TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 02/11 pp146–153
DOI: 10.26599/TST.2019.9010051
Volume 26, Number 2, April 2021

A Hybrid Unsupervised Clustering-Based Anomaly Detection Method

Guo Pu, Lijuan Wang*, Jun Shen, and Fang Dong

Abstract: In recent years, machine learning-based cyber intrusion detection methods have gained increasing popularity. The number and complexity of new attacks continue to rise; therefore, effective and intelligent solutions are necessary. Unsupervised machine learning techniques are particularly appealing to intrusion detection systems since they can detect known and unknown types of attacks as well as zero-day attacks. In the current paper, we present an unsupervised anomaly detection method, which combines Sub-Space Clustering (SSC) and One Class Support Vector Machine (OCSVM) to detect attacks without any prior knowledge. The proposed approach is evaluated using the well-known NSL-KDD dataset. The experimental results demonstrate that our method performs better than some of the existing techniques.

Key words: unsupervised learning; clustering; intrusion detection; feature selection

1 Introduction

Cyber security refers to the technologies, processes, and practices designed to protect internet-connected systems, including networks, computers, programs, and data from attacks, destruction or unauthorized access[1,2]. Intrusion Detection System (IDS) is one part of the cyber security systems. IDS is utilized to discover, determine, and identify intrusions by analyzing data collected through network devices[3].

According to the detection mechanism, IDS can be classified as misuse (also called signature-based) detection and anomaly (also called behavior-based) detection[4]. Misuse detection approaches are designed to detect attacks by using a database of predefined attack patterns. They are highly effective to detect known attacks and preferred for the low false positive rate. Nonetheless, they are unable to defend the system against unknown attacks, because such attacks do not exist in the predefined pattern lists. As network attacks continue to increase in the frequency and diversity, maintaining an updated database is time-consuming and unfeasible. Moreover, misuse detection approaches cannot detect zero-day attacks. On the other hand, anomaly detection approaches use the normal system activity to build normal-operation profiles, identifying anomalies as behaviors that deviate from the normal ones. Such methods are especially appealing because they are able to potentially detect all known and unknown types of attacks as well as zero-day attacks. However, the main disadvantage of anomaly detection approaches is that they require a tuning stage and suffer from high false positive rates.

To improve the detection rate and minimize the false positive rates, many studies describe machine learning techniques for cyber intrusion detection[2,4,5]. The current paper focuses primarily on the anomaly-based IDS, which can be generally divided into three main categories according to the machine learning methods used: supervised (classification), unsupervised (clustering and outlier-based detection), and semi-supervised[4]. In supervised IDS, a model is trained to learn from completely labeled data[6,7]. However, most

• Guo Pu and Lijuan Wang are with the School of Cyber Engineering, Xidian University, Xi'an 710126, China. E-mail: tr1ckyb0uncie@gmail.com; ljwang@xidian.edu.cn.
• Jun Shen is with the School of Computing and Information Technology, University of Wollongong, Wollongong, NSW 2522, Australia. E-mail: jshen@uow.edu.au.
• Fang Dong is with the School of Computer Science and Engineering, Southeast University, Nanjing 211189, China. E-mail: fdong@seu.edu.cn.
* To whom correspondence should be addressed.
Manuscript received: 2019-07-02; revised: 2019-09-02; accepted: 2019-09-09

of the classification models have a similar drawback, namely, in comparison to the misuse-based ones, they fail to detect unknown attacks, and they need to be periodically trained to preserve high detection rates. This is not practical as it is difficult to obtain labeled data. Semi-supervised IDS creates a model by using a small amount of labeled data with a large amount of unlabeled data. On the other hand, in unsupervised IDS, clustering techniques are utilized for finding anomalies in unlabeled data. The aim of clustering algorithms is to separate the given unlabeled data into clusters that achieve high inner similarity and outer dissimilarity, without relying on signatures, explicit description of attack classes, or labeled data for training. Besides the ability to detect known and unknown attacks, unsupervised intrusion detection methods do not require labeled data for the training process. They can answer the attribution and correlation questions by extracting features from different sources[4].

Thus, as the amount and complexity of new attacks increase, effective and intelligent solutions are crucial. In the current paper, we describe an unsupervised clustering-based anomaly detection method, without any prior knowledge. The key contributions of this paper are threefold: firstly, the process of unsupervised anomaly detection based on a clustering technique is described; secondly, a novel anomaly detection method based on Sub-Space Clustering (SSC) and One Class Support Vector Machine (OCSVM) is presented; thirdly, experiments are conducted to compare the proposed method with three other approaches by using different evaluation metrics.

The remainder of the paper is organized as follows. Section 2 introduces a brief background to the state-of-the-art developments in the unsupervised intrusion detection field. Section 3 describes the proposed method and the employed performance metrics. Furthermore, Section 4 evaluates the developed method using the well-known NSL-KDD attacks dataset, and compares its performance with three known methods. Finally, Section 5 concludes the study and sets proposals for the future work.

2 Related Work

Over the last two decades, many machine learning and data mining methods, such as ant colony optimization[8], artificial neural networks[9], particle swarm optimization[10], evolutionary computation[11], and Support Vector Machine (SVM)[12], have been proposed for cyber intrusion detection. We primarily focused on unsupervised approaches that have been proposed for the intrusion detection in the literature.

In Ref. [5], the authors introduced an unsupervised anomaly detection method by combining SSC, Evidence Accumulation (EA), and Density-Based Spatial Clustering of Applications with Noise (DBSCAN) clustering. The SSC[13] method was used to produce multiple data partitions by dividing the original feature space X into N different sub-spaces. In addition, the DBSCAN[14] clustering method was applied to each partition. The EA clustering was employed to rank the degree of abnormality. Subsequently, the well-known KDD99 dataset was utilized to evaluate the proposed method. The experimental results demonstrated that this approach outperformed the traditional methods.

Moreover, Amoli et al.[15] proposed an unsupervised IDS for high-speed networks, which detected zero-day attacks via two separate engines. The first engine used the DBSCAN clustering to detect attacks and the second one found the botnet under different protocols. To evaluate the proposed model, two publicly available datasets were used. The presented model was then evaluated and compared with the K-means and DBSCAN-based outlier detection approaches. Furthermore, the study in Ref. [16] applied the K-means and DBSCAN clustering algorithms to detect anomalous behaviors in unlabeled network and system log data.

A literature survey of machine learning and data mining methods for cyber intrusion detection was also carried out in Ref. [2]. Buczak and Guven[2] provided example studies, described various methods, and explained the importance of the datasets for training and testing the IDS. On the other hand, Nisioti et al.[4] provided a survey of unsupervised methods for anomaly-based IDS, as well as presented and compared feature selection methods for intrusion detection. They highlighted the importance of feature selection techniques for decreasing the computational time and complexity. Lastly, a DBSCAN clustering method to group normal packets versus anomaly was presented in Ref. [17]. Blowers and Williams[17] also summarized the application of machine learning methods to cyber operations.

3 Clustering-Based Unsupervised Anomaly Detection

As an unsupervised approach, a clustering technique is

used to group data according to a similarity measure. The goal of clustering is to attain high intra-cluster similarity (i.e., data within a cluster are similar) and low inter-cluster similarity (i.e., data from different clusters are dissimilar). There are several approaches for clustering the input data, two of which are the well-known K-means and DBSCAN methods.

K-means is a partitioned-based clustering algorithm that produces sphere-like clusters. It is relatively efficient and has been used for medium and large sized dataset. This approach attempts to minimize the intra-cluster distances and maximize the inter-cluster ones. However, its drawback is that the number of clusters K must be pre-specified, which is not a simple or intuitive task. The correct choice of K is often ambiguous, because it is highly dependent on the shape and scale of the distribution of points in a dataset. Moreover, since the initial centroids are often selected randomly, the algorithm is easily trapped in the local optimum.

DBSCAN is a density-based clustering algorithm that produces arbitrarily-shaped clusters. Density is defined as the number of points within a specified radius. It is particularly useful when dealing with spatial clusters or when there is noise in the dataset. It works with two parameters: radius and minimum points. DBSCAN is robust to outliers, and it can even find a cluster completely surrounded by a different cluster. Beyond that, DBSCAN is remarkably useful in many real world problems, as it does not require specification of the number of clusters, such as K in the K-means algorithm.

3.1 SSC-OCSVM algorithm

The current clustering techniques typically lack robustness. In other words, the results of clustering algorithms depend on the algorithm itself, and are also affected by the initialization and parameters of the underlying algorithm. Inspired by the works in Refs. [5, 13, 18], to avoid such a limitation, the current paper proposes an unsupervised anomaly intrusion detection algorithm called SSC-OCSVM, which combines SSC and OCSVM to detect attacks.

SSC is an extension of the traditional clustering techniques. It produces clusters from different small sub-spaces of the original dataset $X \in \mathbf{R}^{n\times m}$. Let n be the number of records and m be the number of attributes or features. The N sub-spaces $X_i \in X$ ($i \in \{1, 2, \ldots, N\}$) is produced by selecting q features from the m attributes. The number of sub-spaces N corresponds to $\binom{m}{q}$. To set the value of q, the downward closure property is taken, which implies that if a collection of samples is in X, it is also in low-dimensional sub-spaces of X. Using small values of q is more efficient and faster[5]. Besides, DBSCAN gives improved results in low-dimensional spaces[19]. Consequently, we set $q = 2$ for SSC, which gives $N = m \times (m-1)/2$.

SVM is a supervised learning model that analyzes data and recognizes patterns. OCSVM is an extension of the SVM method and is especially suitable for unlabeled data[18]. In OCSVM, the support vector model is trained on data that has only one class, which is the normal class. It maps the data into the feature space corresponding to the kernel, and separates them from the origin with maximum margin[18].

The methodology of the presented SSC-OCSVM algorithm has the following steps.

(1) **Initialization**. Set a null dissimilarity vector $\boldsymbol{D}$, and divide the feature space X into N different sub-spaces $X_i \in X (i \in \{1, 2, \ldots, N\})$.

(2) **Clustering and learning**. Apply OCSVM to each sub-space X_i and produce partitions P_i.

(3) **Evidence accumulation**. Update dissimilarity vector $\boldsymbol{D}$ based on each partition P_i. Vector $\boldsymbol{D}$ accumulates the distance between the different outliers found in sub-space X_i. This operation is inspired by the idea of the EA clustering[20].

(4) **Anomaly detection**. Rank vector $\boldsymbol{D}$ and obtain a ranked vector $\boldsymbol{D}_{\text{rank}}$. In $\boldsymbol{D}_{\text{rank}}$, if the dissimilarity value is greater than a predefined threshold value, then the corresponding sample is considered as an anomaly.

The flowchart of the proposed SSC-OCSVM algorithm is illustrated in Fig. 1.

3.2 Measure

Different metrics have been used to measure how well a machine learning method performs in detecting attacks. The most common measures to evaluate the detection ability are as follows.

• Confusion matrix, also known as error matrix, compares the actual results with the predicted ones. It is mainly utilized in supervised learning to evaluate the prediction accuracy of a classifier. A binary confusion matrix is shown in Table 1, each row corresponds to the actual results, while each column corresponds to the predicted ones. True Positive (TP) refers to the actual class Y that was correctly classified as class Y, False Positive (FP) refers to the actual class $\hat{Y}$ that was incorrectly labeled as class Y, False Negative (FN) refers to the actual class Y that was incorrectly marked as class

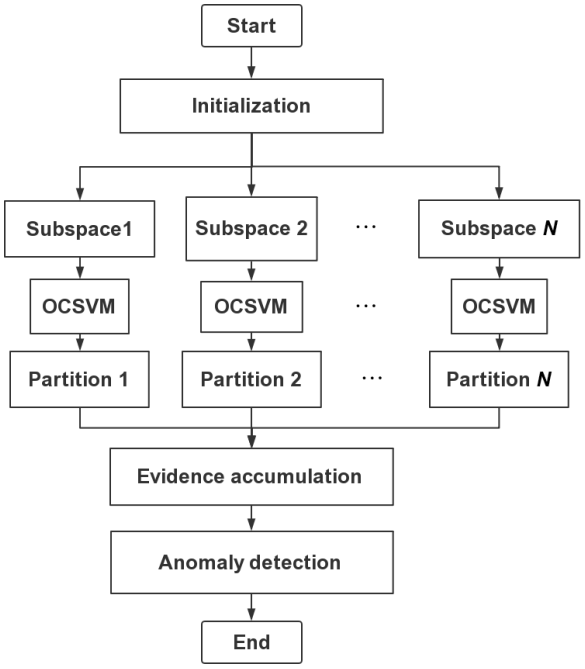

Fig. 1 Flowchart of the SSC-OCSVM algorithm.

Table 1 Binary confusion matrix.

Actual class	Predicted class	
	Y	$\hat{Y}$
Y	TP	FN
$\hat{Y}$	FP	TN

$\hat{Y}$, and true Negative (TN) refers to the actual class $\hat{Y}$ that was correctly classified as class $\hat{Y}$.

• Sensitivity or recall or TP Rate (TPR) or probability of detection or Detection Rate (DR), is the proportion of positive samples that are correctly classified as such, i.e., $\text{TPR} = \text{TP}/(\text{TP} + \text{FN})$.

• False Alarm Rate (FAR) or FP rate represents the proportion of samples that are incorrectly identified as anomalies, i.e., $\text{FAR} = \text{FP}/(\text{FP} + \text{TN})$.

• The Receiver Operating Characteristic (ROC) curve is a method of visualizing the DR against the FAR for different parameter settings. It illustrates the relative trade-offs between DR (on the y-axis) and FAR (on the x-axis)[21].

4 Experimental Evaluation Using the NSL-KDD Dataset

4.1 Preprocessing of dataset

The KDD99[22] is the most widely used network attacks dataset utilized in academia. It includes a wide variety of intrusions simulated in a military network environment. However, the significant issues in the original KDD99 dataset have been shown to highly affect the performance of anomaly detection methods. Therefore, NSL-KDD[23] was proposed to eliminate some of the inherent problems of the KDD99 dataset[24]. Compared with the original KDD99 dataset, the number of records in the NSL-KDD train and test sets are reasonable, and there are no redundant records. Consequently, evaluation results of different research studies will be consistent and comparable. Thus, we evaluate our proposed algorithm using the NSL-KDD dataset.

Each record in the NSL-KDD dataset is described by 41 features, as shown in Table 2. Among these features, protocol_type, service, and flag are non-numeric and should be converted into numeric features. Here, we adopted a one-hot encoding method to transform these three features into numerical ones. Following transforming, the total number of features increased from 41 to 132, and there was a lot of redundancy. Hence, a feature selection process was necessary.

Feature selection is one of the most important steps of the unsupervised anomaly detection[4]. It refers to the process of selecting a subset of the available features that are the most relevant and non-redundant. The quality of selected features can not only enhance the accuracy of the method, but also decrease FAR and computational time. In other words, it directly affects the detection rate and the performance of a machine learning method. In the current paper, we adopted the F-test to select features. Meanwhile, we normalized each feature by removing the mean and scaling to unit variance.

4.2 Splitting of dataset

As mentioned in Ref. [4], when clustering technique is used to detect network attacks, two assumptions exist.

Table 2 Total attributes in the NSL-KDD dataset.

41 features and one label		
duration	su_attempted	same_srv_rate
protocol_type	num_root	diff_srv_rate
service	num_file_creations	srv_diff_host_rate
flag	num_shells	dst_host_count
src_bytes	num_access_files	dst_host_srv_count
dst_bytes	num_outbound_cmds	dst_host_same_srv_rate
land	is_host_login	dst_host_diff_srv_rate
wrong_fragment	is_guest_login	dst_host_same_src_port_rate
urgent	count	dst_host_srv_diff_host_rate
hot	srv_count	dst_host_serror_rate
num_failed_logins	serror_rate	dst_host_srv_serror_rate
logged_in	srv_serror_rate	dst_host_rerror_rate
num_compromised	rerror_rate	dst_host_srv_serror_rate
root_shell	srv_rerror_rate	label

Firstly, the number of normal flows vastly outnumbers the anomalies. Secondly, there is a qualitative difference between the anomalies and normal instances. Our experiment follows the above assumptions in subset splitting. All of the attacks in the NSL-KDD training and testing set can be classified into four types: probe or scan, Denial of Service (DoS), User to Root (U2R), and Remote to Local (R2L). To evaluate the proposed algorithm in detecting these four types of attacks as well as hybrid attacks, respectively, the NSL-KDD dataset was split into four single attack subsets and a mixed type subset. The single attack subset consisted of normal flows and one specific type of attack. The mixed subset consisted of normal flows and a mixture of all four types of attacks. The splitting of dataset is demonstrated in Table 3. The training subset is only used for parameter tuning. After all the parameters and threshold values were tuned properly and fixed for the training subset, the performance of the proposed algorithm on the test subset was evaluated.

4.3 Analysis of the results

The entire experimental process is shown in Fig. 2. After processing and splitting the NSL-KDD dataset, we evaluated our proposed algorithm and compared its performance with the ones of K-means, DBSCAN, and the SSC-EA methods as introduced in the study[5].

K-means and DBSCAN develop outliers and many clusters of different sizes. In terms of the predefined threshold, the larger clusters will be identified as normal flow, and the remaining ones are considered as potential anomalies. We obtained an ROC curve of these two algorithms by setting different threshold values.

Our proposed SSC-OCSVM algorithm will eventually get a sorted dissimilarity vector $\boldsymbol{D}$. The samples are considered as potential anomalies if their corresponding dissimilarity values are greater than the detection threshold values. Analogously, we obtained the ROC curve of the SSC-OCSVM algorithm.

The ROC curve of K-means, DBSCAN, SSC-EA, and SSC-OCSVM on the Test_Probe, Test_DoS, Test_R2L, Test_U2R, and Test_mixed subset are illustrated in Figs. 3 – 7, respectively. From Figs. 3 – 7, it can be noticed that the ROC curve of our proposed algorithm SSC-OCSVM has the largest area under curve, both on the single attack subset and on the mixed one. In other words, the proposed algorithm is able to detect a large fraction of attacks with very low FARs. Meanwhile, for the two low-frequency attack classes, U2R and R2L, it is difficult to detect them as they resemble normal traffic[4]. Moreover, our proposed algorithm achieved particularly high DRs and low FARs in these two attack classes, which is better than the other three methods.

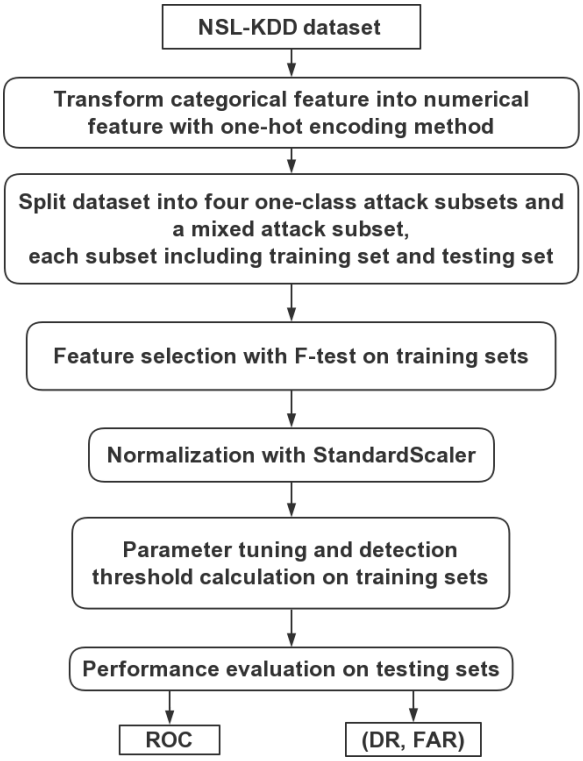

Fig. 2 Entire experimental process.

The DR and FAR of K-means, DBSCAN, SSC-EA, and SSC-OCSVM of all the subsets are given in Table 4 in value pairs of (DR, FAR). Using Table 4, it was validated that normally higher DR results in higher FAR

Table 3 Splitting of the NSL-KDD dataset.

Attack type	Number of subsets									
	Train_Probe	Test_Probe	Train_DoS	Test_DoS	Train_R2L	Test_R2L	Train_U2R	Test_U2R	Train_mixed	Test_mixed
Normal	2000	2000	2000	2000	2000	2000	2000	2000	9000	9000
Probe	100	100	–	–	–	–	–	–	100	100
DoS	–	–	100	100	–	–	–	–	100	100
R2L	–	–	–	–	50	50	–	–	50	50
U2R	–	–	–	–	–	–	50	50	50	50
Total	2100	2100	2100	2100	2050	2050	2050	2050	9300	9300

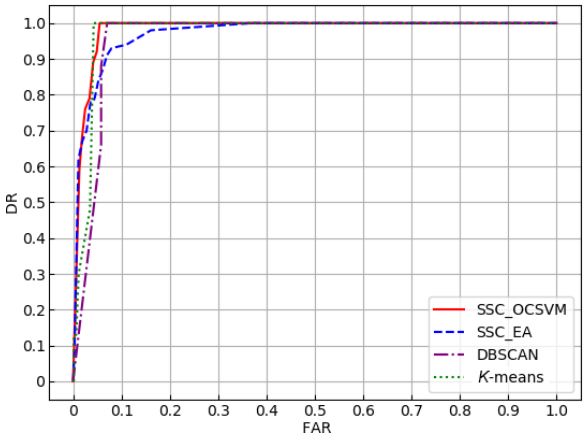

Fig. 3 DR vs FAR on Test_Probe subset.

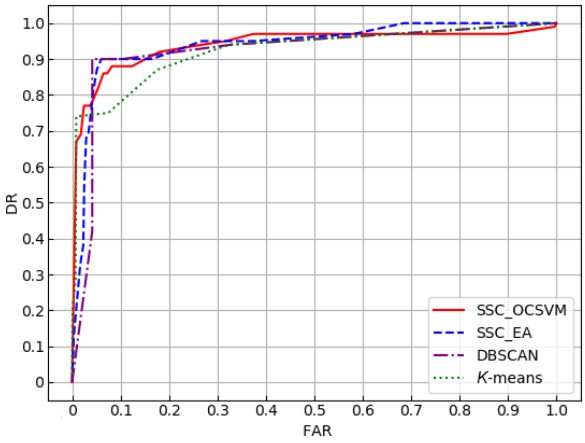

Fig. 4 DR vs FAR on Test_DoS subset.

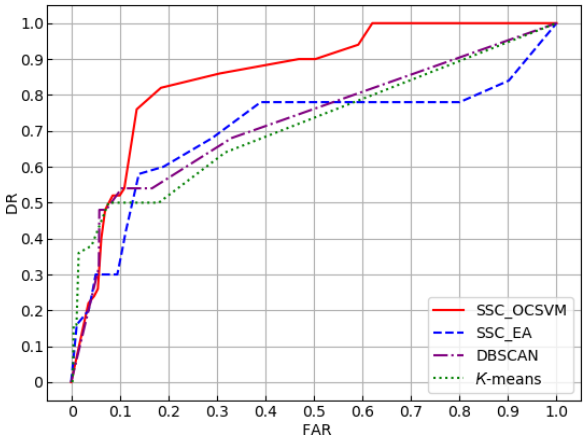

Fig. 5 DR vs FAR on Test_R2L subset.

and lower DR results in lower FAR. As demonstrated in Table 4, the performance of our proposed algorithm is relatively better than that of the other three approaches.

The computation time of SSC-OCSVM, SSC-EA, DBSCAN, and K-means on the Test_mixed subset are 238.88 s, 1060.76 s, 8.15 s, and 0.69 s, respectively. The reason for the computation time of our proposed

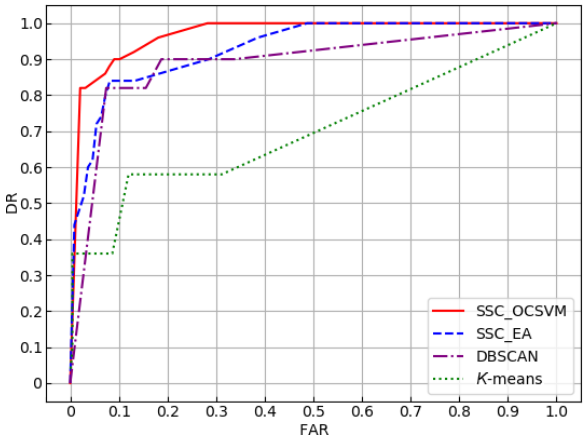

Fig. 6 DR vs FAR on Test_U2R subset.

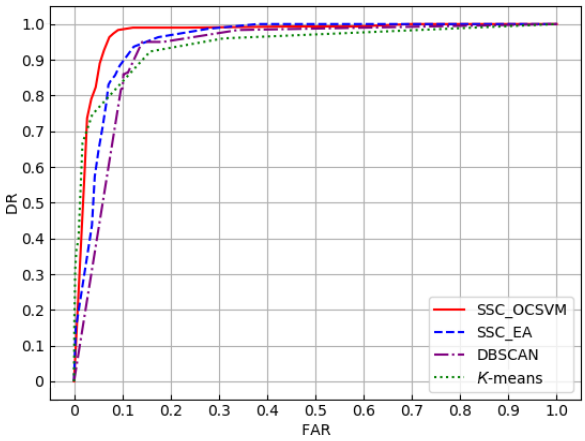

Fig. 7 DR vs FAR on Test_mixed subset.

algorithm being higher than K-means and DBSCAN is the sequential execution of each sub-space. It is noteworthy that in our algorithm, each sub-space was independent and sub-space could actually be executed in parallel.

5 Conclusion and Future Work

In this paper, we presented an unsupervised anomaly detection algorithm SSC-OCSVM, which combined sub-space clustering and one class support vector machine to detect attacks without any prior knowledge. We evaluated the proposed algorithm utilizing the well-known public NSL-KDD network attacks dataset. In addition, we compared its performance with three other clustering algorithms for unsupervised detection available in the literature. The experimental results demonstrate that our algorithm is superior. Future work should focus predominantly on developing an effective feature selection method. In addition, since each sub-space can be clustered independently, the approach can be adapted for parallel computing. We will therefore

Table 4 Comparison of each algorithm in terms of DR and FAR.

Subset	SSC-OCSVM	SSC-EA	DBSCAN	K-means
Train_Probe	(0.95, 0.0680)	(0.95, **0.0635**)	(**0.99**, 0.0810)	(0.97, 0.0910)
Test_Probe	(0.99, 0.0660)	(0.93, 0.0795)	(0.85, **0.0285**)	(**1.00**, 0.1095)
Train_DoS	(0.93, 0.0690)	(0.90, 0.0810)	(**0.98**, 0.0790)	(0.97, **0.0470**)
Test_DoS	(0.85, 0.0780)	(0.90, **0.0600**)	(0.87, 0.0950)	(**0.93**, 0.2130)
Train_R2L	(**0.68**, 0.0960)	(0.18, 0.0980)	(0.38, 0.0760)	(0.38, **0.0725**)
Test_R2L	(**0.52**, 0.0895)	(0.30, 0.0950)	(0.40, **0.0335**)	(0.42, 0.0635)
Train_U2R	(**0.84**, 0.0925)	(0.70, 0.0955)	(**0.84**, 0.0690)	(0.82, **0.0415**)
Test_U2R	(**0.90**, 0.0905)	(0.84, 0.0920)	(0.52, 0.0355)	(0.52, **0.0145**)
Train_mix	(0.91, 0.0900)	(0.87, 0.0950)	(**1.00**, 0.0900)	(0.93, **0.0800**)
Test_mix	(**0.89**, 0.0800)	(0.84, 0.0980)	(0.80, 0.0840)	(0.77, **0.0500**)

implement the parallelization of our algorithm in our future research.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (Nos. 61702398 and 61872079), China 111 Project (No. B16037), and University Global Partnership Network (UGPN) Project of the University of Wollongong 2018 – 2019.

References

[1] M. Rouse, Cyber security, https://searchsecurity.techtarget.com/definition/cybersecurity, 2016.

[2] A. L. Buczak and E. Guven, A survey of data mining and machine learning methods for cyber security intrusion detection, *IEEE Communications Surveys Tutorials,* vol. 18, no. 2, pp. 1153–1176, 2016.

[3] A. Mukkamala, A. Sung, and A. Abraham, Cyber security challenges: Designing efficient intrusion detection systems and antivirus tools, in *Proc. of Enhancing Computer Security with Smart Technology*, New York, NY, USA, 2005, pp. 125–163.

[4] A. Nisioti, A. Mylonas, P. D. Yoo, and V. Katos, From intrusion detection to attacker attribution: A comprehensive survey of unsupervised methods, *IEEE Communications Surveys Tutorials,* vol. 20, no. 4, pp. 3369–3388, 2018.

[5] P. Casas, J. Mazel, and P. Owezarski, Unsupervised network intrusion detection systems: Detecting the unknown without knowledge, *Computer Communications,* vol. 35, no. 7, pp. 772–783, 2012.

[6] I. Kang, M. K. Jeong, and D. Kong, A differentiated one-class classification method with applications to intrusion detection, *Expert Syst. Appl.,* vol. 39, no. 4, pp. 3899–3905, 2012.

[7] F. Kuang, W. Xu, and S. Zhang, A novel hybrid KPCA and SVM with GA model for intrusion detection, *Applied Soft Computing*, vol. 18, pp. 178–184, 2014.

[8] L. Wang and J. Shen, A systematic review of bio-inspired service concretization, *IEEE Transactions on Services Computing*, vol. 10, no. 4, pp. 493–505, 2017.

[9] L. Wang, Q. Zhou, T. Jin, and H. Zhao, Feed-back neural networks with discrete weights, *Neural Computing and Application*, vol. 22, no. 6, pp. 1063–1069, 2013.

[10] L. Wang and J. Shen, Data-intensive service provision based on particle swarm optimization, *International Journal of Computational Intelligence Systems*, vol. 11, pp. 330–339, 2018.

[11] M. Sadiq and A. Khan, Rule-based network intrusion detection using genetic algorithms, *International Journal of Computer Applications*, vol. 18, no. 8, pp. 26–29, 2011.

[12] C. Wagner, J. François, R. State, and T. Engel, Machine learning approach for IP-flow record anomaly detection, *Lecture Notes in Computer Science*, vol. 6640, pp. 28–39, 2011.

[13] L. Parsons, E. Haque, and H. Liu, Subspace clustering for high dimensional data: A review, *SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 90–105, 2004.

[14] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining*, Portland, OR, USA, 1996, pp. 226–231.

[15] P. V. Amoli, T. Hamalainen, G. David, M. Zolotukhin, and M. Mirzamohammad, Unsupervised network intrusion detection systems for zero-day fast-spreading attacks and botnets, *International Journal of Digital Content Technology and Its Applications*, vol. 10, no. 2, pp. 1–13, 2016.

[16] A. Bohara, U. Thakore, and W. H. Sanders, Intrusion detection in enterprise systems by combining and clustering diverse monitor data, in *Proceedings of the Symposium and Bootcamp on the Science of Security*, Pittsburgh, PA, USA, 2016, pp. 7–16.

[17] M. Blowers and J. Williams, Machine learning applied to cyber operations, in *Network Science and Cybersecurity*, Advances in Information Security, vol. 55, pp. 155–175, 2014.

[18] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, Estimating the support of a high-dimensional distribution, *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[19] A. K. Jain, Data clustering: 50 years beyond K-means, *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, 2010.

[20] A. L. N. Fred and A. K. Jain, Combining multiple clusterings using evidence accumulation, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 27, no. 6, pp. 835–850, 2005.
[21] T. Fawcett, An introduction to ROC analysis, *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.
[22] KDD Cup 1999 Data, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, 1999.
[23] NSL-KDD Dataset, https://www.unb.ca/cic/datasets/nsl.html, 2009.
[24] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, A detailed analysis of the KDD CUP 99 data set, in *Proc. of 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, Ottawa, Canada, 2009, pp. 1–6.

Guo Pu received the bachelor degree from Xidian University, Xi'an, China in 2019. He is a master student in Peking University. His main research interest is computer vision.

Jun Shen received the PhD degree from Southeast University, China in 2001. He is an associate professor at the University of Wollongong, Australia. He has published more than 100 papers in journals and conferences in computer science and information system areas. His research interests include web services and e-learning systems, as well as big data. He is a senior member of IEEE, ACM, and ACS. He is a member of IS Curriculum Review Task Force sponsored by ACM/AIS.

Lijuan Wang received the PhD degree from the University of Wollongong, Australia in 2014. She is currently a lecturer at Xidian University, China. She has published 19 papers in journals and conferences in computer science. Her research interests include neural network, computing intelligence, and data security.

Fang Dong received the BS and MS degrees from Nanjing University of Science and Technology, Nanjing, China in 2004 and 2006, respectively, and the PhD degree from Southeast University in 2011. He is currently a professor in Southeast University, China. His current research interests include cloud computing, edge computing, and big data processing.