Paper No: SC-10 **Smart Computing**

Anomaly detection in cloud network data

Tharindu Lakshan Yasarathna*
Software Engineering Teaching Unit
Faculty of Science,
University of Kelaniya, Sri Lanka
tharinduyasarathna@gmail.com

Lankeshwara Munasinghe
Software Engineering Teaching Unit
Faculty of Science,
University of Kelaniya, Sri Lanka
lankesh@kln.ac.lk

Abstract: **Cloud computing is one of the most rapidly expanding computing concepts in the modern IT world. Cloud computing interconnects data and applications served from multiple geographic locations. A large number of transactions and the hidden infrastructure in cloud computing systems have presented a number of challenges to the research community. Among them, maintaining the cloud network security has become a key challenge. For example, detecting anomalous data has been a key research area in cloud computing. Anomaly detection (or outlier detection) is the identification of suspicious or uncommon data that significantly differs from the majority of the data. Recently, machine learning methods have shown their effectiveness in anomaly detection. However, identifying anomalies or outliers using supervised learning methods still a challenging task due to the class imbalance and the unpredictable nature and inconsistent properties or patterns of anomaly data. One-class classifiers are one feasible solution for this issue. In this paper, we mainly focused on analyzing cloud network data for identifying anomalies using one-class classification methods namely One Class Support Vector Machine(OCSVM) and Autoencoder. Here, we used a benchmark data set, YAHOO Synthetic cloud network data set. To the best of our knowledge, this is the first study that used YAHOO data for detecting anomalies. According to our analysis, Autoencoder achieves 96.02 percent accuracy in detecting outliers and OCSVM achieves 79.05 percent accuracy. In addition, we further investigated the effectiveness of a one class classification method using another benchmarked data set, UNSW-NB15. There we obtained 99.10 percent accuracy for Autoencoder and 60.89 percent accuracy for OCSVM. The above results show the neural network-based methods perform better than the kernel-based methods in anomaly detection in cloud network data.**

Keywords: *Anomaly Detection, Cloud Computing, Machine Learning, One-Class Classification*

I. INTRODUCTION

Cloud computing is one of the most rapidly expanding computing concepts in the modern IT world. Cloud computing interconnects data and applications served from multiple geographic locations. Emerging Internet technologies have extended the capabilities of cloud computing which is regarded as an upgraded version of utility computing as well. For example, cloud computing technologies are widely used for subscription-based or pay-per-use services [1]. Using multitenant architecture, Cloud computing delivers a single application via a browser to millions of users in different geographical areas around the world. This technology is called SaaS (Software as a service). Since, Cloud computing involved with thousands of user transactions, information, and communication, cloud security is considered one of the most important aspects of cloud computing. To provide secure cloud computing platforms or services, the availability, integrity, and confidentiality need to ensure. However, a high volume of transactions happens within nanoseconds and the hidden infrastructure of cloud computing are some of the challenging factors for ensuring the security of the cloud computing systems [2]. On the other hand, security attacks are not known to anyone. Some attackers execute masked attacks and, gradually leak sensitive information. To avoid this kind of threat, many approaches and methods have been introduced for security monitoring in cloud computing [3]. Among them, anomaly detection using machine learning algorithms is a widely used approach. Anomaly detection (or outlier detection) is the identification of suspicious or unusual data that significantly differs from the majority of the data. Identifying anomalies or outliers using supervised learning methods is a challenging task because the nature or properties of outliers are not consistent. This paper mainly focuses on analyzing cloud networks and identify abnormal activities in cloud network data using machine learning methods. Here, we used YAHOO Synthetic and real time-series with labeled anomalies data set [4]. To best of our knowledge, this is the first study that uses YAHOO synthetic data for anomaly detection. In addition, we further investigated the effectiveness of a one-class classification method using another benchmarked data set, UNSW-NB15 [5]. We will present our research in detail in the next section. In the following section, we discuss some of the existing state-of-art research related to our topic. In section 2, we present our study with a detailed explanation followed by experimental analysis. We conclude this paper by discussing some open questions to the research community and, our future direction of the research.

II. RELATED WORK

It is a well-known fact that cloud computing has added a new computing paradigm. Like real clouds contain the ice crystals and water drops, the explanation of ‘cloud’ in cloud computing is the collection of networks [6]. Cloud computing extends capabilities of the IT over the internet and it enables subscription-based or pay-per-use services. Users can select cloud plans depending on their requirements and available budget. As a consequence, modern businesses are using cloud computing platforms personalized services to their users. Thus, cloud security has become one of the main factors affects the reliability of cloud computing. For example, especially when confidential information shares over the cloud services [7]. Confidential personal data stored on remote servers provide data availability when necessary through the cloud network which can be crucially vulnerable for major security threats. In addition, sharing data in the cloud is a problem when the cloud service provider is not trusted [8]. Besides, they need to respond to fast-moving changes in data promptly, especially in case of security threats. Unfortunately, there has been no effective ways to

handle and analyze constantly growing datasets manually in the cloud [3]. For example, detecting outliers or anomalies is vital to prevent potential security threats. Anomaly detection refers to the identification of elements or events that are not suitable for the expected pattern or other elements of a data set that is not normally detectable by human experts. This anomaly usually translates into problems such as structural defects, errors or fraud [9].

Machine learning methods have shown their effectiveness in finding anomalies or outliers. Machine learning types can be classified as unsupervised learning, semi-supervised learning, and supervised learning. The progress made in anomaly detection has been mostly supported approaches using supervised machine learning algorithms that need big labeled data-sets to be trained. However, supervised machine learning methods still having difficulties because learning from historical anomaly data and may not effective in forecasts future potential anomalies. Because we cannot pre-defined anomalies [10] and anomalies or outliers do not exhibit consistency patterns or properties. One possible way to solve this problem is training a model to effectively recognize non-anomaly data. For example, one-class classifiers can train to recognize non-anomaly data. In this research, we have used two one-class machine learning algorithms to detect anomalies in the cloud network. They are a one-class support vector machine and Autoencoder. One class support vector machine is the extension of SVM for unlabeled data, could be used for anomaly detection [11] [12]. Autoencoder is a type of neural network-based learning algorithm, which has one approach to automatically learn features from unlabeled data [13].

III. ANOMALY DETECTION IN CLOUD NETWORK

A surge of cloud computing has presented a number of challenges to the research community. Among them, identifying anomalous data is an immense challenge due to the complexity, heterogeneity and dynamic behavior of the data. Anomalies and outliers pose a huge security threat. Therefore, many types of research have been carried out on detecting anomalies in cloud network data. In this section, we discuss the practical use case for anomaly detection in cloud network data with a one-class classification approach.

A. What is anomaly data?

Cloud network services provide their service to users through secured methods such as authorized access. However, it is difficult to ensure that the data received will always be normal. Some data differing significantly from the majority of the data are classified as the anomaly data. Fig. 1 shows some examples of anomaly data in cloud networks.

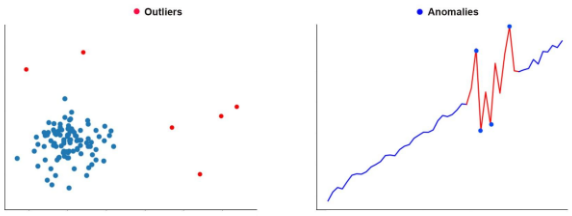

Fig. 1. Anomaly graphs

The first figure of Fig. 1 shows that normal data (blue dots) are clustered around a particular point and the outliers or anomaly data (red dots) are far away from the cluster centroid compared to the normal data. In the second figure, the sequential pattern of data has disrupted in the middle and the anomalous data points lay away from the normal data sequence.

B. One-class classification for anomaly detection

In this research, we choose a supervised machine learning approach for anomaly detection in cloud network data. Our task is to classify data into two classes; anomalies or non-anomalies. Supervised learning methods need rich data to provide good models [14]. Especially, the training data should contain enough data points to represent all possible scenarios. However, in most cases, the training data are imbalanced [15] and thus, produce biased models (model overfitting, Under-fitting). In anomaly data, the number of outliers is proportionately very low compared to normal data. Due to this rarity of anomaly data (data imbalance), multi-class classification approaches have not been effective in anomaly detection. Apart from the data imbalance, the anomalies do not have predefined or consistent patterns. The nature and the properties of anomaly data may vary drastically over time. Thus, the models trained using such data become invalid due to the unpredictable nature of anomaly data. To solve this issue, we analyzed the effectiveness one-class classification approach for anomaly detection. In the one-class classification approach we trained a supervised machine learning model using non-anomalous data so, the model can detect(classify) the non-anomalous data in the presence of anomalous data. In this research we used two one-class classifiers; OCSVM and Autoencoder a neural network-based method.

OCSVM is a natural extension of the one-class support vector machine that is supervised learning models that analyze data and recognize patterns, which are often used for both classification and regression tasks. The OCSVM algorithm (through the kernel) allocates input data instances in a high dimensional feature space and iteratively finds the largest margin hyperplane that best separates the training data from the source [16]. Kernel-based learning methods use implicit mapping of input data in a high-dimensional feature space defined by a kernel function [17]. Therefore, in OCSVM, the support vector model is focused to train data that has only one class, which is considered as the most discriminative class. In our research, the equivalent is the non-anomalous data. This infers the properties of normal cases and can predict the examples are unlike the typical examples from these properties. This is useful for identifying anomalies because of the lack of coaching examples identifies anomalies. For example, there are usually only a few cases of network interference, theft, or other anomalous behavior and thus, easier ways of identifying anomalies. Therefore, one-class models are effective in this situation.

Autoencoder is a neural network-based supervised machine learning algorithm. In general, Autoencoder has three layers; an input layer, an encoding layer, and a decoding layer. An example of Autoencoder shown in Figure 3. When considering the encoder and decoder, they are similar to zip and unzip functions for compression, learned from the dataset. For the best representation of inputs, the encoding layer is forced to learn and the neural network is trained to reconstruct its inputs for this purpose. In this research, Autoencoder is

trained to learn the normal behavior of the corresponding data instances and to be activated if anomalous conditions are measured [18]. If an Autoencoder can learn the correlations between the set of data features that describe the state of a cloud network dataset, then it can consequently notice changes in these correlations that indicate an abnormal state.

It is a common belief that neural network based models perform better than conventional machine learning methods. Numerous researches have shown the effectiveness of neural network models against traditional machine learning methods. On the other hand, neural network models efficiently learn from huge data sets. Fig. 2 shows the variation of model performance with the size of the data. It clearly shows that the neural network models perform far better with large datasets.

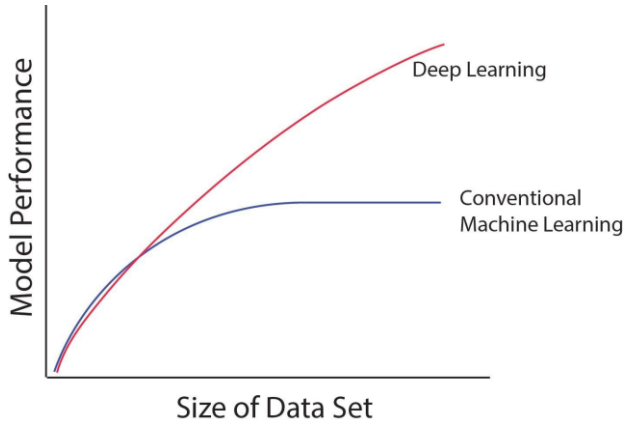

Fig. 2. Variation of model performance over the size of a data set

Further, one-class classifiers to detect anomalies in the cloud network have been selected in order to increase the performance of the study and to achieve accurate results. Two class classifiers could be also used for those tasks, but a trained pre-defined model for detecting anomalies is not possible for every scenario. While using one-class classifiers, a model for the normal flow and activates in the cloud network have been also developed. For this use case, the ultimate goal was to detect anomalies behavior in cloud network data by using a One-class support vector machine and Autoencoder. The discussion has been carried out on some of the existing state-of-art research related to the topic as well. Anomaly detection is the identification of various kinds of activities that are deviated from the normal data behavior. Typically, an anomaly is opened to many newer problems to every domain, and identifying these kinds of activities in cloud network data provided a reasonable solution for cloud users.

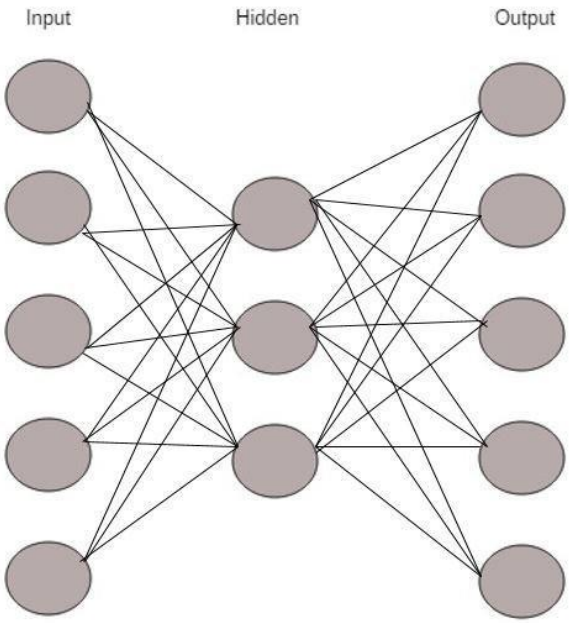

Fig. 3. Autoencoder 3-layer architecture

IV. EXPERIMENTAL ANALYSIS

A. Data

To test our methods, we used two benchmark data sets; YAHOO Synthetic and real time-series with labeled anomalies data set have been used in this research. Data received from the real yahoo servers was essentially a soft version. It provided a good tool for exploring algorithms without the complexity of real data. Yahoo data set contains four (A1, A2, A3, A4) benchmark data sets. The A4 Benchmark includes changepoint anomalies and it contains 8 data features that are being described below in Table I.

- Timestamps: the UNIX timestamp marks every hour

  (hourly sampled data)

  Value: time series value at relevant timestamp
- Anomaly: for an outlier value will be 1
- Changepoint: if the change point was there, the value will be 1
- Trend: the additive trend value for this timestamp
- Noise: the additive noise value for this timestamp
- Seasonality1: seasonality value for a period of twelve hours
- Seasonality2: calculated seasonality value for the daily period
- Seasonality3: calculated seasonality value for the weekly period

TABLE I. FEATURES OF YAHOO DATA SET WITH EXAMPLE DATA

Timestamp	Value	Anomaly	Changepoint	Trend	Noise	Seasonality1	Seasonality2	Seasonality3
1417590000	779.0698	0	0	484	4.135452	142.5	61.85775	86.57662
1417593600	930.3786	0	0	486	-1.03502	246.8172	119.5	79.09633
1417597200	1019.256	0	0	488	5.752064	285	168.9985	71.50542
1417600800	1020.809	0	0	490	13.19709	246.8172	206.9801	63.8145
1419188400	2492.884	0	0	2002	225.7357	235.2	-76.8693	106.8176
1419192000	2440.698	0	0	2004	63.51916	407.3783	-148.5	114.3
1419195600	2385.058	0	0	2006	-2.95404	470.4	-210.011	121.6225
1417683600	-102.321	1	0	0	2.255834	97.2	41.43646	-15.9078
1418997600	3001.277	1	0	1896	78.74763	-407.378	257.2095	-227.961

This data set contained 335,999 records and each record has 8 features that were generated by Yahoo servers. Also, it contained approximately 1800 anomalies data records for this activity. These statistics clearly indicate the class imbalance in the data. We have used approximately 230,000 normal data records to train our model (approximately 70% of the no anomalous data). For the test data set, we used approximately 68,000 records (approximately 30% of the non-anomalous data and approximately 1800 anomalous data).

In addition, we used the UNSW-NB15 network data set for further investigation of selected machine learning algorithms. This data set contains, 138,300 total data with approximately 93,000 clean data and approximately 45800 anomalies data records. To train the model around approximately 65,100 normal data records have been used (approximately 70% of the non-anomalous data). For the test data set, we used approximately 73,200 records (approximately 30% of the no anomalous data and approximately 45800 anomalous data).

B. Experiments

The data sets were pre-processed to clean and to add labels (True for non-anomaly data and false for anomaly data). The pre-processed non-anomaly data set was split randomly into two subsets, one with 70% instances for training and the remaining 30% was merged with anomaly data set for the testing purpose.

a) Experiment 1: One Class Support Vector Machine for anomaly detection

First, we tested the effectiveness of OCSVM for anomaly detection. Here, we used R kernlab [19] library which includes OCSVM implementation in R. The results are shown in Table II.

b) Experiment 2: Autoencoder for anomaly detection

In our second experiment, we tested the effectiveness of Autoencoder for anomaly detection. Here, we used R h20 package which includes Autoencoder [19] implementation. The results are shown in Table II.

C. Experimental Results

In Table II, we compare the anomaly detection performance of one class classifier used in our experiments. It shows that 79.17% overall accuracy for a one-class support vector machine on yahoo cloud network data set and the 60.89% accuracy on UNSW-NB15 network data. In contrast, Autoencoder shows 96.02% accuracy on yahoo cloud network data set and 99.54% accuracy on the UNSW-NB15 network data set.

TABLE II. STATISTICS OF DATA

Classification method	Accuracy(%)	
	UNSW-NB15 Data	***YAHOO Anomaly Data***
One-Class Support Vector Machine	60.89	79.05
Autoencoder	99.10	96.02

The above results show that the neural network-based methods perform better than the kernel-based methods in anomaly detection in cloud network data sets we used in our experiments.

V. DISCUSSION

In this section, we discuss limitations, issues and future direction of this research. In this research, we identified several limitations. Among them, we noticed that both models failed to recognize the non-anomalous outliers show in Fig. 4 and Fig. 5. The Autoencoder struggles in the tail of both data sets. The reconstruction error plots show that the error count accelerates upwards (in the YAHOO data set after approximately 230,000 records and the UNSU-NB15 data set is approximately after 65,000 records). This observation tells that Autoencoder recognizes some normal or innocent data as anomalies data. This information can be used to define the decision boundary for anomaly detection assuming that the last data instances are outliers with respect to the rest of instances in the clean data set.

A. Limitations

a) Class Imbalance Problem: The presence of a roughly equal number of instances in each class paves way for most machine learning algorithms to work their best. However, class imbalance is a common problem in anomaly detection data sets. As an example, within our data sets the sum of anomalies data is comparatively very small. As a solution, some research has used alternative performance metrics (true positives, true negatives, false positives, false negatives) instead of the standard precision of calculating the number of errors to compare solutions. There are many ways to solve this problem, owing to its universality. In general, these can be classified into two main categories: 1) based on sampling; 2) based on cost functions. Basic sampling can be divided into three categories: a) oversampling b) sampling c) mixed oversampling and subsampling. In our future works, we will investigate the effectiveness of those sampling techniques.

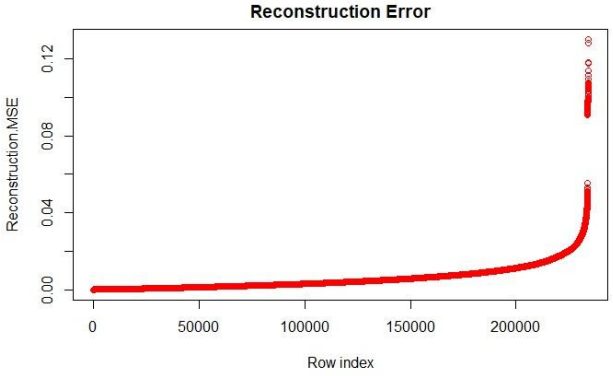

Fig. 4. Reconstruction error for the yahoo data set

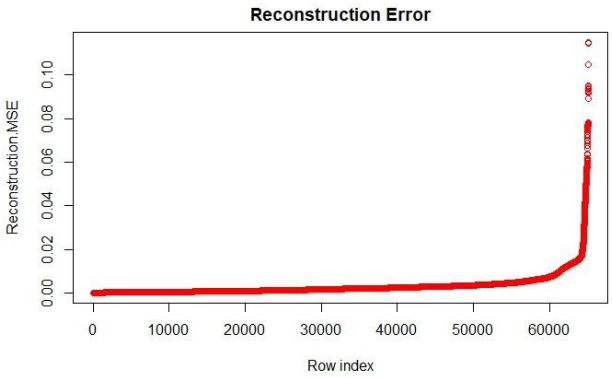

Fig. 5. Reconstruction error for the UNSW-NB15 data set

B. *Analysis of Feature Importance*

Further, we carried out an investigation about the variable importance of the data for Autoencoder. Variable importance provided the statistical significance of the variables in the data set that was used to generate the model [20]. This method was very helpful when we use one-class classification methods to make predictions according to our data if they contribute or do nothing with our generated model. The variable importance of the data sets is shown in Fig. 6 and Fig. 7. In our future works, we will analyze the trade-off between accuracy and reconstruction error by selecting different feature sets according to their importance.

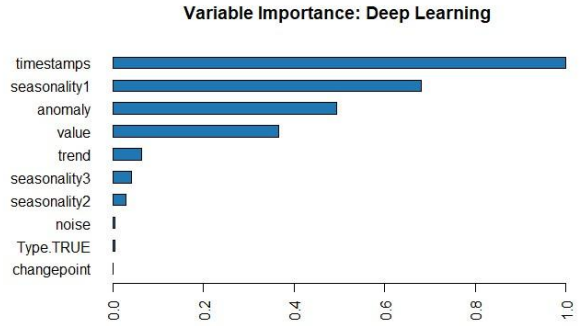

Fig. 6. Variable importance of yahoo data set

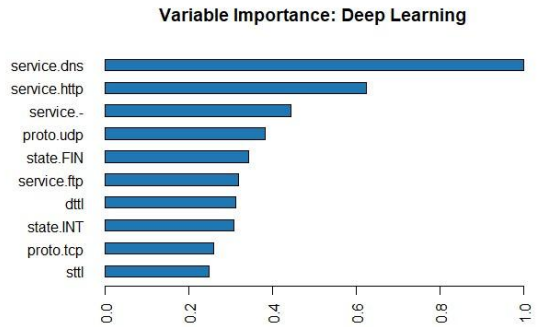

Fig. 7. Variable importance of UNSW-NB15 data set

C. *Future Work*

Apart from that, it is worth investigating the hyperparameters of the classifiers we used in our experiments. Fitting a neural network model takes considerable time. To find the best model, one can carry out a grid search for model parameters. However, this takes lots of time and need high computational power. Thus, we had to use pre-determined or default parameter values of the models.

VI. CONCLUSION

In this research, we investigated the anomaly detection in cloud network data using supervised machine earning methods. Instead of multi-class classifiers, we choose two one-class classifiers, One-Class Support Vector Machine(OCSVM) and Autoencoder algorithms which were trained to detect outliers. We tested both algorithms on two benchmark data sets; yahoo dataset and UNSW-NB15 dataset. To the best of our knowledge, this is the first study that used YAHOO synthetic data for anomaly detection. The experimental results show that the neural network-based Autoencoder performs better in contrast to the kernel-based OCSVM algorithm in anomaly detection. Also, a one-class classification approach provides a better solution for class imbalance problems in anomaly detection.

REFERENCES

[1] G. Gruman and E. Knorr, “what cloud computing really means,” *InfoWorld*, April 2008.

[2] D. Zissis and D. Lekkas, “Addressing cloud computing security issues,” *Future Generation Computer Systems*, vol. 28 pp. 583 – 592, 2012.

[3] M. Gander, M. Felderer, B. Katt, A. Tolbaru, R. Breu, and A. Moschitti, “Anomaly detection in the cloud: Detecting security incidents via machine learning,” vol. 379, pp. 103–116, Jan. 2013.

[4] M. Thill, W. Konen and T. Bäck, "Online anomaly detection on the webscope S5 dataset: A comparative study," *2017 Evolving and Adaptive Intelligent Systems (EAIS)*, Ljubljana, 2017, pp. 1-8.

[5] N. Moustafa and J. Slay, “The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set,” Jan. 2016.

[6] P. Srivastava and R. Khan, “A review paper on cloud computing,” *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 8, no. 6, 2018.

[7] P. Schoo, V. Fusenig, V. Souza, M. Melo, P. Murray, H. Debar, H. Medhioub, and D. Zeghlache, “Challenges for cloud networking security,” vol. 68, pp. 298–313, Jan. 2010.

[8] S. Aldossary and W. Allen, “Data security, privacy, availability and integrity in cloud computing: Issues and current solutions,” *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 4, 2016.

[9] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” *ACM Computing Surveys*, vol. 41, July 2009.

[10] J. Pereira, “Unsupervised Anomaly Detection in Time Series Data using Deep Learning,” Master’s thesis, Instituto Superior Tecnico, Lisbon, 2018.

[11] B. Scholkopf, R. Williamson, A. Smola, J. Shawe, Taylor, and J. Platt, “Support vector method for novelty detection,” vol. 12, pp. 582–588, Nov. 1999.

[12] Y. Chen, J. Qian, and V. Saligrama, “A new one-class svm for anomaly detection,” pp. 3567–3571, Oct. 2013.

[13] R. Chalapathy, S. Chawla, "Deep Learning for Anomaly Detection: A Survey", 2019.

[14] S. Omar, M. Ngadi, H. Jebur, and S. Benqdara, “Machine learning techniques for anomaly detection: An overview*,” International Journal of Computer Applications*, vol. 79, Oct. 2013.

[15] X. Guo, Y. Yin, C. Dong, G. Yang, and G. Zhou, “On the class imbalance problem,” vol. 4, 2008.

[16] L. Maglaras and J. Jiang, “A novel intrusion detection method based on ocsvm and k-means recursive clustering,” vol. 2, pp. 1–10, Jan. 2015.

[17] A. Karatzoglou, A. Smola, and K. Hornik, “kernlab - an s4 package for kernel methods in r,” vol. 69, pp. 721–729, Nov. 2004.

[18] A. Borghesi, A. Bartolini, M. Lombardi, M. Milano, and L. Benini, “Anomaly detection using autoencoders in high performance computing systems,” 2018.

[19] B. Mitra, McCausland, H. Kalutarage, “Modelling iot anomoly detection,” vol. 60, pp. 44–45, 2018.

[20] P. Wei, L. Zhenzhou, and J. Song, “Variable importance analysis: A comprehensive review,” vol. 142, June 2015