Detecting Outliers in Network Transfers with Feature Extraction

Mariam Kiran
Lawrence Berkeley National Laboratory
Berkeley, CA, USA
mkiran@lbl.gov

Cong Wang
Renaissance Computing Institute
Chapel Hill, North Carolina, USA
cwang@renci.org

Nageswara S. V. Rao
Oak Ridge National Laboratory
Oak Ridge, Tennessee, USA
raons@ornl.gov

Anirban Mandal
Renaissance Computing Institute
Chapel Hill, North Carolina, USA
anirban@renci.org

ABSTRACT

Reliable file transfers are essential for successful science computations and experiments that need large complex data files to be moved across long distances. Networks use protocols such as TCP and UDP to support file transfers, and their performance degrades under packet losses and duplications, thereby adversely affecting science applications. Hence, scientists and engineers often monitor network statistics to find and repair network problems that cause such degradations.

Outliers or anomalies can be detected using statistical and machine learning methods that highlight *abnormal* behaviors. However, without knowing what is *normal*, it is very difficult and subjective to determine the anomalies. In this paper, we investigate statistical and machine learning approaches to extract unknown or unsupervised features from TCP and perfSONAR data sets, with the aim to identify anomalies such as packet loss, duplication and retransmission sequencing that affect file transfer performance. Simple statistical and feature extraction methods (e.g. PCA) have shown that detecting anomalies is simpler if labeled data sets of normal behavior are available. While perfSONAR logs record losses, they do not detect duplications and reordering. We show that PCA applied to TCP statistics is able to determine abnormal behavior in the above cases. In particular, our results show that simple feature extraction techniques provide building blocks for more advanced network monitoring tools that can quickly flag anomalies using simple measurements, thereby helping to build confidence in file transfer reliability.

KEYWORDS

anomaly detection, unsupervised feature extraction, statistical analysis

1 INTRODUCTION

Scientific data is growing and placing unprecedented network demands, with complex applications requiring efficient and reliable file transfers for successful computations and experiments. The communication networks tend to make the physical distance irrelevant for science, but only at an added cost of ensuring performance and security, which are both affected by anomalies such as high packet losses and duplications. Various domains employ anomaly detection mechanisms for intrusion or fraud detection, system health monitoring, or detection of outliers that correspond to 'not normal' behavior. Recent advances in machine learning are exploring deep learning and statistical methods to extract behavior outliers in all fields including network research. In particular, network researchers have been using machine learning methods for finding outliers in traffic statistics, hardware/software bugs for improving operations and security since early publications from 1990s [1].

For example, anomaly detection often starts with an understanding what is *normal* behavior through classification or clustering methods. Applications in traffic profiling and intrusion detection, have used machine learning approaches, like Naive Bayes Theorem, support vector machines or random forests, as well as statistical methods like rule-based approaches [2, 12]. Typically, these approaches ingest large amounts of data about packet numbers, flow direction, IP addresses, QoS values, latency or file traces (to name a few) to characterize the traffic behavior.

Apart from anomaly detection, file transfer performance can be improved by using VLANs to set up secure links combined with the reliable packet delivery of TCP protocols. Scientific computations and experiments typically use TCP to deliver scientific data without loss, duplications or alterations of packets, to ensure confidence in their results.

Monitoring tools such as TSTAT [10] can be used to collect statistics of TCP transfers between servers and clients. These statistics include the number of packets sent, received, round trip times and TCP window sizes, which are recorded during TCP transfers. They can be analyzed using statistical and machine learning techniques to infer if packets have been delivered in time or lost during transaction (Figure 1).

Machine learning and statistical analysis of network data typically relies on the prior knowledge of predefined thresholds or normal anticipated behaviors of network traffic. In this paper, we take a step back from this, and investigate the question: *if no knowledge about normal behavior is available, can we still find anomalous behavior in traffic traces?*

1.1 Motivation

Most network research is based on either formula-based methods to understand why certain behavior is observed, or history-based

DISE Workshop@ICML, July, 2018, Stockholm, Sweden
2018. Published at ICML Workshop for Deep Learning for Safety-Critical in Engineering Systems 978-x-xxxx-xxxx-x/YY/MM

[1] https://www.ietf.org/

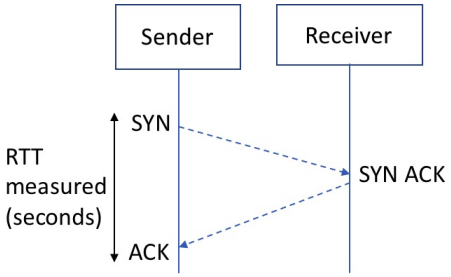


Figure 1: Round Trip Time measured in the three way communication in TCP.

methods to predict future behavior [12]. Understanding what is normal versus abnormal behavior is key to extracting anomalies, but there is no universal definition of normal. Formula-based approaches have worked in cases, but assume certain environmental conditions in which formulas hold true. For example, investigating congestion often refers to TCP window sizes, but these sizes could be affected by other issues on network links such as physical losses. Additionally, packet loss is difficult to capture, if there is a transfer of a large number of small files which result in short bandwidth bursts on the link. Mellia et al. [11] presented a summary of anomaly types that can be detected in file transfers based on data collected at the source. However, their classification methods assume multiple thresholds and a priori known formulas, for anomalies to be detected. In this paper, we focus on extracting features from TCP and perfSONAR statistics where no prior knowledge of behavior is available. Using unsupervised feature extraction (e.g. Principal Component Analysis (PCA)) and statistical methods, we extract patterns that could serve as *features*, which could be building blocks for future network monitoring and detection tools.

Specific contributions of this paper are as follows:

- Statistical analysis methods to extract behavior patterns of TCP RTT estimates in known normal and abnormal traffic traces.
- Unsupervised feature extraction methods (e.g. PCA) to extract features (or patterns) of behavior of RTT estimates in known normal and abnormal traffic traces.
- Experiments on a real network testbed and evaluation of our approaches with data collected from eight experiments using TSTAT [10] and perfSONAR logs [6].
- Feature extraction methodology that can be used to extract features for supervised learning approaches applicable in future detection tools.

This work is provides foundational steps for designing unsupervised feature extraction tools for network research. The rest of this paper has been organized as follows: Section 2 describes related work in the area of traffic anomaly detection and tools used. Section 3 describes the proposed methods for building a feature data set from limited data samples collected from real network runs. This is discussed as experimental setup in Section 4. The results of our analysis and the features are documented in Section 5 with statistical and feature extraction results. Finally, Section 6 discusses the approaches with future work and conclusions presented in Section 7.

2 RELATED WORK

Network engineers deploy various kinds of measurement tools (e.g. perfSONAR, netflow, sflow, to name a few), to measure various properties of network operations like usage traffic, router health, link performance, etc. For example, perfSONAR records loss, jitter and utilization, whereas netflow records packet data with IP addresses, bytes sent and the hops taken. In this paper, we particularly focus on TCP statistics and perfSONAR logs to determine features for reliable file transfers.

Anomalous traffic behaviors have been classified into three main groups [4, 11] - Packet loss, packet duplication and number of retransmissions triggered during a transfer. TCP statistics or TSTAT files contain header details such as packets sent/received, end-to-end RTT, packets observed in a TCP window and 150 more variables [10]. Researchers have used these statistics to compile a list of anomalies in passive traffic analysis [13], or predict future traffic patterns based on historical data [12]. Luca et al. [13] proposed a heuristic classification approach to identify possible anomalies by quantifying the sensitivity of certain parameters on flow measurements. Although based on established literature data, this technique showed that RTT had a high dependence on current traffic load that affected prediction accuracy.

But et al. [3] showed that just RTT estimation is not enough for studying network issues. Applying the Jacobsons algorithm, the authors showed the affect of minimizing packet transfer times on TCP stream jitters, which showed additional factors affect network behaviors.

Traffic anomalies can be a result of bad user or device behavior and can lead to network disruption. Jiang et al. [8] used network topology and destination data to include as part of feature sets detected for anomalous behaviors. By doing so, this method was able to identify ill behaving devices and users which could be rectified for improving operations. Similarly [9] proposed using entropy methods to extract multiple traffic features about flows. While entropy was successful in detecting low-volume anomalies, this method was unable to detect anomalies in high-volume data. In all of these above approaches, the feature extraction method was dependent on the current traffic nature, which can vary dynamically in a large scale distributed network system. As such, these features can not serve as a general feature database that can be used for abnormal file transfers.

Feature extraction can also be used to investigate user behavior with traffic models [14]. Zhang et al. [16] described anomaly detection as a significant step for network intrusion detection systems (NIDS) to work, but highlight that most of these systems are based on known or supervised data sets. The authors recognize that a more robust unsupervised feature extraction method is needed, with features learned from real network data sets, to make the NIDS systems more reliable.

Simple unsupervised feature extraction, based on domain knowledge, can extract certain characteristics from known transfer datasets. This can allow researchers to build relationships among sensitive parameters such as congestion and availability with transfer file

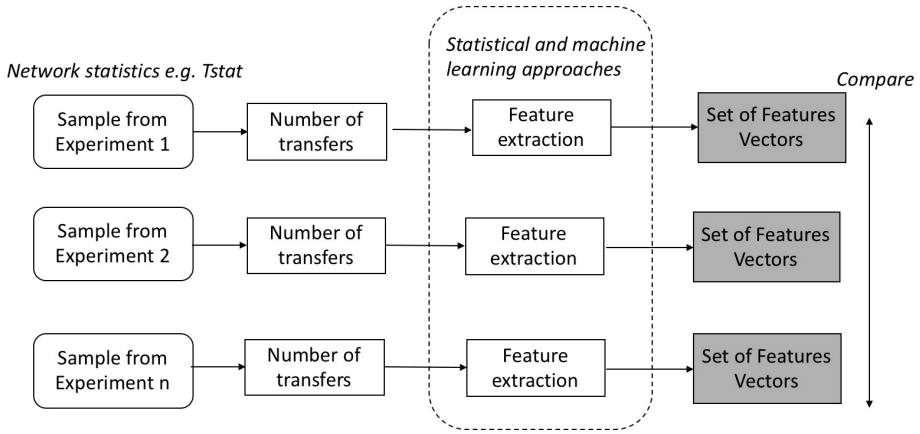


Figure 2: Proposed methodology for extracting features and comparing samples.

type. Simple unsupervised techniques such as Principal component analysis (PCA) and clustering can help build feature filters that can be leveraged for any future machine learning methods. With the goal for understanding packet loss, congestion and the impact on end-to-end performance, this paper will investigate TCP statistics and perfSONAR logs [6], to build feature sets of normal traffic data. In our experiments, we use real network data and ignore specific file details such as file size, workflow stage and link details, to allow core network properties to be extracted as general normal features.

3 PROPOSED METHOD

Figure 2 shows the overall approach for extracting and comparing features across sample traces collected during network transfers. Monitoring tools run at specific times collecting sample data at different times of the day. Each sample contains between 10-100 transfer details, which are independent tests on the link health and file transfer details. Example of two such samples is shown in Table 1, 2 from perfSONAR and TSTAT data.

3.1 Extracting Features

A feature in data sets is a property of the sample, which can be one of the variables such as file size, window size or more. In order to find common features across samples collected from 'normal' and abnormal file traces, we can extract statistical relationships among the various variables (or columns) in a sample set. For example, in normal transfers with no loss, the number of packets entering is always equal to number of packets leaving the link. And any deviation from the rule, is an unidentified feature which stands out as an anomaly. With this we define features as follows:

Definitions: *Features* are properties of one transfer recorded during the monitoring runs. These can be the individual variables or relationships among the variables. A *sample* is a collection of transfers, where each transfer has multiple features.

In order to extract features, we explored using statistical and machine learning algorithms to summarize relationships among data variables, and also catalog common characteristics among sample transfers. Section 4 describes how the experiment was setup to record transfer samples in TSTAT and perfSONAR data. A total of eight experiments were conducted with separate transfer scenarios, normal versus abnormal traffic. Abnormal traffic was recorded by artificially inducing anomalies in the transfers. More details are given in next section.

Table 1: Example perfSONAR data set

Sent	Received	Lost	Duplicated	Reordered
100	99	1	0	0
100	100	0	0	0
100	100	0	0	0

3.2 Statistical feature extraction

Samples for each experiment were analyzed for features or relationships among the variables. Simple statistical techniques such as mean, median and standard deviations were used to find overall characteristics in the transfers.

3.3 Unsupervised feature extraction: PCA

Principal component analysis (PCA) is an unsupervised feature extraction algorithm used in machine learning approaches. It can summarize data properties and characteristics by creating correlation among variables. PCA largely depends on linear combinations and constructs new features that summarize the variation among the data variables. The method outputs eigenvectors and eigenvalues which can be used to highlight the most influential variables and also help reduce dimension of data sets for faster processing. In this paper, PCA extracts new feature sets that contain correlation among the data variables (shown in Figure 3).

4 EXPERIMENTAL SETUP

In this section, we describe a set of experiments conducted in a controlled network testbed called ExoGENI [1], to evaluate the effectiveness of our anomaly detection mechanisms. ExoGENI is a NSF funded IaaS cloud testbed that orchestrates a federation of independent cloud sites located across the US and circuit providers. As shown in Figure 4, our experiment topology consists of two VMs

Table 2: Example TSTAT data set

Client/Server IP addr	Client/Server TCP port	RST sent	ACK sent	...	Average RTT C2S	...	Average RTT S2C	...	win	... more
172.16.0.1	45540	0	16	...	10.417495	...	7.39166	...	21545	...
100	100	0	0	...	10.405245	...	7.204025	...	53638	...
100	100	0	0	...	10.357496	...	7.202116	...	52238	...

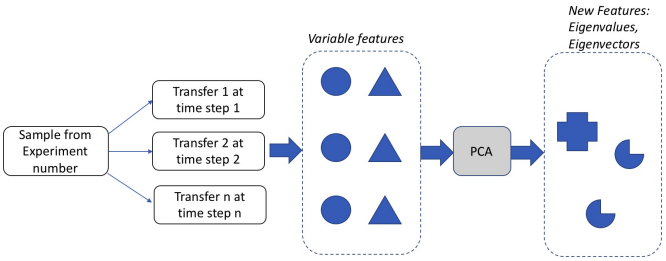


Figure 3: Using PCA to extract new feature sets.

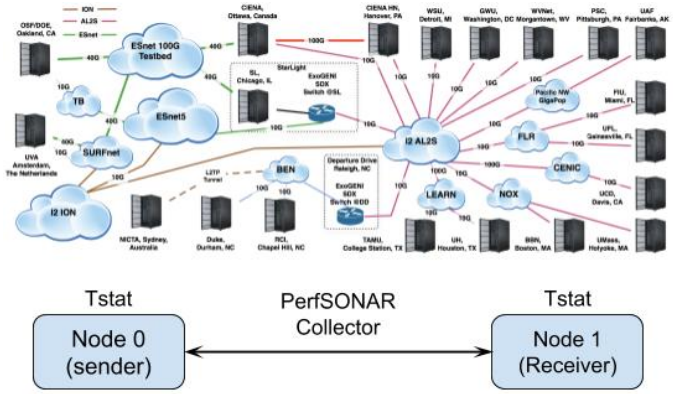


Figure 4: ExoGENI Experimental setup.

provisioned from ExoGENI, sender and receiver. Each node/VM contains 2 cores, and 12GB RAM. The two nodes are connected via a 500 Mbps link.

Each experiment lasts 1 hour, where the sender sends TCP traffic to the receiver using iPerf3 [7]. We generate various types of synthetic network anomalies for different runs using the native Linux Traffic Control (TC) tool, with anomalies outlined in Table 3. We use TSTAT [5] to collect the network measurement data, e.g., RTT, congestion window size, packet count, etc, on the data plane interfaces. In the mean time, we use perfSONAR [6] to actively probe the network performance. PerfSONAR sends probing packets every few seconds, in order to detect network anomalies, including packet loss, duplication, reordering, etc.

5 EVALUATION

5.1 Statistical Analysis

The mean, median and standard deviation of RTT measurements for 8 experimental use cases are shown in Figure 5. For the sender, there is an increasing trend as loss and duplication rate is increased as shown in Figure 5(e), but such trend is less evident for the receiver as shown in Figure 5(f). Interestingly, the variations and standard deviation of RTT also increase with increasing loss and duplication rates for the sender as shown in Figures 5(a) and 5(b), respectively. On the other hand, such trends are not prominent for the receiver.

In addition to TSTAT analysis, simple perfSONAR logs can be used to identify loss as shown in Figure 6 and reordering in Figure 7. However, these graphs are not able to extract features about the data sample, but can be used for real-time detection of loss or reordering being observed.

5.2 Unsupervised Feature extraction using PCA

PCA is a useful method to linearly calculate correlation among the data variables about a transfer. Creating new feature sets based like eigenvectors and eigenvalues, PCA is able to capture most of the information about the transfer data. Figure 8 plots the principal components extracted from offline-analysis of TSTAT data (e.g. TCP window details (max size, min size, cwin max , cwin min), Return time observed (RTO) and round trip times (RTT)). As shown in Figure 3, PCA is able to extract secondary features which can summarize the data. By plotting the principal components 1 and 2 for the complete transfer variables, we are able to extract distinct features for normal traffic versus abnormal traffic scenarios. These principal components have a variance ratio of 0.57 and 0.1, which defines the complete covariance matrix of all the samples. Figure 8 is able to portray that by extracting secondary features (or eigenvalues) for normal traffic traces, and we are able to distinctly find eigenvectors which can be used as independent properties of a normal transfer characteristic. Any abnormal transfer (e.g. with loss, duplication or replication) does not map to the eigenvectors extracted for normal transfers. These results are able to show that PCA can be used for predicting if a transfer is normal versus having abnormal transfer characteristics.

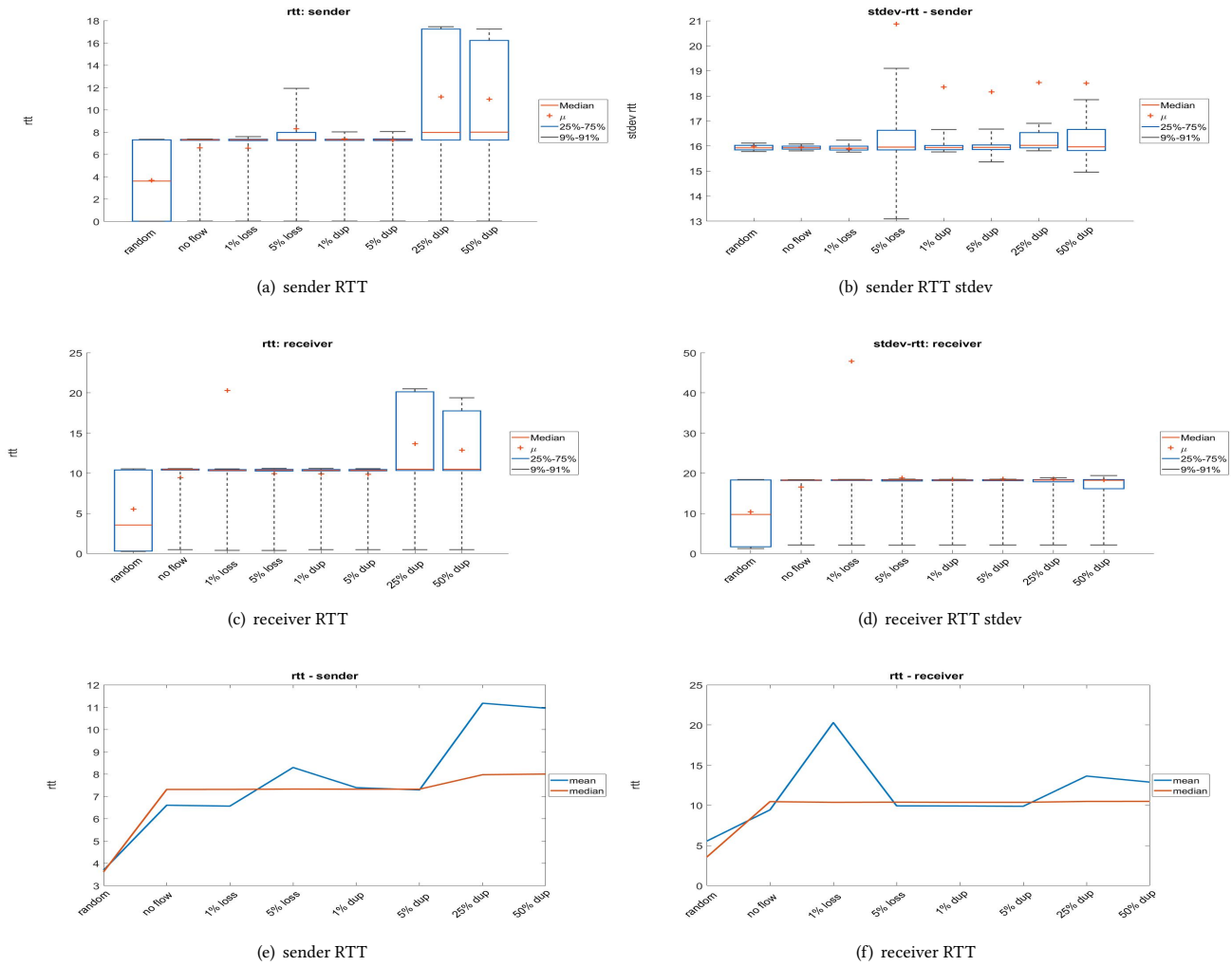
Additionally, in Figure 9, we reduced the initial feature set provided to PCA, only considering the RTT features. The graph shows that this is not enough to extract any meaningful distinctive features about the transfers or be used as a measure for normal versus abnormal character.

6 DISCUSSIONS

Figure 5 shows the statistical analysis on RTT features. RTT is collected at both sender and receiver, and the mean, median and standard deviation is able to show unique characteristics across the sample sets. While we see similar RTT behavior between sender and receiver, in Figures 5(a) and 5(c), in terms of median calculations, the standard deviations show difference across all experiments. In these graphs, it is difficult to extract any independent features that fully describe the characteristics of each transfer sample.

Table 3: Experiment Details

Experiment number	Experiment type (runs for 1 hour)	TSTAT Sample size	perfSONAR sample size
1	Random Traffic (iperf sends traffic at random)	152	120
2	No flow	120	120
3	Loss introduced 1%	120	90
4	Loss introduced 5%	120	50
5	Duplicate packets artificially 1%	60	60
6	Duplicate packets artificially 5%	60	60
7	Reordering packets artificially 25% - 50%	50	50
8	Reordering packets artificially 50% - 50%	60	60



(a) sender RTT

(b) sender RTT stdev

(c) receiver RTT

(d) receiver RTT stdev

(e) sender RTT

(f) receiver RTT

Figure 5: Mean, median and standard deviation of TCP RTT estimates of sender and receiver.

Compared to statistical results, PCA is able to extract unique feature sets that can be used to individually monitor abnormal transfers. However, PCA also has its limitations, being highly dependent on linear characteristics and the variance among the data samples. Still, in this paper it shows simple PCA can group features on small data samples to extract features with unknown prior behavior. This mechanism can be expanded to other unsupervised feature extraction methods, building network monitoring tools that can 'flag' abnormal transfers as soon as it is recorded.

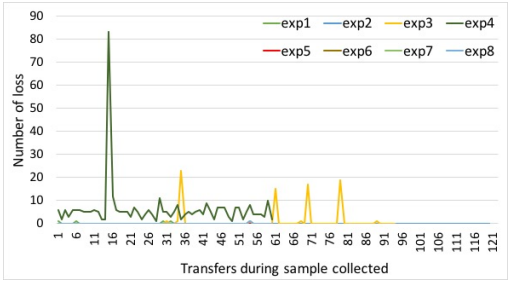


Figure 6: Loss recorded with perfSONAR in experiments.

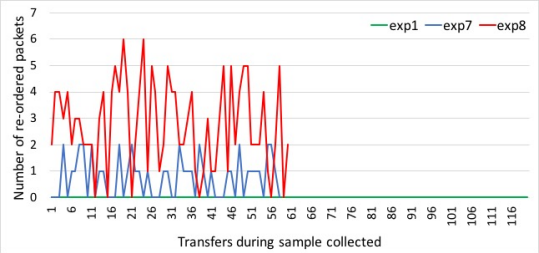


Figure 7: Reordering recorded with perfSONAR.

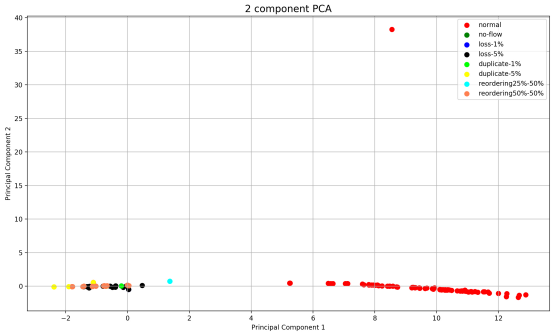


Figure 8: PCA results for TSTAT with RTT, RTO and window details.

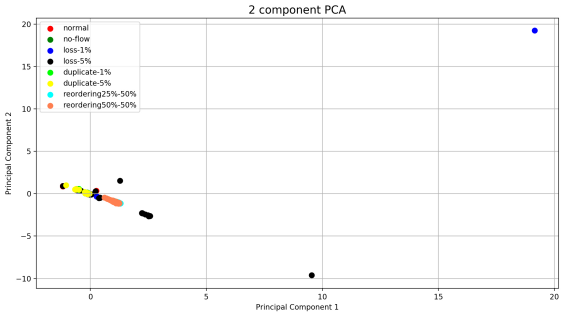


Figure 9: PCA results for just RTT.

7 CONCLUSIONS

Traditionally, machine learning approaches in networks have been based on previously known rules or labeled data sets. But before these techniques can be adapted to networks, we need to identify features that are initially unknown. Identifying characteristics and properties can help identify outlier behavior as soon as it is detected. In this paper, we studied this via statistical and unsupervised feature extraction methods. Results of applying an unsupervised feature extraction method, like PCA, was able to create distinct feature sets (based on eigenvectors) that were seen to be unique to normal transfer characteristics.

In the future, this work will be expanded to more unsupervised feature extraction methods than minimize the variance in PCA outputs. These features can be correlated with workflow performances when larger and complex transfers are chained together as in [15].

Further work will lead to documenting the features of normal traffic and unique features that distinguish loss, versus duplication and replication transfer samples. Our experiments showed that these three abnormal behaviors were not captured with current techniques and requires further investigation.

REFERENCES

[1] Ilya Baldin, Jeff Chase, Yufeng Xin, Anirban Mandal, Paul Ruth, Claris Castillo, Victor Orlikowski, Chris Heermann, and Jonathan Mills. 2016. ExoGENI: A Multi-Domain Infrastructure-as-a-Service Testbed. In *The GENI Book*. 279–315.

[2] N. Bansal and R. Kaushal. 2015. Unusual internet traffic detection at network edge. *International Conference on Computing and Network Communications (CoCoNet)* (2015).

[3] J. But, U. Keller, D. Kennedy, and G. Armitage. 2005. Passive TCP stream estimation of RTT and jitter parameters. In *The IEEE Conference on Local Computer Networks 30th Anniversary (LCN'05)*.

[4] P. Casas, P. Fiandino, S. Wassermann, S. Traverso, A. D'Alconzo, E. Tego, F. Matera, and M. Mellia. 2016. Unveiling Network and Service Performance Degradation in the Wild with mPlane. *IEEE Communications Magazine* 54 (2016), 71–79.

[5] A. Finamore, M. Mellia, M. Meo, M. M. Munafo, P. D. Torino, and D. Rossi. 2011. Experiences of Internet traffic monitoring with tstat. *IEEE Network* 25, 3 (May 2011), 8–14. https://doi.org/10.1109/MNET.2011.5772055

[6] Andreas Hanemann, Jeff W. Boote, Eric L. Boyd, Jérôme Durand, Loukik Kudarimoti, Roman Łapacz, D. Martin Swany, Szymon Trocha, and Jason Zurawski. 2005. PerfSONAR: A Service Oriented Architecture for Multi-domain Network Monitoring. In *Service-Oriented Computing - ICSOC 2005*, Boualem Benatallah, Fabio Casati, and Paolo Traverso (Eds.).

[7] Iperf [n. d.]. Iperf - The network bandwidth measurement tool. https://iperf.fr/. ([n. d.]).

[8] Dingde Jiang, Zhengzheng Xu, Peng Zhang, and Ting Zhu. 2014. A Transform Domain-based Anomaly Detection Approach to Network-wide Traffic. *J. Netw. Comput. Appl.* 40, C (April 2014), 292–306.

[9] Anukool Lakhina, Mark Crovella, and Christophe Diot. 2005. Mining anomalies using traffic feature distributions. *In Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '05)* (2005), 217–228.

[10] Marco Mellia. 2002. TCP Statistic and Analysis Tool. *IEEE Network* 16 (2002).

[11] M. Mellia, M. Meo, L. Muscariello, and D. Rossi. 2008. Passive analysis of TCP anomalies. *Computer Networks* 52 (2008), 663–2676.

[12] M. Mirza, J. Sommers, P. Barford, and X. Zhu. 2010. A Machine Learning Approach to TCP Throughput Prediction. *IEEE/ACM Transactions on Networking* 18, 4 (Aug 2010), 1026–1039.

[13] Luca Muscariello, Marco Mellia, and Michela Meo. 2006. Passive Identification and Analysis of TCP Anomalies. *Distributed Cooperative Laboratories: Networking, Instrumentation, and Measurements* (2006).

[14] D. Rossi, M. Mellia, and C. Casetti. 2003. User patience and the Web: a hands-on investigation. *Global Telecommunications Conference* (2003).

[15] Ozan Tuncer, Emre Ates, Yijia Zhang, Ata Turk, Jim Brandt, Vitus J. Leung, Manuel Egele, and Ayse K. Coskun. 2017. Diagnosing Performance Variations in HPC Applications Using Machine Learning. In *High Performance Computing*, Julian M. Kunkel, Rio Yokota, Pavan Balaji, and David Keyes (Eds.). Springer, 355–373.

[16] J. Zhang and M. Zulkernine. 2006. Anomaly based network intrusion detection with unsupervised outlier detection. *IEEE Communications* (2006).