Revisit Network Anomaly Ranking in Datacenter Network Using Re-ranking

Shaohan Huang[‡], Carol Fung [†],Kui Wang[*‡], Yaqi Yang[‡], Zhongzhi Luan[*‡], Depei Qian[*‡]
[*]Beijing municipal key laboratory of network technology, Beihang University, Beijing, China
[†]Computer Science Department, Virginia Commonwealth University, Richmond, Virginia, USA
[‡]Sino-German Joint Software Institute, Beihang University, Beijing, China
buaahsh@foxmail.com, cfung@vcu.edu, kwang@buaa.edu.cn,
yaqi.yang@buaa.edu.cn, zhongzhi.luan@jsi.buaa.edu.cn, depeiq@buaa.edu.cn

***Abstract*—With the continuous growth of modern datacenter networks in recent years, network intrusions targeting those datacenters have also been growing rapidly. In this situation, system monitoring and intrusion detection become essential to control the risks of such networks. There are many network anomaly detection systems being used to identify significant anomalies in datacenter networks. However, they often focus on detecting significant anomalies, while ignoring insignificant anomalies oftentimes. Existing anomaly ranking models are not accurate in detecting insignificant anomalies. This becomes an issue when attacks are from insignificant anomaly traffic. In this paper, we revisit the network anomaly ranking problem and propose a re-ranking model based on a commonly used unsupervised network anomaly ranking method. We introduce several new features into the re-ranking model to capture extra information about outliers. Our experimental results based on real datacenter network data demonstrate that the proposed re-ranking model improves the ranking quality over the unsupervised method, especially for insignificant outliers.**

I. Introduction

With the continuous advances of datacenter networks and datacenter applications, the datacenter has become indispensable infrastructure to our daily lives. At the same time, various types of datacenter network attacks also become increasingly rampant, such as computer viruses, worms, trojan, and distributed denial of service. Those attacks are threatening the security of datacenters. Known attacks can be detected by signature-based intrusion detection technologies. On the other hand, anomaly detection technologies are commonly employed detect unknown attacks in datacenters. Many types of anomaly detection methods for datacenter protection have been proposed in the past few years, such as methods based on statistical theory [1], data mining [2], machine learning [3], and a combination of the above methods.

In the literature, most network anomaly detection methods focus on either anomaly detection or anomaly taxonomy. Anomaly detection commonly adopts binary classification models. The main purpose is to determine whether the observed traffic is normal or not. The anomaly taxonomy improves the anomaly detection, by further identifying the types of abnormality, such as scanning or denial of service. Since the anomaly taxonomy provides the type of attacks for the detected anomalies, it is no longer a simple binary classification, nor a simple simple multi classification. Most works in the literature focus on either anomaly detection or anomaly taxonomy, few have provided the ranking of anomalies for datacenter monitoring. Although extensive research on network traffic anomaly detection have been conducted and various types of anomaly detection systems have been proposed, little research has been done on anomaly ranking. Anomaly ranking can be seen as a ranking model instead of a classification model.

In reality many reputable network anomaly detection techniques [4], [5] have been adopted by datacenters, and have shown effectiveness in identifying significant anomalies. Their detection methods are commonly based on identifying outliers from the monitored traffic. A common way of ranking ourliers is based on their distances from the centroid of all normal points. The further an outlier is from the centroid, the more abnormal the outlier is. Distance to the centroid can be an effective metric to rank significant anomalies. However, we found this method has poor performance in ranking insignificant outliers, which degrades the effectiveness of anomaly detection accuracy. For example, an attacker who uses Distributed Denial of Service attacks (DDoS) may only start to attack when there is a large volume of data being exchanged between nodes. In this case, it is necessary to investigate insignificant anomaly.

To improve anomaly ranking for insignificant outliers, we propose a new anomalies ranking model, which integrates a re-ranking model based on unsupervised anomaly detection algorithm. We employ a support vector regression model for the re-ranking task. The re-ranking model reuses features from the unsupervised learning model and also introduces new features based on the k-nearest neighbours algorithm. In order to evaluate the results of the re-ranking model, we propose a novel evaluation method for outliers ranking based on Discounted Cumulative Gain. Evaluation results demonstrate that the re-ranking model can improve over the unsupervised anomaly ranking model by 8% in average, and the new ranking result can improve the detection rate of insignificant abnormalities.

The main innovation and contributions of this paper are listed as follows:

- Our work takes a different approach to visit the anomaly detection problem, by ranking the degree of abnormality

in datacenter network.
- We propose a re-ranking model based on unsupervised network anomaly system and using the new features to overcome some sorting deficiency than the unsupervised model.
- We propose a new evaluation method to evaluate our anomaly re-ranking model.

The remainder of the paper is organized as follows: Section 2 gives an overview of work about network anomaly detection in the literature; Section 3 presents our anomaly re-ranking system and our proposed evaluation method MDCG; In Section 4 we conduct anomaly ranking experiments using datacenter network traffic dataset and the corresponding results. Finally, in section 5, we conclude this paper and summarize our future work.

II. Related Work

Network anomaly detection has become an area of active research during the last decade. Methods for detecting network anomalies can be broadly classified into statistical theory [1], data mining [2], classification and clustering [6]. Our work focuses on unsupervised methods and ranking methods.

The authors present an unsupervised method for classification and characterization of security related anomalies, and attacks occurring in honeypots. They use a clustering technique based on sub-space-density clustering to identify clusters and rank outliers in multiple low-dimensional spaces.

Delimargas et al. [7] reports an optimized version of Principal Component Analysis (PCA) to detect network-wide traffic anomalies in highly aggregated traffic flows. While there are some limitations on the classic PCA methods, the modified method can possess promising capabilities to efficiently detect network anomalies.

In [4] authors use two feature extractors: a data-dependent normalization and a spectrum kernel for mapping data elements to a feature space. They also introduce three different clustering algorithms: fixed-width clustering, optimized k-NN, and one class SVM to detect network anomaly based on the 1999 KDD Cup data-set.

Reference [5] reports that the unsupervised cluster-based method can detect new intrusions. Authors use a hierarchical clustering method to cluster data and use the standard Euclidean distance for inter-patterns similarity, from the same data-set with [4].

Density-based and grid-based clustering algorithms are also suitable for unsupervised anomaly detection. The authors of [8] combine density-grid-based clustering algorithm to improve computational complexity, obtaining similar detection results.

There has been many work in the field of unsupervised anomaly detection. Most methods are based on clustering and outliers ranking. However, to the best of our knowledge, our approach is the first to focus anomaly ranking and detect the insignificant abnormalities in datacenter network.

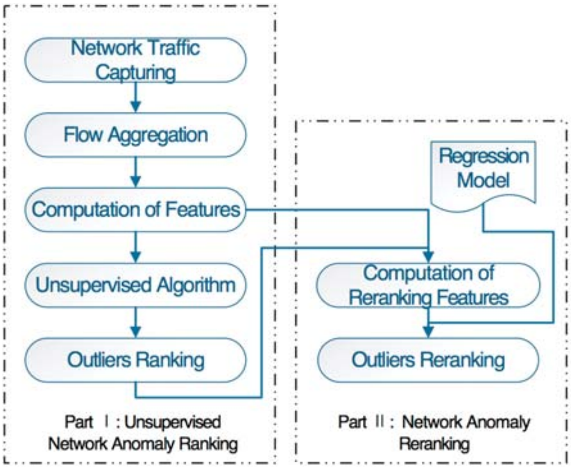
Network Traffic Capturing
Flow Aggregation
Computation of Features
Unsupervised Algorithm
Outliers Ranking
Part Ⅰ : Unsupervised Network Anomaly Ranking
Regression Model
Computation of Reranking Features
Outliers Reranking
Part Ⅱ : Network Anomaly Reranking

Fig. 1. Overall Structure of Network Anomaly Re-ranking System

III. Network Anomaly Ranker and Evaluation Method

In this section, we first describe at a high-level design of our anomaly ranking system and its difference from other network anomaly detection systems. Then we present in detail of our re-ranking model and new features selection. Finally, we demonstrate the evaluation results for our anomaly ranking model.

A. Overview of Network Anomaly Ranking System

Figure 1 shows our system structure and work flow. The network anomaly ranker system consists of two parts - unsupervised network anomaly ranking and network anomaly re-ranking.

The first part of the system has a similar structure with a conventional network anomaly detection system such as the unsupervised network anomaly detection (UNADA) [9]. It consists of five sub-parts: network traffic capturing, flow aggregation, features, selection unsupervised algorithm and outlier ranking.

Our system performs network anomaly ranking on single-link packet-level traffic, which are captured and aggregated into IP flows. IP flows are analyzed using either IPsrc aggregation key or IPdst aggregation key [9]. These classifications can describe different spatial structure anomalies, including 1-to-N anomalies and N-to-1 anomalies. 1-to-1 anomalies can be seen as a special case of 1-to-N or N-to-1, and N-to-N can be treated as multiple N-to-1 or 1-to-N anomalies.

Selecting features from aggregated flows is a key to detect anomalies [10]. We use 10 features in this paper: number of source/destination IP addresses and ports [11], ratio of the number of sources to the number of destinations [8], packet arrival rate, ratio of ICMP and SYN packets [12], average packet size, and ratio of the number max packet size to the average packet size. According to Fernandes's work on signature-based anomaly-characterization [3], these traffic features can be used to describe common network attacks such as DoS, DDoS, port scans, and worms/virus propagation.

Next we use several unsupervised algorithms, such as UNADA [9], PCA [7], DBSCAN [13] and one-class SVM [4], for the purpose of anomaly detection. These algorithms are commonly used for anomaly detection. They can identify outliers, which are remarkably different from the rest of the samples. After all, an outlier is a sample that does not belong any cluster.

The outliers ranking module determines the level of difference of these outliers and rank these anomalies based on the degree of difference. The core process of outliers ranking is the computing of the distance between the outlier and the centroid of normal data samples. The further the distance is, the higher the degree of abnormality is. We then sort these anomalies according to computed distances and output the sorted anomalies sequence as the ranking result.

The second part of our system is network anomaly re-ranking, which consists of re-ranking features selection and outliers re-ranking. In addition to the features in ranking subsystem, we add several new features into re-ranking process, including the score of the unsupervised system, ranking sequence, K-normality, and K-density of the outliers. Outliers re-ranking uses support vector regression model to detect and sort these anomalies using the new features. The details of this part will be described in next the chapters.

B. New Features Selection and Re-ranking Model

Anomaly re-ranking model system uses the output sequence of unsupervised detection as its input and reorders the sequence to reach the final output of our system.

The first challenge of the reordering system is the feature selection. In an unsupervised anomaly detection system, the distance (Euclidean distance or Mahalanobis distance) is commonly used to measure the level of abnormality. This way detectors can find the significant outliers easily, but may ignore insignificant outliers due to the low sensitivity. Therefore we need more new feature to overcome this problem.

In addition to 10 unsupervised models features, we add four new features into the reordering model. Because the input of the re-ranking model is output of unsupervised subsystem, both unsupervised model detected outliers. The four new features we have chosen are: sorted fraction in the unsupervised model, reciprocal of position in the old sequence, K-normality and K-density.

The K-normality and K-density are the two compound features that further describe the characters of outliers. The value of K-normality describes the condition of the k nearest neighbor points around the outliers. If we select k nearest points around the outlier, the value of j/k means there are j points labeled as abnormal based on the former unsupervised learning model. The K-normality values range from 0 to 1.

The idea of the K-density came from some density-based machine learning algorithms such as DBSCAN. The intuition is that the further an outlier is from its neighbors, the more abnormal it should be. K-density can be calculated using the ratio of average distance between the outlier and its K nearest points to the distance from the centroid of all normal points. As shown in Figure 2, all normal points are marked red, all anomaly points are marked black and the green star is the centroid of all normal points. In Figure 2 the outlier A is should be ranked more abnormal than the outlier B since it is further separated from the mass. However, the outlier A is closer to the centroid of all the normal points (green star) compared to outlier B, so it will be ranked to be less abnormal than B based on Euclidean distance. The use of the K-density feature can fix this ranking error.

After including new features, we apply a regression model in our network anomaly re-ranking. In statistics, regression analysis is a statistical process for estimating the relationships among variables.

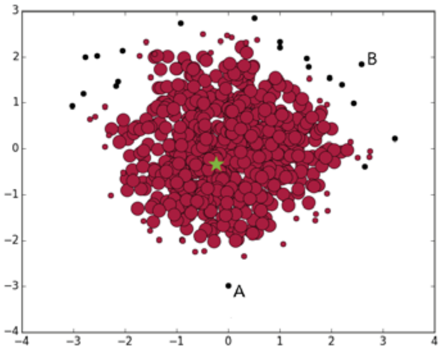

Fig. 2. Special anomaly situation

Regression analysis is a supervised machine learning method. Before running the model, we need labelled history traffic as training data to build such regression model. The training data is generated form the output of unsupervised anomaly detection system on old traffic capturing. Then we detect new outliers based on this model.

Support Vector Regression was chosen for the re-ranking task. Support Vector Machines are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis [14]. The method of Support Vector Classification can be extended to solve regression problems. This method is called Support Vector Regression (SVR). SVR contains all the main features to support a maximum margin algorithm [15]. In SVR a loss function is used to penalize regression errors that are greater than a threshold. Such loss functions usually lead to the sparse representation of the decision rules, resulting in significant algorithmic and representational advantages.

C. Effective Evaluation Method

In this subsection, we propose a new ranking evaluation method for outliers ranking based on Discounted Cumulative Gain (DCG). We modified DCG to match anomaly ranking named MDCG.

We use an example to explain the evaluation methods. Suppose there are i anomalies, j suspicious and k notices in the output sequence. Besides these abnormal cases, there are l irrelevant examples in this sequence. An ideal ranking result should start with the i anomalies, followed by j suspicious and end with k notices. The more consistent the arrangement of ranking results are with the ideal ranking sequence, the higher score it achieves.

An effective evaluation method should satisfy the following three conditions:

1) Scores are relevant to both the degree of anomaly and the position of each anomaly item.
2) Different level of anomalies have different weights.
3) Front position anomaly scores higher than rear position.

In order to find a better measuring algorithm for anomaly ranking, we investigated some common ranking evaluation methods such as Precision@K (P@K), Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), and Normalized Discounted Cumulative Gain (NDCG). P@K, MAP, and MRR use binary relevance, NGCG uses multiple levels of relevance, which is more suitable for anomaly ranking.

DCG is a popular measuring method to evaluate web search and ranking tasks. Using a graded relevance scale of documents in a search engine result set, DCG measures the usefulness, or gain, of a document based on its position in the result list. The gain is accumulated from the top of the result list to the bottom, with the gain of each result discounted at lower ranks [16].

The Discounted Cumulative Gain accumulated at a particular rank position p is defined as:

$$DCP_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{log_2(i+1)} \tag{1}$$

Where p represents the ranked position and rel_i is the graded relevance of the result at position i.

In order to be cross-query comparable, producing the maximum possible DCG till position p as Ideal DCG (IDCG) and normalized discounted cumulative gain (nDCG) is ratio of DCG of IDCG, which is computed as:

$$nDCP_p = \frac{DCP_p}{IDCP_p} \tag{2}$$

DCG computes Cumulative Gain of the items on top p positions, disregarding the other positions. That is reasonable for information retrieval. The lower the ranked position of a relevant document, the less useful it is for the user, since it is less likely to be examined. However, our anomaly ranking uses a different approach. Besides the top p positions, we accept other anomaly cases that can be ordered at more forward position. Therefore we modified the DCG formulation as following:

$$MDCP_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{log_2(pos_i)} \tag{3}$$

Where p represents the number of abnormal cases and is the position in sequence for i abnormal case. The is weight of i abnormal example. We can also compute the normalized MDCG where IMDCG is the Ideal MDCG.

IV. Experimental Evaluation

We evaluate the effectiveness of network anomaly re-ranking model using the real data set from the public MAW-I repository of the WIDE project [16] and the datacenter network dataset collected by our monitoring system from Amazon Web Services (AWS). Our experimental results show that the new ranking model can benefit the ranking result of insignificant abnormalities and achieve satisfying result in MDCG.

A. Data Collection

We have collected labelled network traffic traces to evaluate our system. MAWILab is a database for researchers to evaluate their traffic anomaly detection methods. The dataset collected using tcpdump consists of daily raw packet traces of 15 minutes long each day for the last fourteen years.

Besides the MAWI traffic dataset, we also collect IP level datacenter network logs from our AWS nodes. We deployed monitoring endpoints in 8 AWS nodes over two months. We used tcpdump toolkit to collect our datacenter data, which has the same data structure with the MAWI dataset.

Combining the results from multiple distinct and independent anomaly detectors, MAWI applies a voting mechanism to generate labelled traces. In the MAWI archive, it annotates traffic anomalies with four different labels: anomalous, suspicious, notice, and benign. The label anomalous is assigned to all abnormal traffic and should be identified by any effective anomaly detector. The suspicious is assigned to all traffic that is probably anomalous but not clearly identified by MAWI methods. The label notice is assigned to all traffic that has been reported by at least one anomaly detector. All the other traffic are labeled benign because none of the anomaly detectors identified them. We use the same voting mechanism to generate labelled dataset traces in our datacenter network traffic.

We refer the MAWI traffic labelled dataset and our collected datacenter dataset as our ground truth. If the ranking model performs great, it can replace several anomaly detectors in network anomaly detection.

B. Building Support Vector Regression Model

Support Vector Regression Model is a supervised machine learning method. Unlike unsupervised algorithms, it requires training data and the model needs to be trained before prediction.

All traffic in the dataset is annotated with four different labels: anomalous, suspicious, notice, and benign. Anomalous, suspicious and notice are three different levels of abnormal and should be ordered accordingly by ranking system. To differentiate the difference between the different levels of anomaly, we assign different weights to each type. We weight anomalous to be 1, suspicious to be 0.8 and notice to be 0.05. These weights are applied in training data generation and the MDCG formulation for evaluation.

As we mentioned before, in the re-ranking sub system, we need build the Support Vector Regression model to reorder the

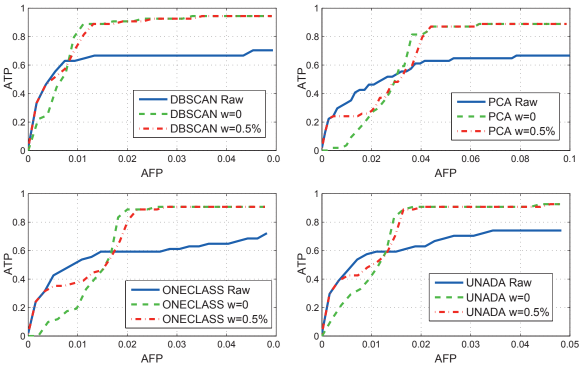

Fig. 3. Average True Positives(ATP) Rate vs Average False Positive(AFP) Rate in MAWI

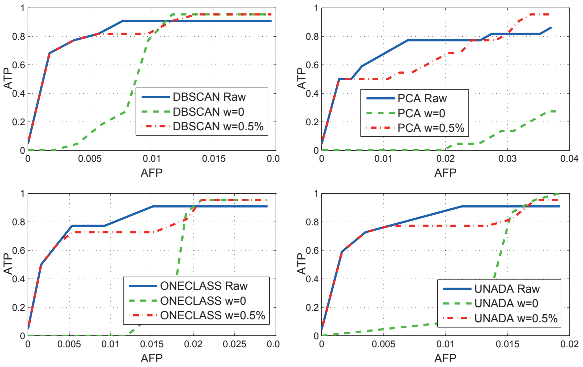

Fig. 4. Average True Positives(ATP) Rate vs Average False Positive(AFP) Rate in MAWI

output of unsupervised detector. The labelled data are divided into two parts: training dataset and testing dataset.

In the training phase, traffic data shall be sorted by unsupervised system. We use the output of the unsupervised detector, which is a sequence of sorted data points and real sequence result as input for the regression model. We then transform the sequence into an array of class labels. For regression model, these class labels are dependent variables and they are continuous values. We generate class labels based on anomaly weights. Our strategy is selecting the max score of unsupervised detector as C (ceiling). The formulation of class labelling can be written as: $Label = C \times w$.

Where w is the weight of the label class. The advantage of using the weight of abnormal label is that it is consistent with the evaluation method. It means that method of generating training data is reasonable.

After the training dataset is ready, we fit the SVR model with scikit-learn [17] , which is a machine learning toolkit in Python. It is a simple and efficient tool for data mining and data analysis. We train SVR model with the Radial Basis Function (RBF) kernel, where two parameters must be considered: C and γ. The parameter C can be used to tune the trad-off between the fitting on samples and smoothness of the outcome. A low C makes the decision surface smooth, while a high C aims at classifying all training examples correctly. In our model, we set C 1.0. γ is another parameter defines how much influence a single training example has. We set $\gamma = 0$ considering our traffic data is unbalanced.

C. Ranking outlier in labelled traffic with Ground Truth

To evaluate our proposed re-ranking system, we build our system following the design shown in Figure 1. We compare the performance of our re-ranking model with other models using different clustering algorithms.

Figure 3 presents the Receiver Operating Characteristic (ROC) curve in the detection of anomalies in MAWI, using IPsrc as key. We compare the performance of four common unsupervised algorithms: UNADA, DBSCAN, PCA and one-class SVM as the selected learning model in part I. All these methods use the Mahalanobis distance to their centroid as ranking criteria. In Figure 3, the ROC curve reports the performance of detecting anomalous classes, which is more significant than other labels. Figure 4 presents the ROC curve in the detection of the whole abnormal points including anomalous, suspicious and notice.

In the two figures, Raw lines represent the ranking result of unsupervised algorithms and $w = 0$ line shows the performance of re-ranking model when unsupervised output are fully re-ordered. We can see that with re-ranking the performance drop down using anomalous classes as standard. But in Figure 4 , when we evaluate the entire abnormal traffic, the re-ranking method can achieve better performance. This outcome shows that significant anomalies have been detected by unsupervised model. Re-ranking algorithm will affect the ranking position for most significant anomalies. Through Figure 4 we can see that the re-ranking model can handle the insignificant cases and get better ROC than unsupervised method.

Through the above experimental results, we decide the best strategy is to re-rank the insignificant anomalies while keep the significant anomalies in place. That will not only affect significantly abnormal results, but also improve non-significant abnormal ranking. At the same time we can use MDCG to evaluate our methods.

In the next experiment, we do not change the top w percent of the unsupervised model outputs, re-order the rest of sequence using regression model.

As shown in Figure 3 and Figure 4, using the notable exception as evaluation standards, while maintaining the previous 0.5%, the results of unsupervised models are mostly the same as the re-ranking algorithm. When we include all abnormal points into the evaluation, the ranking quality is improved after re-ranking model.

Table 1 shows the performance of our system under different clustering models and w settings. We can see that when w is 0, MDCG value declines compared to the model without re-ranking. Despite the fact that figure 4 shows better ROC curve when w is 0 compared to the raw method, MDCG decreases since significant abnormality have more weight. When we slightly increase the portion of the top w% of most significant data points without re-ranking them, MDCG value increases. Among the four selected clustering algorithms, UNADA shows the best performance. For all the four different algorithms, the average enhancement is 8% through re-ranking. We can conclude that the re-ranking model can improve the quality of network anomaly ranking.

TABLE I
MDCG VALUE FOR MAWI DATA

-	DBSCAN	UNADA	SVM	PCA
Raw	59.50%	61.80%	56.74%	55.38%
0	57.48%	57.38%	52.49%	51.05%
0.1%	**69.19**%	68.28%	63.03%	55.62%
0.5%	68.98%	**69.47**%	**64.45**%	**62.63**%
1%	67.90%	68.81%	64.17%	62.57%
5%	64.70%	65.92%	61.33%	61.29%
10%	63.52%	65.15%	60.62%	60.14%
Max Increase	9.69%	7.67%	7.71%	7.29%

Then we set the threshold w to 0.5% and conduct experiments in our collected AWS datacenter network traffic using IPsrc and IPdst as keys. We further demonstrate the performance of re-ranking system using IPsrc as key in Table 2 and using IPdst as key in Table 3. The results show that in both cases, the re-ranking can enhance the anomaly detection by a gain of 6-8% to 9-11% respectively. Thus, we have verified the effectiveness of our system to rank network anomaly in datacenter network.

TABLE II
MDCG VALUE FOR DC DATA, IPSRC KEY

-	DBSCAN	UNADA	SVM	PCA
Raw	61.48%	63.37%	57.32%	53.11%
0.5%	69.32%	70.29%	65.34%	61.93%
Increase	7.84%	6.92%	8.02%	8.82%

TABLE III
MDCG VALUE FOR DC DATA, IPDST KEY

-	DBSCAN	UNADA	SVM	PCA
Raw	54.21%	58.45%	56.28%	52.33%
0.5%	63.85%	67.64%	67.34%	61.47%
Increase	9.64%	9.19%	11.06%	9.14%

V. CONCLUSION AND FUTURE REMARKS

In this paper we take a different approach to solve the anomaly detection problem through anomaly ranking. We propose a re-ranking model that is based on an unsupervised learning model plus another round of ranking adjustment by introducing some additional features. We compare our model with previous work in the field of unsupervised anomaly ranker in datacenter network environment. Our results based on real data show that our re-ranking model can improve the detection rate on insignificant anomalies. New model can improve the detection rate of insignificant abnormalities.

As our future work, we plan further experimental evaluation of our re-ranking model in cloud network environment. Additional work is to integrate our anomaly ranking system with a diagnosis engine that can not only rank anomaly network but can also determine the cause of a ranked anomaly.

VI. ACKNOWLEDGEMENT

This research is supported by the National High Technology Research and Development Program of China (Grant No.2012AA01A302), the National Natural Science Foundation of China (Grant No. 61361126011).

REFERENCES

[1] Marina Thottan and Chuanyi Ji. Anomaly detection in ip networks. *Signal Processing, IEEE Transactions on*, 51(8):2191–2204, 2003.

[2] Wenke Lee, Salvatore J Stolfo, and Kui W Mok. A data mining framework for building intrusion detection models. In *Security and Privacy, 1999. Proceedings of the 1999 IEEE Symposium on*, pages 120–132. IEEE, 1999.

[3] Guilherme Fernandes and Philippe Owezarski. Automated classification of network traffic anomalies. In *Security and Privacy in Communication Networks*, pages 91–100. Springer, 2009.

[4] Eleazar Eskin, Andrew Arnold, Michael Prerau, Leonid Portnoy, and Sal Stolfo. A geometric framework for unsupervised anomaly detection. In *Applications of data mining in computer security*, pages 77–101. Springer, 2002.

[5] Leonid Portnoy, Eleazar Eskin, and Sal Stolfo. Intrusion detection with unlabeled data using clustering. In *In Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001*, 2001.

[6] Debdutta Barman Roy and Rituparna Chaki. State of the art analysis of network traffic anomaly detection. In *Applications and Innovations in Mobile Computing (AIMoC), 2014*, pages 186–192. IEEE, 2014.

[7] Athanasios Delimargas, Emmanouil Skevakis, Hassan Halabian, Ioannis Lambadaris, Nabil Seddigh, Biswajit Nandy, and Rupinder Makkar. Evaluating a modified pca approach on network anomaly detection. In *Next Generation Networks and Services (NGNS), 2014 Fifth International Conference on*, pages 124–131. IEEE, 2014.

[8] Kingsly Leung and Christopher Leckie. Unsupervised anomaly detection in network intrusion detection using clusters. In *Proceedings of the Twenty-eighth Australasian conference on Computer Science-Volume 38*, pages 333–342. Australian Computer Society, Inc., 2005.

[9] Pedro Casas, Johan Mazel, and Philippe Owezarski. Unada: unsupervised network anomaly detection using sub-space outliers ranking. In *NETWORKING 2011*, pages 40–51. Springer, 2011.

[10] Nigel Williams, Sebastian Zander, and Grenville Armitage. A preliminary performance comparison of five machine learning algorithms for practical ip traffic flow classification. *ACM SIGCOMM Computer Communication Review*, 36(5):5–16, 2006.

[11] Anukool Lakhina, Mark Crovella, and Christophe Diot. Mining anomalies using traffic feature distributions. In *ACM SIGCOMM Computer Communication Review*, volume 35, pages 217–228. ACM, 2005.

[12] Anukool Lakhina, Mark Crovella, and Christiphe Diot. Characterization of network-wide anomalies in traffic flows. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, pages 201–206. ACM, 2004.

[13] Philippe Owezarski. Unsupervised classification and characterization of honeypot attacks. In *Network and Service Management (CNSM), 2014 10th International Conference on*, pages 10–18. IEEE, 2014.

[14] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[15] Steve R Gunn et al. Support vector machines for classification and regression. *ISIS technical report*, 14, 1998.

[16] Kenjiro Cho, Koushirou Mitsuya, and Akira Kato. Traffic data repository at the wide project. In *Proceedings of the annual conference on USENIX Annual Technical Conference*, pages 51–51. USENIX Association, 2000.

[17] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research*, 12:2825–2830, 2011.