International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 13, Number 11 (2020), pp. 3987-3996
 https://dx.doi.org/10.37624/IJERT/13.11.2020.3987-3996

GAN based Augmentation for Improving Anomaly Detection Accuracy in Host-based Intrusion Detection Systems

Kangseok Kim
Dept. of Cyber Security, Ajou University, 16499, Suwon, Korea
Dept. of Artificial Intelligence and Data Science, Graduate School of Ajou University, 16499, Suwon, Korea
ORCID: 0000-0001-8950-7577

Abstract

This study proposes a methodology for anomaly detection in HIDS using supervised and semi-supervised anomaly detection approaches by applying GAN (Generative Adversarial Network) based data augmentation. An anomaly-based intrusion detection system detects abnormal patterns based on deviations from expected normal behaviors; however, such a system has a low detection rate. Also a detection accuracy may vary depending on whether abnormal samples are used during learning. Moreover, it may vary according to the degree of class imbalance that means the imbalance of data class distributions. To avoid the problem and to enhance the low predictive accuracy, it might need to augment minority datasets through the creation of new samples. Therefore, recently, some of existing studies have involved the development of intrusion detection models using machine/deep learning algorithms to overcome the limitations of existing anomaly-based intrusion detection methodologies and to avoid class imbalance problems. In a similar vein, this study proposes a method for improving classification performance of normal and abnormal data in anomaly-based intrusion detection systems by applying data augmentation using GAN. To verify the effectiveness of the proposed anomaly detection method, we use the ADFA-LD Dataset which consists of system call traces for attacks on the latest operating systems. Experiments were performed using SVM (Support Vector Machine) and CNN (Convolution Neural Network) for classification, and GAN and SMOTE for data augmentation, respectively. The experimental results indicated that GAN based approach provides a slightly more reliable way of working with data augmentation than SMOTE. In addition, it was confirmed based on the experimental results that the classification performance can be improved as the number of samples belonging to each imbalanced class increases.

Keywords: anomaly detection, host based intrusion detection system, system calls, cyber security, machine / deep learning, GAN

I. INTRODUCTION

Recently, with the rapid evolution of software, hardware, and networks, people, objects, and spaces have been becoming more closely connected through the development of real-time information service systems such as social network services (SNSs) and Internet of Things (IoTs) over Internet. But at the same time, among them, the connectivity and the facilitated flows of information and massive data over Internet have exposed them to various threat factors, including hacking and malwares, such as computer virus, worm, and ransomware. To mitigate such threats, firewalls, which form hardware or software at the frontline of increasing security, prevent intrusions from untrusted external networks to trusted internal networks. Nevertheless, these networks may be still considerably vulnerable to attacks. In addition, with the advent of advanced intelligent cyber threats, the importance of threat detection and security has increased significantly in recent systems and networks. Therefore, intrusion detection systems (IDSs) [1, 2, 3], which have been studied for a long time, have been developed as next-generation security technology against constantly evolving attacks.

Typically, network packets pass through IDSs after passing through firewalls, and IDSs generate an alert if they detect malicious activities or determine anomalies in the incoming data [4]. In addition, IDSs detects and responds to unauthorized activities against target systems that are not certified [5]. Thus, an IDS is an important tool for detecting security violations in real time. IDSs can be classified into two types of intrusion detection systems: a host-based IDS (HIDS) and network IDS (NIDS) based on the position and purpose of detection area according to data source-based classification [6]. In order to detect malicious behaviors such as DoS attacks and port scans, an HIDS analyzes information collected from specific host systems, while an NIDS monitors network traffic [7, 8]. Unlike the NIDS which detects attack vectors based on network traffic, the HIDS focuses on monitoring and analyzing internal systems, instead of external networks. In general, HIDS provides intrusion detection methods for each individual host, providing broader security than NIDS.

HIDSs can further be categorized by the type of model used for intrusion detection, namely misuse detection method and anomaly detection method. Both use information extracted from the target data to determine if an intrusion has occurred [9, 10, 11]. The misuse detection method, which is used in a signature-based (or knowledge-based) HIDS, is effective in detecting known attack vectors (known intrusion events); nevertheless, it is vulnerable to intrusions from unknown attack vectors. The anomaly detection is the identification of abnormal patterns based on deviations from expected normal behaviors [12]. Therefore, there is a need for anomaly detection methods to detect abnormal patterns (unknown attack vectors or anomalies) that deviate from normal behavior patterns based on existing network usage scenarios, internal system calls, and so on [7, 13, 14].

In order to define normal behavior patterns, it is, therefore, necessary to extract normal behavior and anomaly patterns in HIDS. Then, anomaly detection models can be developed using machine/deep learning algorithms based on iterative

learning and data mining models with mathematical and statistical methods on these extracted patterns. Recently the research on cyber security has been emerging with machine learning and artificial neural networks. In cyber security, the accuracy for anomaly detection models based on machine/deep learning may vary depending on whether abnormal samples are used during learning. Therefore, in this study, we consider two cases: when both normal samples and abnormal samples exist in a training data set, and only normal samples exist in a training data set which is called one-class classification (or semi-supervised classification learning). Also we consider a class-imbalance problem in learning. As the anomaly detection model is applied, the incidence of abnormal samples may be significantly lower than that of normal samples, and then the distribution of samples is said to be unbalanced. In order to avoid this problem, studies [15, 16] such as data augmentation or oversampling to augment the minority class dataset through the creation of new samples have been conducted.

Therefore, the purpose of this study is to increase the accuracy of anomaly detection by applying machine/deep learning models to preprocessed data from system call sequence dataset released by [17]. Then an N-gram [18] method, which is one of data representation techniques, is used to preprocess the system call sequence dataset. In addition, after applying data augmentation using Generative Adversarial Networks (GANs) to the preprocessed data, this study conducts a method for improving the accuracy of anomaly detection in HIDS using supervised and semi-supervised classification learning methods.

The remainder of the paper is organized as follows. Section II discusses previous studies that integrate machine/deep learning algorithms and intrusion detection systems. Section III describes a set of approaches conducted in this study for classification of anomalies in Host-based Intrusion Detection Systems. Section IV describes the experiments conducted in this study using supervised and semi-supervised anomaly detection methods by applying data augmentation using GAN to the preprocessed data. Finally, Section V provides conclusions and directions for future research.

II. RELATED WORK

A significant number of anomaly detection models in HIDS have been proposed to increase an accurate detection rate and to reduce a false alarm rate. Many studies have improved HIDSs by evaluating the recognition of abnormal patterns using HMM (Hidden Markov Model), KNN (K-Nearest Neighbor), Logistic Regression, SVM (Support Vector Machine), Ensemble algorithm, and so on. In addition, extensive research has been performed on applying data mining techniques on the new dataset to develop models for HIDS [11]. The paper [19] provided a survey of HIDSs with system calls, from the viewpoint of algorithms, techniques, datasets, application areas, and future research trends to inspire researchers about system-call-based HIDS in the big data and cloud environment. The paper [20] also discussed about deep learning-based IDSs through reviews such as input data, detection, deployment, and evaluation strategies. Recently, researches for intrusion detection have been moving from machine learning technologies towards various kinds of artificial neural networks (ANNs) technologies such as CNN (Convolution Neural Networks), LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit), Autoencoder, and GAN (Generative Adversarial Network) and so on. Therefore, this section describes existing models or methods tried to detect anomalies in HIDS using machine learning or artificial neural networks.

In the study [21], considering the advancements in computer systems, as a preliminary work, researchers used ADFA-LD dataset to evaluate a new host-based anomaly detection system (HADS) instead of outdated datasets that were previously used. The common patterns and frequency of attacks were evaluated by KNN-based HADS with the AFDA-LD dataset. Although acceptable detection results were obtained from the proposed HADS, it still had a weakness in that it could not identify the behaviors of some attacks from normal behaviors through the model. Also, since deep learning generally outperforms classical machine learning as the amount of data increases [22], the HADS model will be ineffective with large-scale training data in detecting and classifying future cyberattacks. Hence, the paper [22] proposed a scalable solution through hybrid DNNs (Deep Neural Networks) framework (monitoring network and host-level events) for detecting and classifying cyberattacks occurring in very large volumes. The work showed that DNNs perform well in comparison to classical machine learning classifiers with various datasets to identify the best algorithm which can effectively work in detecting unforeseen and unpredictable cyberattacks.

The study in [23] proposed a modified vector space representation to extract patterns from labelled ADFA-LD and ADFA-WD system call trace datasets, varying term-size. Also the study considered binary class and multiclass classification for evaluation with various machine learning classification algorithms and conformed that higher term-size preserves more system call sequence information. In the paper [6], various machine learning techniques have been carried out for finding the cause of problems associated in detecting intrusive activities. The study described the difficulties associated with detecting low-frequency attacks using machine learning techniques. It has motivated researchers to explore deep learning approaches to detect the low-frequency attacks. The study [24] described a survey of deep learning-based anomaly detection and presented key assumptions to differentiate between normal and anomalous behaviors. Also it discussed the computational complexity of anomaly detection techniques. The study [25] discussed the deep model based anomaly detection techniques used to overcome the limitations from traditional algorithms in real world examples from LinkedIn production systems. This paper [26] described a computationally efficient anomaly based intrusion detection model through the incorporation of stacked CNNs with GRUs to obtain reduced training times.

In [27], researchers developed a frequency-based misuse detection method using an ensemble classification. After preprocessing the raw ADFA-LD system call traces using N-gram method, patterns were generated by extracting features; in addition, the number of patterns were balanced based on class through SMOTE (Synthetic Minority Over-sampling Technique) [28] that is an approach to the construction of classifiers from imbalanced datasets, which mean an unequally represented datasets. In a similar vein, our work

International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 13, Number 11 (2020), pp. 3987-3996
 https://dx.doi.org/10.37624/IJERT/13.11.2020.3987-3996

was also done to balance the distribution of samples through the data augmentation of minority class while comparing SMOTE and GAN. GAN [29] is a type of generative model that is implemented by simultaneously two neural networks (Generative Network and Adversarial Network) competing with each other. The GAN model can create a real-like new image by learning features extracted from actual image data [30, 31, 32]. In particular, the study [33] applied numerical data, not image data, to GAN. The study showed that the model can also produce data similar to training dataset in learning by the two competing models.

The paper [34] proposes a methodology for host-based anomaly detection using a semi-supervised algorithm (one-class classifier) combined with a PCA-based feature extraction technique called Eigentraces on ADFA dataset. The paper [35] studied supervised / semi-supervised machine learning approaches for Host-based Intrusion Detection using system calls identifiers with ADFA dataset. The study used dimensionality reduction such as PCA (Principal Component Analysis), autoencoder, and RF-RFE (Random Forest - Recursive Feature Elimination) based on hybrid feature retrieval technique combining Integer Data Zero Watermark method and Frequency-based System Call modeling. In a similar vein, our work was also done not only with binary class and multiclass classification for evaluation, but also with supervised classification and semi-supervise classification.

As mentioned in the paper [25], recent advancement in deep learning techniques has made it possible to largely improve anomaly detection performance compared to the classical approaches. Therefore, our work has performed CNN-based classification for anomaly detection and GAN-based augmentation for oversampling of minority classes.

III. METHODOLOGY

This section describes a set of approaches conducted for classification of anomalies in Host-based Intrusion Detection Systems. Section A describes the experimental dataset used in this study and Section B describes data preprocessing using N-gram method. Section C describes data augmentation to solve class-imbalance problems in learning and Section D presents machine learning / deep learning models performed in this study. For classification, SVM and CNN based models were explored to develop an anomaly detection model in HIDS. Also GAN model was used for data augmentation of minority class through the creation of new sample data. It was compared to SMOTE used for data oversampling as well. Fig. 1 shows a simplified systematic representation for methodology used in the proposed study.

III.I Dataset

Since computer and network systems have evolved, new attack vectors and vulnerabilities have emerged. Therefore, HIDS developed on the basis of existing datasets does not properly take into account the features of current attack vectors, so these existing datasets are not suitable for HIDS evaluation and validation [17, 21]. Thus, alternative datasets reflecting current attack vectors have been proposed in [17]; an example of such a dataset is the research dataset provided by the Australian Defense Force Academy (ADFA) [17]. In many recent works, the ADFA dataset along with the latest attack vector features have been used for research on intrusion detection verification. In particular, the ADFA dataset was developed to evaluate a system call based HIDS as well as anomaly detection in signature-based HIDS.

The ADFA dataset is divided into the ADFA Linux dataset (ADFA-LD) and ADFA Windows dataset (ADFA-WD). The ADFA-LD reflects the features of current Linux-based operating systems, compared to many existing datasets used to evaluate the HIDS, and consists of thousands of system call traces collected from Linux local servers for the most recent attacks and vulnerabilities that occur in various applications. Considering this, the ADFA-LD is expected to become a new benchmark data for evaluating and verifying HIDS.

Thus, in this study, using ADFA-LD, we extracted the attack patterns against the current HIDS and applied the machine / deep learning and data mining techniques to the patterns to improve the accuracy of anomaly detection in HIDSs.

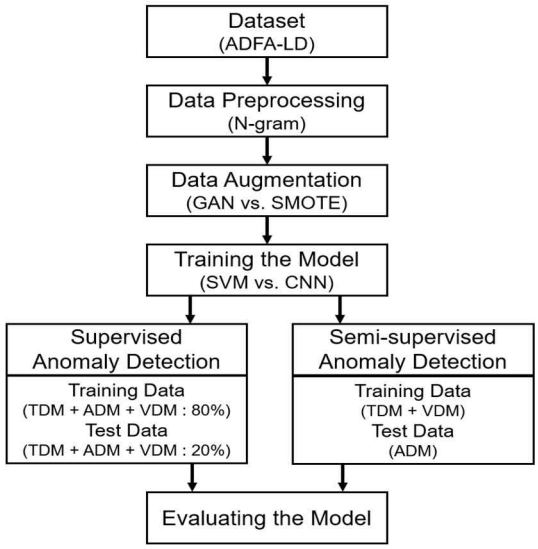


Fig. 1. A Simplified Systematic Representation for the Proposed Anomaly Detection Methodology

As previously mentioned, the ADFA-LD has thousands of normal traces collected from hosts on Linux servers, including abnormal trace files for six new types of cyberattacks, general user behavior and cyberattack path, and audit daemon setup, among others. In particular, during sampling periods for the ADFA-LD, a host captures system-call traces that are generated by normally functioning legitimate programs and stores the corresponding data in a file. Among them, 8-20 abnormal call traces are stored as attack data files using call traces generated after a cyberattack is initiated against the test host. As listed in Table 1, the ADFA-LD consists of three different data groups, each of which contains their own system call trace files. These data groups include training data master (TDM) and validation data master (VDM) groups, which represent normal data, whereas attack data master (ADM) group consists of call traces representing attack data. Furthermore, the ADM consists of six types of attack data: “Adduser”, “Hydra-FTP”, “Hydra-SSH”, “JavaMeterpreter”, “Meterpreter”, and “Web-Shell”.

International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 13, Number 11 (2020), pp. 3987-3996
 https://dx.doi.org/10.37624/IJERT/13.11.2020.3987-3996

Table 1. Data Groups in ADFA-LD Dataset

Data Groups	Type of Traces	Number of Traces
TDM	Normal	833
VDM	Normal	4372
ADM	Adduser	91
	Hydra-FTP	162
	Hydra-SSH	176
	JavaMeterpreter	124
	Meterpreter	75
	Web-Shell	118

III.II Data Preprocessing

The system-call trace data [17] are represented as a series of integer numbers corresponding to system calls made on Linux operating system. We apply machine / deep learning algorithms to the system call trace data and then classify the process operation into normal behaviour or six attack types. First, we used an N-gram technique to extract attribute vectors from the system call trace dataset. The N-gram method involves cutting a sample text into a contiguous sequence of N characters or words. For an N-gram of size 1, i.e., N = 1, the N-gram is referred to as unigram (1-gram), while for an N-gram of size 2, i.e., N=2, the N-gram is referred to as bigram (2-gram). In this study for N-gram, a word units consist of system call numbers, and the number of system call sequence attributes is derived by creating an array of N words according to the given word order. By doing this step repetitively, the call attributes of the system call traces can be obtained. Fig. 2 shows an example of applying the N-gram technique on system call trace data. In particular, N-gram data is expressed as a two-dimensional matrix; the columns of this matrix consist of the attribute values by matching the entire word belonging to each gram according N, while the rows represent instances that belong to each trace. The value corresponding to the row and column of the data represents the number of occurrences of N-gram in each trace as shown in Table 2. As the value of N increases, the model becomes more complicated and requires considerably more storage space, thereby increasing processing time. Therefore, in this study, we limited N to 1 to 5. Furthermore, in order to extract those instances that occur most frequently in an entire trace, we extracted and used only instances that they were used more than once in the entire trace and had more than 30% (0.3) of all the instances in the entire trace because the used instances were small.

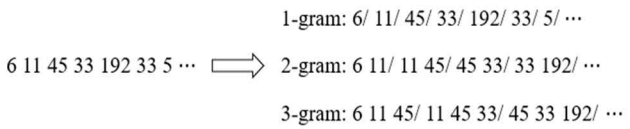


Fig. 2. An Example of N-Gram units.

III.III Data Augmentation

This work was done in two cases: SMOTE (Synthetic Minority Over-sampling Technique) [28] for data oversampling and GAN for data augmentation to solve the class-imbalance problem in learning. In this paper we use the terms "data oversampling" and "data augmentation" interchangeably. The SMOTE is an approach to oversampling minority class from unequally represented class datasets. Hence, for minority class augmentation, we used their implementation from the imbalanced-learn python library [36].

Table 2. The Number of Occurrences of N-Gram in each Trace

N-gram / System Call Trace	1-gram				···	5-gram			
	x_1	x_2	···	x_α	···	$x_{1'}$	$x_{2'}$	···	$x_{\alpha'}$
d_1	$N_{1,1}$	$N_{1,2}$	···	$N_{1,\alpha}$	···	$N_{1,1'}$	$N_{1,2'}$	···	$N_{1,\alpha'}$
d_2	$N_{2,1}$	$N_{1,2}$	···	$N_{1,\alpha}$	···	$N_{2,1'}$	$N_{1,2'}$	···	$N_{1,\alpha'}$
···	···	···	···	···	···	···	···	···	···
d_m	$N_{m,1}$	$N_{m,2}$	···	$N_{m,\alpha}$	···	$N_{m,1'}$	$N_{m,2'}$	···	$N_{m,\alpha'}$

GAN [29] is a type of unsupervised learning model used for dimensionality reduction, visualization, feature extraction, and so on. It is also a kind of generative model that can be used for data augmentation. That is, it is a kind of active model that can generate the data itself. The GAN is implemented by simultaneously two artificial neural networks (Generative Network and Adversarial Network) competing against each other (Generator and Discriminator). Then the generator takes any input data (noise vector data) and generates fake data. The discriminator takes real (original) and fake data, and distinguishes whether each of them is real or not. Therefore, the algorithm can produce data similar to the training (or input) dataset in learning by the two competing networks [30, 31, 32, 33]. Thus, a data augmentation approach using Generative Adversarial Network (GAN) was applied to the preprocessed data in this study. Fig. 3 depicts a simplified generative adversarial neural networks.

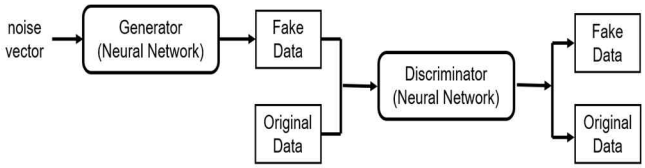


Fig. 3. A simplified generative adversarial neural networks.

Also, the work of constructing training / test dataset from dataset was done in two cases: supervised anomaly detection vs. semi-supervised anomaly detection (or one-class classification) methods, depending on whether or not abnormal (minority class) samples are used during learning. The supervised anomaly detection approach is a method of learning both normal and abnormal sample data and labels together in a given training data set. The semi-supervised anomaly detection method is to train only normal samples with no anomalies in a given training data set. In each case, abnormal samples were augmented through the creation of new samples since the accuracy for the anomaly detection model based on machine / deep learning may vary depending on whether abnormal samples are used during learning. The data augmentation was performed by SMOTE and GAN, as mentioned earlier.

International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 13, Number 11 (2020), pp. 3987-3996
 https://dx.doi.org/10.37624/IJERT/13.11.2020.3987-3996

III.IV Applied Machine / Deep Learning Models

A detection model structure can generally vary depending on collected or benchmarked dataset, data preprocessing techniques, machine / deep learning models, etc., as well as hardware / software performance. There are a variety of model structures for constructing classification models in machine / deep learning fields. Especially a model structure may differ depending on the performance of learning algorithms. Machine learning is an application of artificial intelligence that studies algorithms and technologies that enable computers to automatically learn from data. Deep learning is an advanced field of machine learning which uses multi-layer networks. The layers are connected through nodes, which represent the mathematical computation of learning processes [37]. Deep networks are generally classified into three main categories [11]: (1) generative (unsupervised learning) architectures to learn automatically from an unlabelled dataset, (2) discriminative architectures (supervised learning) to distinguish patterns for predictive tasks, and (3) hybrid architectures incorporating both generative and discriminative models. Recently, researches for intrusion detection have been moving from machine learning technologies towards various kinds of artificial neural networks technologies. This section briefly describes machine / deep learning models (SVM and CNN) used in this study.

In the study [38], we considered machine learning algorithms such as SVM, Logistic Regression and KNN algorithms for anomaly detection. In this work, for anomaly detections, CNNs (Convolution Neural Networks) and SVM (Support Vector Machine) models were compared based on the previous study. The SVM algorithm is easy to apply and has strong performance, so it is one of the most practical supervised learning algorithms used for classification, regression and anomaly detection in the field of traditional machine learning. In particular, SVM algorithm is a method of finding a hyperplane that maximizes the margins that are farthest from data among the hyperplanes that dichotomically divide data based on training data [39]. The SVM algorithm supports different kernel functions, which can solve large dimension issues, i.e., SVM does not suffer from problems associated with high dimensionality; in addition, the generalization ability of the SVM method can be enhanced by increasing margins during the training process [40]. Considering these features of SVM algorithm, it would be appropriate to classify the data represented using moderate-sized matrices preprocessed for experiments.

CNNs [41] is a class of deep neural networks used in various applications and domains such as text, image and video processing. The hidden layers in CNNs are composed of convolutional layers and pooling layers. Filters, the detecting functions of CNNs, are used to extract related features from input data using convolution operations that generate feature maps. CNNs is a generally good modelling for the recognition of huge amounts of images, but they can detect the important features like a series of related events or traces in a variety of sequence data. Also, in general, CNNs show higher efficiency in terms of time and performance than ANNs (Artificial Neural Networks). Therefore, it would be a good candidate for classification. RNNs (Recurrent Neural Networks) can also be a good candidate for sequential event data. Future work will consider RNN modellings such as LSTM and GRU with various sequential or temporal benchmarked dataset.

III.V Evaluation

This section briefly describes classification metrics to evaluate the performance of models performed in this study. For the purpose of evaluation, Accuracy, True Positive Rate (TPR) and False Alarm Rates (FPR) were defined as:

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN)$$

$$\text{True Positive Rate (TPR) or Recall} = TP / (TP + FN)$$

$$\text{False Positive Rate (FPR)} = FP / (FP + TN)$$

Where P: real positive cases in the data, N: real negative cases in the data, TP: True Positive, FP: False Positive, TN: True Negative, and FN: False Negative.

Also ROC (Receiver Operating Characteristic) curve plots TPR on Y-axis against FPR on X-axis of all possible thresholds for a binary classifier. Then another metric related to ROC curve is AUC (Area Under the ROC Curve). AUC provides overall measure of performance of a model across all the possible classification thresholds [42, 43]. AUC-ROC curve is a performance measurement for classification problem at various thresholds. AUC refers to the degree to which the model can distinguish between classes. The AUC score was used for performance evaluation in this study.

In next section we will show the experimental results obtained from using the preprocessed datasets with methods mentioned in this section.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this study, we conducted and compared experiments using machine / deep learning algorithms: SVM and CNN for classification, and GAN and SMOTE for data augmentation respectively. Also, to construct train / test dataset from preprocessed dataset using N-gram technique, it was done in two cases: supervised anomaly detection and semi-supervised anomaly detection (or one-class classification) methods, depending on whether or not abnormal (minority class) samples are used during learning. In each case, abnormal samples were augmented through (the creation of synthetic data based on original attack dataset) the creation of new samples by SMOTE and GAN. The TDM and VDM data are the normal data, while the ADM data consists of the six types of attack data, which are listed along with their labels in Table 3. Then, the training data was used to model the algorithm, and the test data was used to validate the algorithm to ensure accuracy. In the case of supervised anomaly detection, the attack data to be detected, the ADM dataset, was used with a ratio 2 to 8 for the training data.

For SVM model, we conducted experiments using Radial Basis Function (RBF) as kernel functions. The AUC scores measured after applying the RBF kernel function by adjusting hyperparameter C are shown in Fig. 6, 7, 8, and 9 compared to the CNN. For semi-supervised anomaly classification, this work used an implementation of scikit-learn python library [44, 45], which is a version of SVM implementation suitable for one-class classification on a training dataset containing only regular data.

International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 13, Number 11 (2020), pp. 3987-3996
 https://dx.doi.org/10.37624/IJERT/13.11.2020.3987-3996

Table 3. ADFA-LD Data Labeling and Number of Attack Data according to the Ratio of Attack Data to Training Data

Data Group	Type of Traces	Label	Ratio			
			0.3	0.5	0.7	1
TDM	Normal	0	5205	5205	5205	5205
VDM	Normal	0				
ADM	Adduser	1	208	312	468	624
	Hydra-FTP	2	312	572	780	1145
	Hydra-SSH	3	364	624	884	1249
	JavaMeterpreter	4	260	416	624	884
	Meterpreter	5	156	260	364	520
	Web-Shell	6	260	416	572	832

For CNN, this model performs three convolution and max pooling operations, followed by two fully connected layers. The convolution is applied on the input data using a convolution filter with size 1x2 to produce a feature map. Feature maps of various sizes (?x108x1, ?x54x1, ?x27x1) were generated through the operations. To generate the feature maps, the Relu activation function was applied. Pooling was then applied over the feature maps using a 1x1 max pooling with a stride of 2 without padding. After those operations, the model flattens last feature map to the size (?x27) and passes the flattened data to a deep neural network (fully connected layers with dropout 0.5). The model was trained with 50 epochs. Then the model returns classification results. The loss function of classifier was calculated by binary cross entropy and was optimized through the Adam optimizer (learning rate = 0.0001, beta 1 = 0.5, beta 2 = 0.99). Fig. 4 shows the architecture of the CNN model performed.

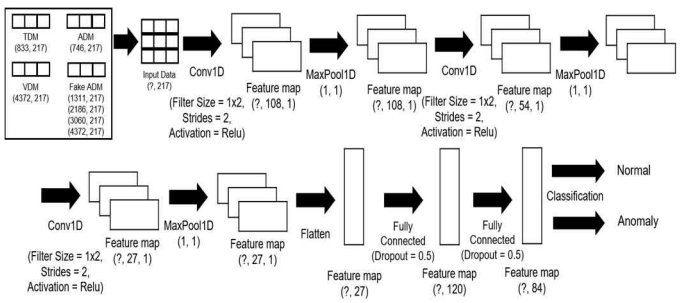


Fig. 4. CNN Model Architecture
(in the case of 3 hidden layers)

For GAN, the model was performed for data augmentation. Fig. 5 depicts the generative adversarial neural networks performed. The GAN was implemented by executing simultaneously generative neural network (generator) and adversarial neural network (discriminator). Then the generator takes any noise vector data and generates fake anomaly data. The discriminator takes both original and fake anomaly data, and distinguishes whether each of them is real or fake. In this work we used the loss function of WGAN [47] which is one of variants of GAN since it can improve the stability of learning by redefining the loss function of GAN. The size of hidden layer is 128 and the epoch was set to 50. The loss function of discriminator was calculated by softmax cross entropy and was optimized through the Adam optimizer (learning rate = 0.001, beta 1 = 0.5, beta 2 = 0.99). The random noise data were extracted from a uniform distribution with a minimum value of -1 and a maximum value of 1 to set a random number. The execution environment is as follows: Ubuntu 18.04.4 LTS, Intel® Xeon® Processor E5-2620v4, 2.10 GHz, GTX 1080 GPU, and 64GB RAM.

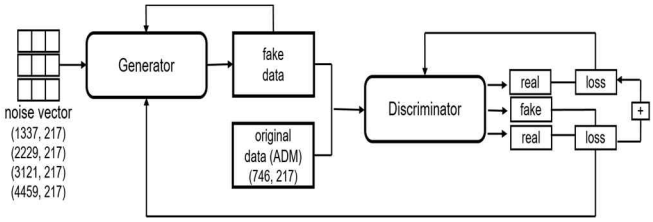


Fig. 5. Approach to GAN based Augmentation Performed

We evaluated the performance of SVM and CNNs models for classification of anomalies using the AUROC (Area Under the ROC curve) score. Also the performance of GAN and SMOTE for data augmentation with the classification performance was evaluated. The minority class datasets (original attack datasets) were augmented by 0.3, 0.5, 0.7, and 1 to 1 ratios for the normal class dataset, respectively in the case of supervised anomaly detection. In the case of semi-supervised anomaly detection, TDM and ADM samples were augmented respectively in the ratios as shown in the Table 3.

This study has compared several aspects of the classification performance of SVM and CNNs classifiers trained with samples increased by GAN and SMOTE, respectively in supervised anomaly detection and semi-supervised anomaly detection scenarios.

The first scenario describes the binary classification performance of supervised anomaly detection, as shown in Fig. 6. The figure shows the AUC Scores for binary classification of SVM and CNN classifiers trained with datasets augmented by SMOTE and GAN, respectively. Fig. 7 shows the summary of the highest prediction accuracies of classification performance of the SVM (hyperparameter C=10000) and CNN (with 3 hidden layers) classifiers. It also shows a comparison of SMOTE and GAN augmentation methods with the classification performance. The results show that models trained with augmented data performed better classification performance than those trained with the original dataset. The higher the rate at which minority class samples (attack data samples) increase, the better the results as well. In addition, in data augmentation, GAN performed slightly better than SMOTE in terms of classification results. As considering AUC scores, SVM had slightly better or similar results than the CNNs. The SVM model seems to be appropriate with the experimental datasets.

In general, SVM is a good model for moderate-sized of training time. In these cases, artificial neural networks suitable for large datasets may be the right model. Thus, when data augmentation is required, that is, as the size of dataset increases, CNN can be a more suitable model for data classification than SVM. Future experimental work will estimate training time as well as classification performance of models being performed as data increases.

International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 13, Number 11 (2020), pp. 3987-3996
 https://dx.doi.org/10.37624/IJERT/13.11.2020.3987-3996

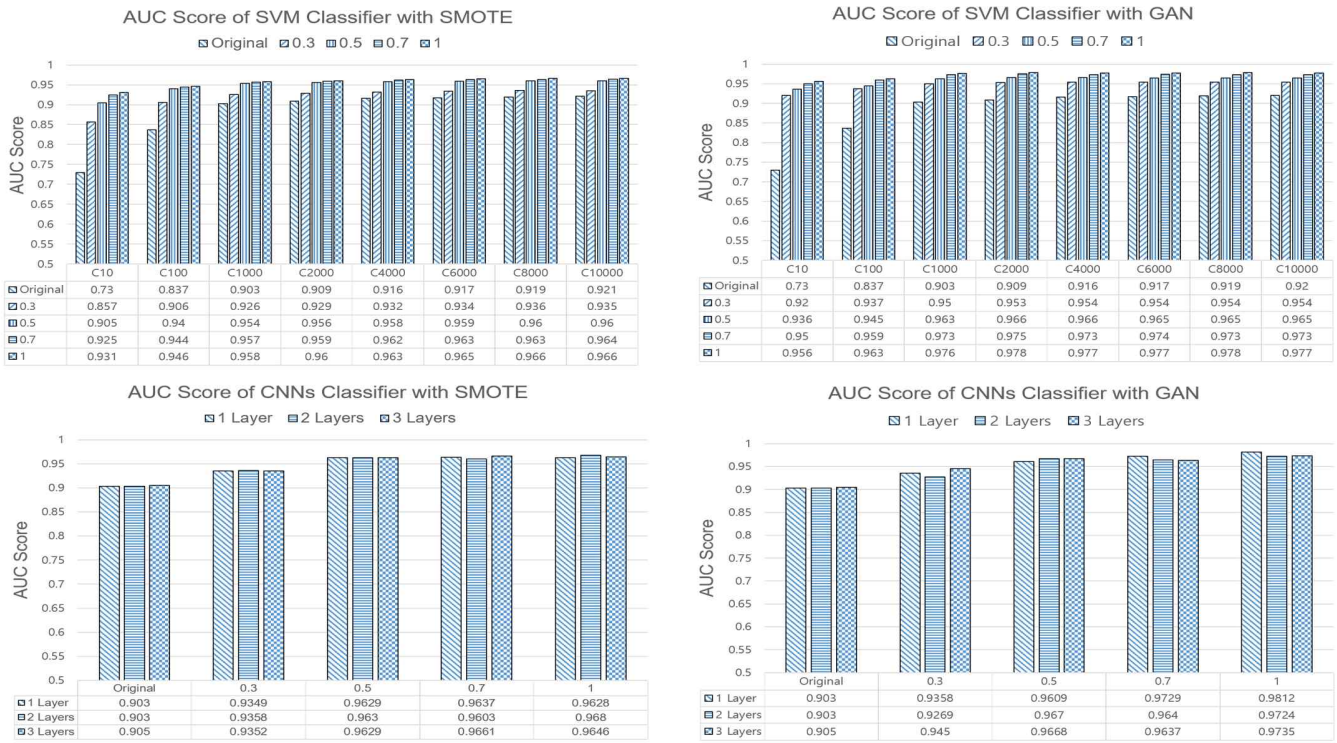


Fig. 6. Performance (AUC Score) for Binary Classification of SVM and CNN Classifiers Trained with Datasets Augmented by SMOTE and GAN, respectively, in Supervised Anomaly Detection Approach

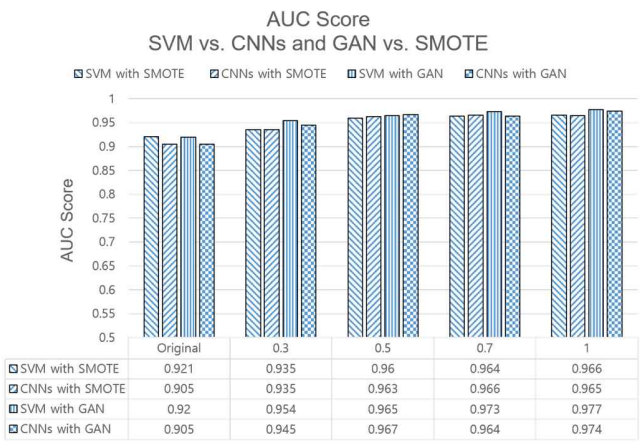
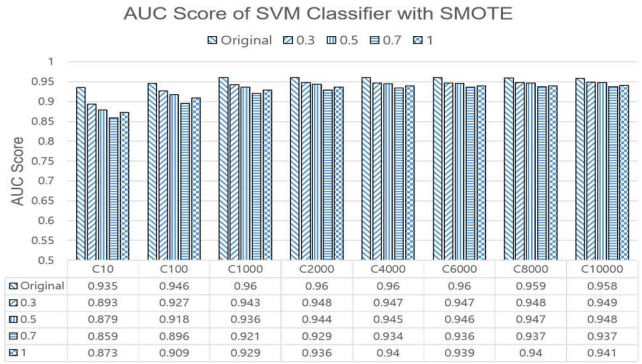


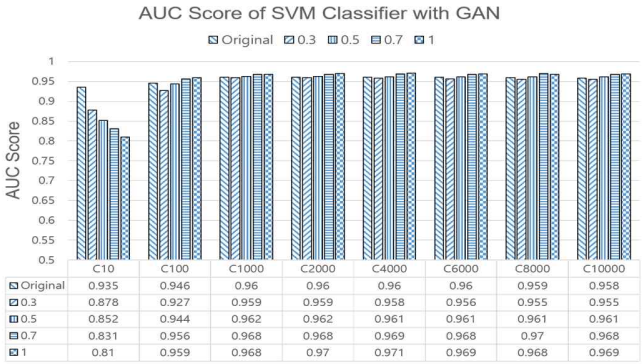
Fig. 7. The Summary of the Highest Prediction Accuracies of Classification Performance of SVM (C=10000) and CNN (with 3 hidden layers) Classifiers

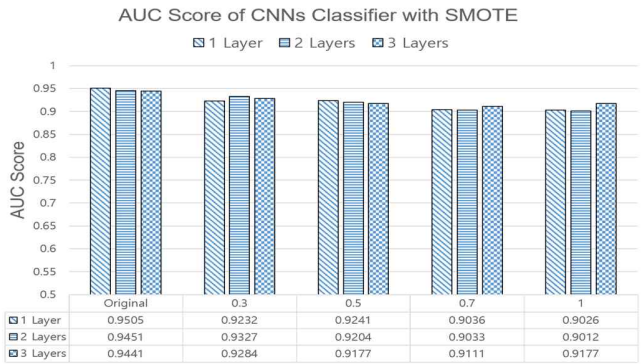
In the second scenario, Fig. 8 describes the multiclass classification performance in supervised anomaly detection. The given training and testing datasets respectively were classified into 7 classes, each with a different number of samples for each class. In the approach of multiclass classification, increasing the minority class sample growth rate in data augmentation by GAN shows better results than the original datasets as shown in Fig. 8. However, as the growth rate in data augmentation by SMOTE increased, the classification performance gradually decreased. As synthetic samples by SMOTE are generated and increased, the noise data also seems to increase with the sample data. Therefore, these noisy data can affect classification performance and appear to be relatively ineffective compared to GAN. In addition, GAN performed better than SMOTE in data augmentation to improve classification performance. The AUC scores for each class were summed and averaged. When considering the AUC score on average, SVM gave slightly better or similar results than CNN, as in the case of binary classification.

Fig. 9 shows the summary of the highest prediction accuracies of classification performance of the SVM (C=10000) and CNN (with 3 hidden layers) classifiers. When comparing the detection rates of binary classification and multiclass classification, the performance of binary classification was better than that of multiclass classification. This is because despite the data growth, there is still not enough training data to improve the model performance as the number of samples belonging to each class is still fewer than the samples belonging to the class of binary classification.

International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 13, Number 11 (2020), pp. 3987-3996
 https://dx.doi.org/10.37624/IJERT/13.11.2020.3987-3996







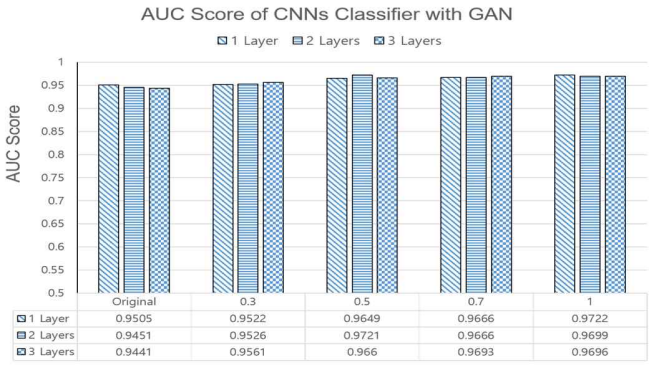


Fig. 8. Performance (AUC Score) for Multiclass Classification of SVM and CNN Classifiers Trained with Datasets Augmented by SMOTE and GAN, respectively, in Supervised Anomaly Detection Approach

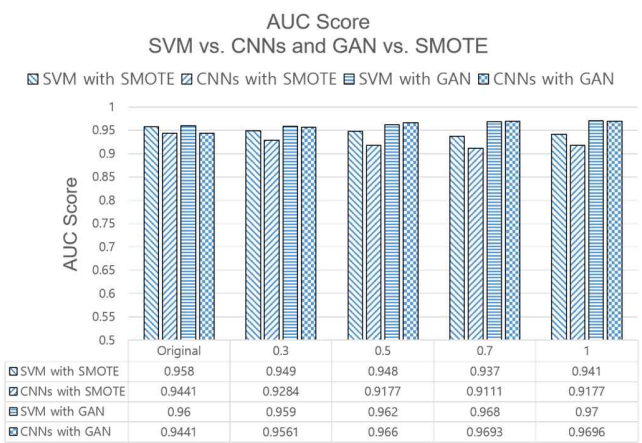


Fig. 9. The Summary of the Highest Prediction Accuracies of Classification Performance of SVM (C=10000) and CNN (with 3 hidden layers) Classifiers

In the third scenario, Table 4 describes the binary classification performance (recall scores) in semi-supervised anomaly detection. Classifiers of the semi-supervised anomaly detection approach were constructed by training only normal samples with no anomalies in a given training data set. The test dataset containing only anomalies was used to detect anomalies with classifiers trained with the training dataset. The experiment was performed with augmentation of training dataset and test dataset, respectively as shown in the table. In addition, a comparison between classifiers trained with augmented data and classifiers trained with original dataset is shown in the table. Overall, GAN's task provided better recall scores than SMOTE. However, the augmentation task in semi-supervised anomaly detection scenario was not stable (or robust) than that in supervised anomaly detection scenario in terms of recall scores and augmentation rates. GAN provided a slightly more reliable method (a little less variance over recalls) for data growth task than SMOTE. Compared to increasing the training dataset alone, increasing the training and test datasets at the same rate resulted in a lower recall score. The reason is that it is difficult for classifiers to detect increased unseen anomalies as the test dataset grows. The SVM provided more robust and slightly better classification accuracy than CNNs as in the case of supervised anomaly detection approach.

Table 4. Binary Classification Performance (Recall Score) of SVM and CNN with SMOTE and GAN in Semi-Supervised Anomaly Detection Approach

Training data: Test data (# of samples)		SVM		CNN	
Training	Test	SMOTE	GAN	SMOTE	GAN
833	746	0.961	0.961	0.874	0.874
1311	746	0.966	0.972	0.88	0.942
2186	746	0.964	0.97	0.921	0.946
3060	746	0.97	0.973	0.956	0.969
4372	746	0.97	0.976	0.972	0.972
1311	1311	0.961	0.952	0.899	0.913
2186	2186	0.94	0.943	0.943	0.936
3060	3060	0.958	0.964	0.933	0.935
4372	4372	0.964	0.97	0.952	0.958

These findings show that the accuracy for anomaly detection model based on machine / deep learning can vary depending on whether abnormal samples are used during learning, and binary classification task can result in better detection accuracy than multiclass classification task. In addition, the classification task in supervised anomaly detection scenario resulted in better detection accuracy than that in semi-supervised anomaly detection scenario. Data augmentation method was more stable in supervised anomaly detection scenario compared to semi-supervised anomaly detection scenario. Also SVM was a little more suitable model for data classification than CNN in experiments with the moderate-sized datasets. Future work will examine training time and resource consumption for the models with vast datasets.

V. CONCLUSIONS

This study proposed a method for improving classification performance of normal and abnormal data in anomaly-based intrusion detection systems by applying data augmentation using GAN. The experiments for improving classification performance were performed by comparing SVM and CNN classifiers trained with datasets preprocessed with N-gram. ADFA-LD, which consists of various system call traces for attacks on the latest operating systems, was preprocessed with N-gram technique. In addition, this study suggested using GAN to generate synthetic data for augmenting the samples of minority classes in class imbalanced classification scenarios. The performance of GAN for data augmentation through classification performance was compared with SMOTE technology. GAN provided a slightly more reliable way of working with data augmentation than SMOTE. The results show that GAN approach can improve the accuracy of classifiers trained with data augmented for minority classes.

To construct training / test dataset, it was done in two approaches: supervised anomaly detection and semi-supervised anomaly detection methods, depending on whether or not abnormal samples are used during learning. In each case, abnormal samples were augmented through the creation of synthetic data based on original attack dataset. Data augmentation method was more stable in supervised anomaly detection scenario compared to semi-supervised anomaly detection scenario. Based on experimental results, it was confirmed that the classification performance can be improved as the number of synthetic samples belonging to each imbalanced class increases.

In conclusion, the methodology proposed in this study enables the detection of normal data and attack data as well as the classification of each attack data by extracting the patterns and features of anomalies using machine / deep learning algorithms and applying them to anomaly detection in the HIDS, thereby significantly improving the HIDS, and thus, accurate detection rate.

Most machine learning algorithms can produce inefficient or unstable classifiers when learning with large datasets as well as class imbalance datasets. As dataset grows, in general, machine learning models may require a lot of training time. Then, artificial neural networks suitable for huge datasets may be the right model. Hence, future work will consider using of a variety of machine learning and deep learning models with vast datasets, and estimate training time as well as classification performance of models being performed as dataset increases. We will also consider the use of GAN with a variety of datasets depending on the degree of class imbalance.

ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT: Ministry of Science and ICT) (No. NRF-2019R1F1A1059036).

REFERENCES

[1] D. Wagner and P. Soto, “Mimicry Attacks on Host-Based Intrusion Detection Systems,” Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS '02), pp. 255-264, Washington DC, 18-22 Nov. 2002, http://dx.doi.org/10.1145/586110.586145

[2] K. A. Scarfone and P. M. Mell, "Guide to Intrusion Detection and Prevention Systems (IDPS)," NIST special publication, Feb. 20, 2007. https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=50951

[3] H-J. Liao, C-H. R. Lin, Y-C. Lin, and K-Y. Tung, “Intrusion Detection System: A Comprehensive Review,” Journal of Network and Computer Applications, vol. 36, no. 1, pp. 16-24, Jan. 2013, https://doi.org/10.1016/j.jnca.2012.09.004

[4] H. Cavusoglu, B. Mishra, and S. Raghunathan, “A Model for Evaluating IT Security Investments,” Communications of the ACM, vol. 47, no. 7, pp. 87-92, July 2004.

[5] K. Richards, “Network based Intrusion Detection: A Review of Technologies,” Computers & Security, vol. 18, no. 8, pp. 671-682, 1999.

[6] P. Mishra, V. Varadharajan, U. Tupakula and E. S. Pilli, “A Detailed Investigation and Analysis of Using Machine Learning Techniques for Intrusion Detection,” IEEE Communications Surveys & Tutorials, vol. 21, no. 1, pp. 686-728, Firstquarter 2019, https://doi,org/10.1109/COMST.2018.2847722

[7] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, “A Survey of Intrusion Detection Techniques in Cloud,” Journal of Network and Computer Applications, vol. 36, no. 1, pp. 42-57. Jan. 2013.

[8] A. Azab, M. Alazab, and M. Aiash, “Machine Learning Based Botnet Identification Traffic,” 2016 IEEE Trustcom/BigDataSE/ISPA, Tianjin, 2016, pp. 1788-1794, https://doi.org/10.1109/TrustCom.2016.0275

[9] O. Depren, M. Topallar, E. Anarim, and M. K. Ciliz, “An Intelligent Intrusion Detection System (IDS) for Anomaly and Misuse Detection in Computer Networks,” Expert Systems with Applications, vol. 29, no. 4, pp. 713-722, Nov. 2005.

[10] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Macia-Fernandez, and E. Vazquez, “Anomaly-based Network Intrusion Detection: Techniques, Systems and Challenges,” Computers & Security, vol. 28, no. 1-2, pp. 18-28, 2009.

[11] G. Creech and J. Hu, “A Semantic Approach to Host-based Intrusion Detection Systems using Contiguous and Discontiguous System Call Patterns,” IEEE Transactions on Computers, vol. 63, no. 4, pp. 807-819, Apr. 2014.

[12] L. Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, and M. Kloft, “Deep One-Class Classification,” Proceedings of the 35th International Conference on Machine Learning, PMLR 80, pp.4393-4402, 2018.

[13] A. Torkaman, G. Javadzadeh, and M. Bahrololum, “A Hybrid Intelligent HIDS Model using Two-layer Genetic Algorithm and Neural Network,” 5th Conference on Information and Knowledge Technology (IKT), pp. 92-96, 28-30 May 2013.

[14] P. García-Teodoroa, J. Díaz Verdejoa, G. Maciá-Fernándeza, and E. Vázquez, “Anomaly-based Network Intrusion Detection: Techniques,” Computers & Security, vol. 28, no. 1-2, pp. 18-28, 2009, http://dx.doi.org/10.1016/j.cose.2008.08.003.

[15] H. He and E. A. Garcia, "Learning from Imbalanced Data," IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 9, pp.1263-1284, 2009, https://doi.org/10.1109/TKDE.2008.239

[16] L. Mathews and S. Hari, "Learning from Imbalanced Data," in Encyclopedia of Information Science and Technology, pp. 1825–1834, IGI Global, Fourth edition, 2018.

[17] G. Creech and J. Hu, "Generation of a New IDS Test Dataset: Time to Retire the KDD Collection," Wireless Communications and Networking Conference (WCNC 2013), Shanghai, 7-10 April 2013 http://dx.doi.org/10.1109/WCNC.2013.6555301

[18] X. Zhang, Y, Wang, M. Gou, M. Sznaier, and O. Camps, "Efficient Temporal Sequence Comparison and Classification using Gram Matrix Embeddings on a Riemannian Manifold," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4498-4507, June 2016. https://doi.org/10.1109/CVPR.2016.487

[19] M. Liu, Z. Xue, X. Xu, C. Zhong, and J. Chen, "Host-based Intrusion Detection System with System Calls: Review and Future Trends," ACM Computing Surveys (CSUR), vol. 51, no. 5, pp. 1-36, Nov. 2018. https://doi.org/10.1145/3214304

[20] A. Aldweesh, A. Derhab, and A. Z. Emam, "Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues," Knowledge-Based Systems, Vol. 189, Feb. 15, 2020. https://doi.org/10.1016/j.knosys.2019.105124

[21] M. Xie and J. Hu, "Evaluating Host-Based Anomaly Detection Systems: A Preliminary Analysis of ADFA-LD," 6th IEEE International Congress on Image and Signal Processing (CISP '03), pp. 1711-1716, Dec. 2013.

[22] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," IEEE Access, vol. 7, pp. 41525-41550, 2019, https://doi.org/10.1109/ACCESS.2019.2895334

[23] B. Borisaniya and D. Patel, "Evaluation of Modified Vector Space Representation Using ADFA-LD and ADFA-WD Datasets," Journal of Information Security, vol. 6, no. 3, pp. 250-264, 2015.

[24] R. Chalapathy and S. Chawla, "Deep Learning for Anomaly Detection: A Survey," arXiv:1901.03407, Jan. 2019. https://arxiv.org/pdf/1901.03407.pdf

[25] R. Wang, K. Nie, T. Wang, and B. Long, "Deep Learning for Anomaly Detection," WSDM '20: Proceedings of the 13th International Conference on Web Search and Data Mining, Jan. 2020, pp. 894–896, https://doi.org/10.1145/3336191.3371876

[26] A. Chawla, B. Lee, S. Fallon, and P. Jacob, "Host based Intrusion Detection System with Combined CNN/RNN Model", ECML PKDD 2018 Workshops. ECML PKDD 2018. Lecture Notes in Computer Science, Springer, Cham. vol. 11329, Sept. 10-14, 2018, https://doi.org/10.1007/978-3-030-13453-2_12

[27] E. Aghaei, "Machine Learning for Host-based Misuse and Anomaly Detection in UNIX Environment," M.S. Thesis, Computer Science in University of Toledo, May 2017.

[28] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," Journal of Artificial Intelligence Research, vol. 16, pp. 321-357, June 2002.

[29] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Nets," Advances in Neural Information Processing Systems, pp. 2672–2680, 2014.

[30] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," Proceedings of The International Conference on Learning Representations (ICLR), pp. 1-16, 2016, https://arxiv.org/abs/1511.06434

[31] R. A. Yeh, C. Chen, T. Y. Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic Image Inpainting With Deep Generative Models," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5485-5493, 2017, https://arxiv.org/abs/1607.07539

[32] M. Frid-Adar, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Synthetic Data Augmentation using GAN for Improved Liver Lesion Classiffication", IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), vol. 15, 2018, 10.1109/ISBI.2018.8363576

[33] F. H. K. S. Tanaka, and C. Aranha, "Data Augmentation Using GANs," Proceedings of Machine Learning Research, ArXiv, vol. abs/1904.09135, 2019, https://arxiv.org/abs/1904.09135

[34] E. Aghaei and G. Serpen, "Host-based Anomaly Detection using Eigentraces Feature Extraction and One-class Classification on System Call Trace Data," ArXiv, p.11, 2019.

[35] R. Taj, "A Machine Learning Framework for Host based Intrusion Detection using System Call Abstraction," Master Thesis, Dalhousie University, Halifax, Nova Scotia, Apr. 2020. https://dalspace.library.dal.ca//handle/10222/78469

[36] G. Lemaˆıtre, F. Nogueira, and C. K. Aridas, "Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning," Journal of Machine Learning Research, vol. 18, pp.1-5, 2017, https://www.jmlr.org/papers/volume18/16-365/16-365.pdf

[37] I. Goodfellow, Y. Bengio and A. Courville, "Deep Learning," MIT Press, Nov. 2016, http://www.deeplearningbook.org

[38] Y-K. Shin and K. Kim, "Comparison of Anomaly Detection Accuracy of Host-based Intrusion Detection Systems based on Different Machine Learning Algorithms," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 11, no. 2, 2020.

[39] Y. B. Bhavsar and K. C. Waghmare, "Intrusion Detection System Using Data Mining Technique: Support Vector Machine," International Journal of Emerging Technology and Advanced Engineering, vol. 3, no. 3, pp. 581-586, March 2013.

[40] W. S. Noble, "What is a Support Vector Machine?," Nature Biotechnology, vol. 24, no. 12, pp. 1565-1567, Dec. 2006.

[41] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based Learning Applied to Document Recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998, doi: 10.1109/5.726791

[42] N. R. Cook, "Use and Misuse of the Receiver Operating Characteristic Curve in Risk Prediction," Circulation, vol. 115, no. 7, pp. 928–935, Feb. 2007. [Online]. Available: https://www.ahajournals.org/doi/10.1161/CIRCULATIONAHA.106.672402

[43] R. A. Maxion and R. R. Roberts, "Proper Use of ROC Curves in Intrusion / Anomaly Detection," University of Newcastle upon Tyne, Computing Science Tyne, UK, p. 33, 2004.

[44] Scikit-learn One-class SVM API, https://scikit-learn.org/stable/modules/generated/sklearn.svm.OneClassSVM.html

[45] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research (JMLR), vol. 12, no. 85, pp. 2825-2830, 2011, https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html

[46] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein Generative Adversarial Networks," Proceedings of the 34th International Conference on Machine Learning (ICML'17), vol. 70, pp.214-223, Aug. 2017,
https://dl.acm.org/doi/10.5555/3305381.3305404