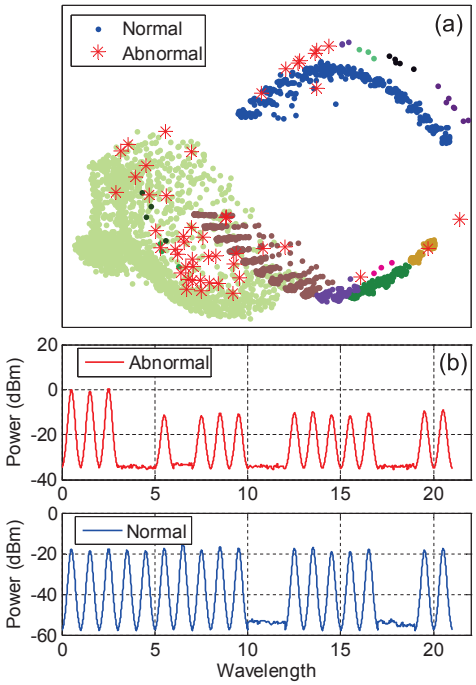

Fig. 4: (a) Clustering results from the DCM ($\varepsilon = 2$, $MinPts = 8$), (b) a comparison between abnormal and normal instances.

trend that f_n decreases with $MinPts$ but increases with ε, while f_p behaves oppositely. This is because a larger value of $MinPts$ means a higher minimum node density required to form clusters, and therefore facilitates the successful detection of low-density outliers/anomalies (lower f_n). However, increasing $MinPts$ will also increase the probability that normal data instances are incorrectly identified as anomalies (higher f_p). On the other hand, according to the principle of the DBSCAN algorithm, node densities increase with ε. Hence, increasing ε encourages the forming of clusters and hinders the capability of differentiating normal and abnormal behaviors. With $\varepsilon = 2$ and $MinPts = 8$, the DCM is able to detect all the anomalies while achieving a false positive rate of only 0.9%. We mapped the original 22-dimension dataset into a two-dimension dataset using the principal components analysis (PCA) technique and visualized the clustering results from the DCM in Fig. 4(a) (normal data instances are marked with dots and we used different colors to distinguish different clusters). Overall, the DCM obtains 12 normal clusters and 50 outliers. Note that, due to the transformation with the PCA, some outliers in Fig. 4(a) seems to lie inside clusters which are in fact not the real cases. Fig. 4(b) shows a comparison between abnormal and normal instances, where we can intuitively see two different patterns (the abnormal instance has a higher noise level and three high-powered channels).

Tab. 2: Results of the false negative and false positive rates from the DNN classifier&regressor (%).

$\bar{\delta}_0$	-0.80	-0.82	-0.85	-0.86	-0.87	-0.88
f_n	0.0	0.0	1.9	3.9	13.5	50.0
f_p	20.6	6.5	4.1	2.0	0.4	0.1

Next, we evaluated the performance of the DNN classifier & regressor. We implemented a DNN with three hidden layers ($[128, 128, 128]$). Based on the clustering results in Fig. 4(a), the output layer of the DNN contains 14 neurons (1 as the anomaly indicator, another 1 for outputting the predicted node density and the rest 12 for indicating the classification result). We divide the original dataset into the training and testing sets with a ratio of $4:1$. Table 2 shows the results of f_n and f_p from the DNN classifier & regressor with different values of $\bar{\delta}_0$. Here, we use $\bar{\delta}_0$ instead of δ_0 since we have normalized the node densities to enable the DNN to perform classification and regression tasks simultaneously. The results indicate that the DNN precisely predicts low densities for the abnormal instances and f_n decreases with $\bar{\delta}_0$ (recall that a data instance s_i is detected as abnormal if $\delta_i \leq \delta_0$). As expected, f_p increases with $\bar{\delta}_0$. When compared with the DCM, we can see that the DNN classifier & regressor significantly reduces the time-complexity while sacrificing only slightly the detection accuracy, i.e., still being able to detect all the anomalies but with a higher f_p as 6.5%.

Conclusions

In this paper, we presented a hybrid unsupervised/-supervised learning scheme to achieve real-time and self-taught anomaly detection in optical networks. Evaluations with an experimental dataset show that the proposed scheme can identify 100% of the anomalies without any prior knowledge of abnormal network behaviors while the false positive rate was only 6.5%.

Acknowledgments

This work was supported in part by DOE DE-SC0016700, and NSF NeTS 1302719. The authors would also like to thank G. Liu and K. Zhang at UC Davis for providing the experimental dataset.

References

[1] A. Vela et al., "BER degradation detection and failure identification in elastic optical networks," J. Lightwave Technol., Vol. **35**, no. 21, p. 4595 (2017).

[2] D. Rafique et al., "Cognitive Assurance Architecture for Optical Network Fault Management," J. Lightwave Technol., Vol. **36**, no. 7, p. 1443 (2018).

[3] D. Rafique et al., "Anomaly Detection for Discrete Sequences: A Survey," IEEE Trans. Knowl. Data Engi., Vol. **24**, no. 5, p. 823 (2012).

[4] M. Ester et al., "A density-based algorithm for discovering clusters in large spatial databases with noise," Proc. KDD, p. 226, Portland (1996).

[5] G. Liu et al., "The first testbed demonstration of cognitive end-to-end optical service provisioning with hierarchical learning across multiple autonomous systems," Proc. OFC, Th4D.7, San Diego (2018).