Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 01, January-2023

A Review of Machine Learning-based Algorithms for Intrusion Detection System

Datti Emmanuel Useni
Department of Mathematical Science
Abubakar Tafawa Balewa University
Bauchi, Nigeria

Okere Chidiebere Emmanuel
Department of Mathematical Science
Abubakar Tafawa Balewa University
Bauchi, Nigeria

Abdulsalam Ya'u Gital
Department of Mathematical Science
Abubakar Tafawa Balewa University
Bauchi, Nigeria

Goteng Kuwunidi Job
Department of Mathematical Science
Abubakar Tafawa Balewa University
Bauchi, Nigeria

Mustapha Abdulrahman Lawal
Department of Mathematical Science
Abubakar Tafawa Balewa University
Bauchi, Nigeria

Abuzairu Ahmad
Department of Mathematical Science
Abubakar Tafawa Balewa University
Bauchi, Nigeria

***Abstract*— Networks play important roles in modern life, and cyber security has become a dynamic research area. An intrusion detection system (IDS) which is an important cyber security method, monitors the state of software and hardware running in the network. Despite decades of development, existing IDSs still face challenges in improving the detection accuracy, reducing the false alarm rate and detecting unknown attacks. To solve the above problems, many researchers have focused on developing IDSs that exploit on machine learning methods. Machine learning methods can automatically discover the essential differences between normal data and abnormal data with high accuracy. In addition, machine learning methods have strong generalizability, so they are also able to detect unknown attacks. In this paper, we conducted a comprehensive review on machine learning techniques used in building IDS.**

Keywords—Intrusion Detection; Machine Learning; Artififcial Neural Network;Support Vector Machine and K-Nearest Neighbour

I. INTRODUCTION

Intrusion Detection Systems (IDSs) plays a key role in passive defense (Wu et al., 2019) [1] aiming to detect malicious activity in different application domains such as the ones described in (Thing, V. L. L. & Wu, J. 2016) [2] & Omer, M. (2019) [3]. IDSs have been deployed in conjunction with active defense systems, such as honeypots. Two well-known approaches exist in IDS research: Host-based Intrusion Detection System (HIDS) and Network Intrusion Detection System (NIDS). The first approach monitors the target machine's network interfaces and configurations, requiring specific settings attuned to the host machine (Choudhary, S. & Kesswani, N. 2019) [4]. In dissimilarity to the host-based activity, a NIDS monitors all incoming and outgoing packets on the computer network and is designed upon signature- and anomaly-based detection approaches. By 2019, the cost to the global economy due to cybercrime is projected to reach $2 trillion as reported by Juniper Networks. Among the contributory felonies to cybercrime is intrusions, which is defined as illegal or unauthorized use of a network or a system by attackers (Anup K. G. Aaron, S. & Michael, S. 1999) [5], an intrusion detection system (IDS) is used to identify the said malicious activity. The most common method used for uncovering intrusions is the analysis of user activities (Srinivas, M. Guadalupe, J. & Andrew, S. 2002) [6] & (Anup K. G. Aaron, S. & Michael, S. 1999) [8]. However, the aforementioned method is laborious when done manually, since the data of user activities is massive in nature (Jeremy, F. 1994). An automated process through machine learning algorithms is proposed.

The purpose of this review is to explore machine learning based algorithms used in IDS, compare between Signature-based Anomaly-based IDS and also the benchmark dataset used in IDS.

The remaining of this paper is organized as follows: In section I related researches on ML based IDS are discussed, Section 3 presents different techniques or methods that have been used in modeling IDS, Machine learning and its algorithms are discussed in Section 4, Section 5 and 6 x-rays Datasets and statement of the research problem while the methodology and architecture of the proposed system are given in Section 7 and finally Section 8 gives a detailed conclusion and the future work.

II. RELATED RESEARCH ON MACHINE LEARNING BASED IDS

Machine learning is a type of data driven method in which understanding the data is the first step. Thus, we adopt the type of data source of as the main classification thread. In this section, we introduce various ways to apply machine learning to IDS design for different data types. The different types of data reflect different attack behaviors, which include host behaviors and network behaviors. Host behaviors are reflected by system logs, and network behaviors are reflected by network traffic. There are multiple attack types, each of which has a unique pattern. Thus, selecting appropriate data sources is required to detect different attacks according to the attack characteristics.

A. *Packet Parsing-Based Detection*

Various types of protocols are used in network communications, such as HTTP and DNS. These protocols have different formats; the packet parsing-based detection methods primarily focus on the protocol header fields. The usual practice is to extract the header fields using parsing tools (such as Wireshark or the Bro) and then to treat the values of the most important fields as feature vectors. Packet parsing-based detection methods apply to shallow models.

Mayhew, M. Atighetchi, M. Adler, A. & Greenstadt, R. (2015) [9] proposed an SVM- and K-means-based packet detection method. They captured packets from a real enterprise network and parsed them with Bro. First, they grouped the packets according to protocol type. Then, they clustered the data with the K-means++ algorithm for the different protocol datasets. Thus, the original dataset was grouped into many clusters, where the data from any given cluster were homologous. The header fields provide basic packet information from which feature can be extracted used with using classification algorithms to detect attacks. Their precision scores for HTTP, TCP, Wiki, Twitter, and E-mail protocols reached 99.6%, 92.9%, 99%, 96%, and 93%, respectively.

Hu, L. Li, T. Xie, N. & Hu, J. (2015) [10] proposed a fuzzy C-means based packet detection method. The fuzzy C mean algorithm introduces fuzzy logic into the standard K-means algorithm such that samples belong to a cluster with a membership degree rather than as a Boolean value such as 0 or 1. They used Snort to process the DARPA 2000 dataset, extracting Snort alerts, source IPs, destination IPs, source ports, destination ports, and timestamps. Then, they used this information to form feature vectors and distinguished false alerts from true alerts by clustering the packets. To reduce the influence of initialization, they ran the clustering algorithms ten times. The results showed that the fuzzy C-means algorithm reduced the false alarm rate by 16.58% and the missed alarm rate by 19.23%.

B. *Payload Analysis-Based Detection*

Apart from packet parsing-based detection, payload analysis-based detection places emphasis on the application data. The payload analysis-based methods are suitable for multiple protocols because they do not need to parse the packet headers.

Yu, Y. Long, J. & Cai, Z. (2017) [11] utilized a convolutional autoencoder to extract payload features and conducted experiments on the CTU-UNB dataset. This dataset includes the raw packets of 8 attack types. To take full advantage of convolutions, they first converted the packets into images. Then, they trained a convolutional autoencoder model to extract features. Finally, they classified packets using learned features. The precision, recall and F-measure on the test set reached 98.44%, 98.40%, and 98.41% respectively.

Liu, H.; Lang, B.; Liu, M.& Yan, H. (2019) [12] It should be noted that this method does not include encrypted payloads. Shallow models depend on manual features and private information in packets, leading to high labor costs and privacy leakage problems. As a type of unstructured data, payloads can be processed directly by deep learning models.

Zeng, Y.; Gu, H.; Wei, W. & Guo, Y. (2019) [13] proposed a payload detection method with multiple deep learning models. They adopted three deep learning models (a CNN, an LSTM, and a stacked autoencoder) to extract features from different points of view. Among these, the CNN extracted local features, the RNN extracted time series features, and the stacked autoencoder extracted text features. The accuracy of this combined approach reached 99.22% on the ISCX 2012 dataset.

Rigaki, M. & Garcia, S. (2018) [14] used a GAN to improve the malware detection effect. To evade detection, malware applications try to generate packets similar to normal packets. Taking the malware FLU as an example, the command & control (C & C) packets are very similar to packets generated by Facebook. They configured a virtual network system with hosts, servers, and an IPS. Then, they started up the malware FLU and trained a GAN model. The GAN guided the malware to produce packets similar to Facebook. As the training epochs increased, the packets blocked by the IPS decreased and packet that passed inspection increased. The result was that the malicious packets generated by the GAN were more similar to normal packets. Then, by analyzing the generated packets, the robustness of the IPS was improved.

C. *Flow-Based Attack Detection*

Flow data contains packets grouped in a period, which is the most widespread data source for IDSs. The KDD99 and the NSL-KDD datasets are both flow data. Detecting attacks with flow has two benefits: (1) Flow represents the whole network environment, which can detect most attacks, especially DOS and Probe. (2) Without packet parsing or session restructuring, flow preprocessing is simple. However, flow ignores the content of packets; thus, its detection effect for U2R and R2L is unsatisfactory. When extracting flow features, packets must be cached packets; thus, it involves some hysteresis. Flow-based attack detection mainly includes feature engineering and deep learning methods. In addition, the strong heterogeneity of flow may cause poor detection effects. Traffic grouping is the usual solution to this problem.

III. TECHNIQUES USED FOR INTRUSION DETECTION SYSTEM

In this section we discussed the two different detection methods for intrusion detection system and discussed the differences between the two methods as shown in table 1 below.

A. *Signature-based IDS*

Signature-based detection is also called misuse detection, the basic idea of the signature based is to represent attack behaviors as signatures. The detection process matches the signatures of samples using a signature database. Signature-based IDS may detect an attack/intrusion if the attack's signature is already stored in the internal database. These systems can detect known attacks very accurately and this is the reason why they are widely used in the industry (Philokypros et al., 2018) [15]. The main problem in constructing signature-based detection systems is to design efficient signatures. The advantages of this detection method is that it has a low false alarm rate and it reports attack types as well as possible reasons in detail; the disadvantages are that it has a high missed alarm rate, lacks the ability to detect unknown attacks, and requires maintaining a huge signature database.

B. Anomaly based learning IDS

In anomaly based, the system would first need to learn the NORMAL behaviour, traffic or protocol set of the network. When the system has learnt the normal state of a network and the types of packets and throughput it handles on a daily basis, taking into account peak times, when traffic is detected that is out of the normal state of the network, the anomaly based detection system would take action. An IDS which is anomaly based will monitor network traffic and compare it against an established baseline. The baseline will identify what is “normal” for that network- what sort of bandwidth is generally used, what protocols are used, what ports and devices generally connect to each other- and alert the administrator or user when traffic is detected which is anomalous, or significantly different than the baseline. The issue is that it may raise a False Positive alarm (Mohammad et al., 2012) [16] for a legitimate use of bandwidth if the baselines are not intelligently configured. Anomaly detection assumes that normal user behavior is seamlessly observable and adequately different from intrusive. It builds up a model for the normal user profile and the user behavior that differs from the established one, flagged as intrusion (Hamid, Y. Sugumaran, M. & Balasaraswathi, V. 2016) [17].

TABLE I: DIFFERENCE BETWEEN SIGNATURE-BASED AND ANOMALY-BASED METHODS

	Signature-Based Method	**Anomaly-Based Method**
Detection efficiency	High, decrease with scale of signature database	Dependent on model complexity
Interpretation	Design based on domain knowledge, strong interpretative ability	Outputs only detection results, weak interpretative ability
Unknown attack detection	Only detects known attacks	Detects known and unknown attacks
Dependence on domain Knowledge	Almost all detections depend on domain knowledge	Low, only the feature design Depends on domain knowledge
Detection performance	Low false alarm rate; High missed alarm rate	Low missed alarm rate; High false alarm rate

IV. MACHINE LEARNING METHOD (ML)

Machine learning is the study of algorithms that improve their performance with experience and are meant to computerize exercises; the machine takes every necessary step consummately furthermore in a maintained way. It is a type of artificial intelligence that provides computers with the ability to learn without being explicitly programmed (Yasir, H., Sugumaran, M. & Ludovic, J. 2017) [18].

ML Algorithms in IDS: Shallow machine learning models for IDS primarily include the artificial neural network (ANN), support vector machine (SVM), K-nearest neighbor (KNN), naïve Bayes, logistic regression (LR), decision tree, clustering, and combined and hybrid methods. Some of these methods have been studied for several decades, and their methodology is mature. They focus not only on the detection effect but also on practical problems, e.g., detection efficiency and data management.

C. Artificial Neural Network (ANN)

The design idea of an ANN is to mimic the way human brains work. An ANN contains an input layer, several hidden layers, and an output layer. The units in adjacent layers are fully connected. An ANN contains a huge number of units and can theoretically approximate arbitrary functions; hence, it has strong fitting ability, especially for nonlinear functions. Due to the complex model structure, training ANNs is time-consuming. It is noteworthy that ANN models are trained by the backpropagation algorithm that cannot be used to train deep networks.

D. Support Vector Machine (SVM).

The strategy in SVMs is to find a max-margin separation hyperplane in the n-dimension feature space. SVMs can achieve gratifying results even with small-scale training sets because the separation hyperplane is determined only by a small number of support vectors. However, SVMs are sensitive to noise near the hyperplane. SVMs are able to solve linear problems well. For nonlinear data, kernel functions are usually used. A kernel function maps the original space into a new space so that the original nonlinear data can be separated. Kernel tricks are widespread among both SVMs and other machine learning algorithms (Teng, S. Wu, N. Zhu, H. Teng, L. & Zhang, W. 2017) [19].

E. K-Nearest Neighbor (KNN)

The core idea of KNN is based on the manifold hypothesis. If most of a sample’s neighbors belong to the same class, the sample has a high probability of belonging to the class. Thus, the classification result is only related to the top-k nearest neighbors. The parameter k greatly influences the performance of KNN models. The smaller k is, the more complex the model is and the higher the risk of overfitting. Conversely, the larger k is, the simpler the model is and the weaker the fitting ability (Kuttranont, P. et al., 2017) [20].

F. Naïve Bayes

The Naïve Bayes algorithm is based on the conditional probability and the hypothesis of attribute independence. For every sample, the Naïve Bayes classifier calculates the conditional probabilities for different classes. The sample is classified into the maximum probability class (Fouladi, R.F. Kayatas, C.E. & Anarim, E. 2016) [21].

G. Decision tree

The decision tree algorithm classifies data using a series of rules. The model is tree like, which makes it interpretable. The decision tree algorithm can automatically exclude irrelevant and redundant features. The learning process includes feature selection, tree generation, and tree pruning. When training a decision tree model, the algorithm selects the most suitable features individually and generates child nodes from the root node.The decision tree is a basic classifier. Some advanced algorithms, such as the random forest and the extreme gradient boosting (XGBoost), consist of multiple decision trees (Goeschel, K., 2016) [22].

H. *Clustering*

Clustering is based on similarity theory, i.e., grouping highly similar data into the same clusters and grouping less-similar data into different clusters. Different from classification, clustering is a type of unsupervised learning. No prior knowledge or labeled data is needed for clustering algorithms; therefore, the data set requirements are relatively low. However, when using clustering algorithms to detect attacks, it is necessary to refer external information. K-means is a typical clustering algorithm, where K is the number of clusters and the means is the mean of attributes. The K-means algorithm uses distance as a similarity measure criterion. The shorter the distance between two data objects is, the more likely they are to be placed in the same cluster. The K-means algorithm adapts well to linear data, but its results on nonconvex data are not ideal. In addition, the K-means algorithm is sensitive to the initialization condition and the parameter K. Consequently, many repeated experiments must be run to set the proper parameter value (Peng, K.; Leung, V.C. & Huang, Q. 2018) [23].

I. *Ensembles and Hybrids*

Every individual classifier has strengths and shortcomings. A natural approach is to combine various weak classifiers to implement a strong classifier. Ensemble methods train multiple classifiers; then, the classifiers vote to obtain the final results. Hybrid methods are designed as many stages, in which each stage uses a classification model. Because ensemble and hybrid classifiers usually perform better than do single classifiers, an increasing number of researchers have begun to study ensemble and hybrid classifiers. The key points lie in selecting which classifiers to combine and how they are combined (Jabbar, M. Aluvalu, R. & Reddy, S. 2017) [24].

J. *J48 Classifier*

This classifier is designed to improve the implementation of the C.4.5 algorithm which is implemented by Ross Quilan (Quinlan, J. R. 2014) in 1993. The expected output based on this classifier is in the form of decision binary trees but with more stability between computation time and accuracy (Bhullar, M. S. & Kaur, A., 2012) [25]. Regarding to decision tree structure the leaf node had a decision of expected output.

K. *Random Tree Classifier*

RT is one of tree classifiers using this classifying the number of trees should be fixed before implementing. Each individual tree represents a single decision tree. Each individual tree has randomly selected attributes from dataset. Therefore, the random tree classifier could be considered as a finite group of decision trees. The procedure of predicting the entire dataset is to migrate several decision trees outputs and choose the winner expected class based on total numbers of votes (Cutler, A. & G. Zhao, 2001) [26].

V. BENCHMARK DATASET USED IN IDS

The chore duty of machine learning is to extract valuable information from data; hence, the performance of machine learning depends upon the quality of the input data. Understanding data is the foundation of machine learning methodology. For IDSs, the adopted data should be easy to get and imitate the behaviors of the hosts or networks. The common source data types for IDSs are packets, flow, sessions, and logs (Hongyu, L. & Bo, L. 2019) [32]. Building a dataset is complex and time-consuming. Constructed dataset can be reused repeatedly by many researchers.

VI. RESEARCH STATEMENT PROBLEM

Most research conducted for IDS are traditional (signature) based methods and expert rules based, these methods are not efficient and too tedious because it involves manual procedures making the methods not sufficient (Radford et al., 2018) [33] hence the introduction of machine learning techniques, here the procedures are automated. Even with introducing machine learning techniques most are trained on one single huge dataset, hence prone to be over-fitting when new types of attack are presented, also researchers have designed comprehensive machine learning systems for IDS but most of the machine learning algorithms designed does not ascertain more efficient ways to perform a more accurate classification on the datasets. Hence, the need for research in finding a model with higher detection accuracy is not supposed to be overemphasized.

Also, most of the researches conducted in this area uses only one dataset, there is also a need to use a hybridized dataset for testing IDS models.

VII. CONCLUSION AND FUTURE WORK

In this paper we first acknowledged the effort of researchers that have contributed immensely in this research area then we carefully discussed the solution techniques or methods used in developing IDS which includes signature-based and anomaly-based learning methods, then we looked at Machine Learning as a suitable technique for IDS and discussed different ML techniques and finally discussed different datasets benchmark used so far in IDS, we also give the research statement problem and presented the proposed methodology and architecture of the system. In future, we are going to develop an efficient machine learning IDS that will automatically detect attacks in network software and hardware based on the methodology described above.

TABLE II: Advantages and Disadvantages of different machine learning algorithms

Reference	**Alg.**	**Advantages**	**Disadvantages**	**Improvement Measures**
Kuang, F. Zhang, S. Jin, Z. & Xu,W.(2015) [27]	SVM	Learn useful information from small train set; Strong generation capability	Do not perform well on big data or multiple classification tasks; Sensitive to kernel function parameters	Optimized parameters by particle swarm optimization (PSO)
Syarif, A.R. & Gata, W.(2017) [28].	KNN	Apply to massive data; Suitable to nonlinear data; Train quickly; Robust to noise	Low accuracy on the minority class; Long test times; Sensitive to the parameter K	Reduced comparison times by trigonometric inequality; Optimized parameters by particle

				swarm optimization (PSO); Balanced datasets using the synthetic minority oversampling technique (SMOTE)
Mahmood, H.A.(2018) [29]	Naïve Bayes	Robust to noise; Able to learn incrementally	Do not perform well on attribute-related data	Imported latent variables to relax The independent assumption
Shah, R. Qian, Y. Kumar, D. Ali, M.& Alvi, M. (2017) [30]	LR	Simple, can be trained rapidly; Automatically scale features	Do not perform well on nonlinear data; Apt to overfitting	Imported regularization to avoid overfitting
	Decision Tree	Automatically select features; Strong interpretation	Classification result trends to majority class; Ignore the correlation of data	Balanced datasets with SMOTE; Introduced latent variables
Peng, K.; Leung, V.C. & Huang, Q. (2018) [31]	K-Means	Simple, can be trained rapidly; Strong scalability; Can fit to big data	Do not perform well on nonconvex data; Sensitive to initialization; Sensitive to the parameter K	Improved initialization method

TABLE III Summary of related papers, algorithms and datasets used

TITLE OF PAPER	ALGORITHM USED	DATASET USED
Machine Learning Techniques for Intrusion Detection: A Comparative Analysis		KDDCup99
An Intrusion Detection System Using Machine Learning Algorithm	Bayes Net, J48 and Random Forest	KDDCup99
Use of machine learning in big data analytics for insider threat detection.	SVM and K-means	Private dataset
False positive elimination in intrusion detection based on clustering.	Fuzzy C-means	DARPA 2000
TR-IDS: Anomaly-based intrusion detection through text-convolutional neural network and random forest	CNN(Convolutional Neural Network)	ISCX 2012
Deep-Full-Range: A Deep Learning Based Network Encrypted Traffic Classification and Intrusion Detection Framework	CNN, LSTM, and autoencoder	ISCX 2012
Network intrusion detection through stacking dilated convolutional autoencoders.	Autoencoder	CTU-UNB
Reducing false positives in intrusion detection systems using data-mining techniques utilizing9support vector machines, decision trees, and naive Bayes for off-line analysis	SVM, decision tree, and Naïve Bayes	KDD99
Parallel KNN and Neighborhood Classification Implementations on GPU for Network Intrusion Detection	KNN	KDD99
Clustering approach based on mini batch K-means for intrusion detection system over big data.	K-means	KDD99
Convolutional Neural Networks for Multi-class Intrusion Detection System.	CNN	NSL-KDD and UNSW-NB15
Network Intrusion Detection Based on Stacked Sparse Autoencoder and Binary Tree Ensemble Method.	Autoencoder and XGBoost	NSL-KDD
Deep Adversarial Learning in Intrusion Detection: A Data Augmentation Enhanced Framework.	GAN	KDD99
SVM-DT-based adaptive and collaborative intrusion detection.	SVM	KDD99
A hybrid spectral clustering and deep neural network ensemble algorithm for intrusion detection in sensor networks	DNN	KDD99 and NSL-KDD
A novel hierarchical intrusion detection system based on decision tree and rules-based models.	Decision tree	CICIDS 2017
Real-time anomaly-based distributed intrusion detection systems for advanced Metering Infrastructure utilizing stream data mining	K-means	Private dataset
HAST-IDS: Learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection	CNN	DARPA 1998 and ISCX 2012
Design of intelligent KNN-based alarm filter using knowledge-based alert verification in intrusion detection.	KNN	Private dataset

ACKNOWLEDGEMENT

We wish to thank research our supervisors Ass. Prof. A.Y Gital for his immense contribution and academic guidance towards the success of this research work.

REFERENCES

[1] Wu, W. Li, R. Xie, G. An, J. Bai, Y. Zhou, J. Li, K. (2019). A Survey of Intrusion Detection for In-Vehicle Networks, *IEEE Transactions on Intelligent Transportation Systems* PP (1) (2019) 1–15. doi:10.1109/TITS.2019.2908074.

[2] Thing, V. L. L. & Wu, J. (2016). Autonomous Vehicle Security: A Taxonomy of Attacks and Defences, in: Proceedings - 2016 IEEE International Conference on Internet of Things; IEEE Green Computing and Communications; IEEE Cyber, Physical, and Social Computing; IEEE Smart Data, iThings-GreenCom-CPSCom-Smart Data 2016, 2017, pp. 164–170. doi:10.1109/iThings-GreenCom-CPSCom-SmartData.2016. 52.

[3] Omer, M. (2019). Implementing security techniques to lower the probability of IoT- devices getting hacked (2019).

[4] Choudhary, S. & Kesswani, N. (2019). Analysis of KDD-Cup’99, NSL-KDD and UNSW-NB15 Datasets using Deep Learning in IoT, Procedia Computer Science 167 (2019) (2020) 1561–1573. doi:10.1016/j.procs.2020.03.367. URL https://doi.org/10.1016/j.procs.2020.03.367

[5] Anup K. G. Aaron, S. & Michael, S. (1999). Learning Program Behavior Profiles for Intrusion Detection.. In Workshop on Intrusion Detection and Network Monitoring, Vol. 51462. 1–13.

[6] Srinivas, M. Guadalupe, J. & Andrew, S. (2002). Intrusion detection: support vector machines and neural networks. In proceedings of the IEEE International Joint Conference on Neural Networks (ANNIE), St. Louis, MO. 1702–1707.

[7] Anup K. G. Aaron, S. & Michael, S. (1999). Learning Program Behavior Profiles for Intrusion Detection.. In Workshop on Intrusion Detection and Network Monitoring, Vol. 51462. 1–13.

[8] Jeremy, F. (1994). Artificial intelligence and intrusion detection: Current and future directions. In Proceedings of the 17th national computer security conference, Vol. 10. Baltimore, USA, 1–12.

[9] Mayhew, M. Atighetchi, M. Adler, A. & Greenstadt, R. (2015). Use of machine learning in big data analytics for insider threat detection. *In Proceedings of the MILCOM 2015-2015 IEEE Military Communications Conference*, Canberra, Australia, pp. 915–922.

[10] Hu, L. Li, T. Xie, N. & Hu, J. (2015). False positive elimination in intrusion detection based on clustering. In Proceedings of the 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Zhangjiajie, China, pp. 519–523.

[11] Yu, Y. Long, J. & Cai, Z. (2017). Network intrusion detection through stacking dilated convolutional autoencoders. *Secur. Commun. Netw*. 4184196.

[12] Liu, H.; Lang, B.; Liu, M.& Yan, H. (2019) CNN and RNN based payload classification methods for attack detection. Knowl.-Based Syst. **2019**, 163, 332–341. [CrossRef]

[13] Zeng, Y.; Gu, H.; Wei, W. & Guo, Y. (2019). Deep-Full-Range: A Deep Learning Based Network Encrypted Traffic Classification and Intrusion Detection Framework. *IEEE Access*, 7, 45182–45190.

[14] Rigaki, M. & Garcia, S. (2018). Bringing a gan to a knife-fight: Adapting malware communication to avoid detection. In Proceedings of the 2018 IEEE Security and PrivacyWorkshops (SPW), San Francisco, CA, USA, 24 May 2018; pp. 70–75.

[15] Philokypros, P. I. Vassilios G. V. Ioannis D. M. & Michael D. L.(2018). A Signature-based Intrusion Detection System for the Internet of Things. In*: Information and Communication Technology Form*. eprints@whiterose.ac.uk https://eprints.whiterose.ac.uk/

[16] Mohammad S. H, Abdul M and Abu N. B (2012): An Implementation of Intrusion Detection System Using Genetic Algorithm. International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.2.

[17] Hamid, Y. Sugumaran, M. & Balasaraswathi, V. (2016). IDS Using Machine Learning - Current State of Art and Future Directions," *British Journal of Applied Science & Technology*, vol. 15, no. 3, pp. 1–22.

[18] Yasir Hamid, M. Sugumaran & Ludovic Journaux (2017). Machine Learning Techniques for Intrusion Detection: A Comparative Analysis. DOI: http://dx.doi.org/10.1145/2980258.2980378

[19] Teng, S. Wu, N. Zhu, H. Teng, L. & Zhang, W. (2017). SVM-DT-based adaptive and collaborative intrusion detection. *IEEE/CAA J. Autom*. Sin.108–118. *Transmission Systems for Communications*, 3rd ed., Western Electric Co., Winston-Salem, NC, 1985, pp. 44–60.

[20] Kuttranont, P. Boonprakob, K. Phaudphut, C.; Permpol, S. Aimtongkhamand, P. KoKaew, &U.Waikham, (2017). B.So-In, C. Parallel KNN and Neighborhood Classification Implementations on GPU for Network Intrusion Detection. J. Telecommun. Electron. Comput. Eng. (JTEC), 9, 29–33.

[21] Fouladi, R.F.; Kayatas, C.E.; Anarim, E. (2016). Frequency based DDoS attack detection approach using naive Bayes classification. In Proceedings of the 2016 39th International Conference on Telecommunications and Signal Processing (TSP), Vienna, Austria, 27–29 June 2016; pp. 104–107.

[22] Goeschel, K (2016). Reducing false positives in intrusion detection systems using data-mining techniques utilizing support vector machines, decision trees, and naive Bayes for off-line analysis. *In Proceedings of the SoutheastCon* 2016, Norfolk, VA, USA, pp. 1–6.

[23] Peng, K.; Leung, V.C. & Huang, Q. (2018). Clustering approach based on mini batch kmeans for intrusion detection system over big data. IEEE Access **2018**, 6, 11897–11906. [CrossRef]

[24] Jabbar, M. Aluvalu, R. & Reddy, S. (2017). Cluster based ensemble classification for intrusion detection system. In Proceedings of the 9th International Conference on Machine Learning and Computing, Singapore, 24–26 February 2017; pp. 253–257.

[25] Bhullar, M. S. & Kaur, A. (2012). Use of data mining in education sector," in Proceedings of the World Congress on Engineering and Computer Science, vol. 1, pp. 24–26.

[26] Cutler, A. & G. Zhao, (2001). "Pert-perfect random tree ensembles," *Computing Science and Statistics*, vol. 33, pp. 490–497.

[27] Kuang, F. Zhang, S. Jin, Z. & Xu,W.(2015). A novel SVM by combining kernel principal scomponent analysis and improved chaotic particle swarm optimization for intrusion detection. Soft Comput. **2015**, 19, 1187–1199. [CrossRef]

[28] Syarif, A.R. & Gata, W.(2017). Intrusion detection system using hybrid binary PSO and K-nearest neighborhood algorithm. In Proceedings of the 2017 11th International Conference on Information & Communication Technology and System (ICTS), Surabaya, Indonesia, 31 October 2017; pp. 181–186.

[29] Mahmood, H.A.(2018). Network Intrusion Detection System (NIDS) in Cloud Environment based on Hidden Naïve Bayes Multiclass Classifier. Al-Mustansiriyah J. Sci. **2018**, 28, 134–142. [CrossRef]

[30] Shah, R. Qian, Y. Kumar, D. Ali, M.& Alvi, M. (2017). Network intrusion detection through discriminative feature selection by using sparse logistic regression. Future Internet **2017**, 9, 81. [CrossRef]

[31] Peng, K.; Leung, V.C. & Huang, Q. (2018). Clustering approach based on mini batch kmeans for intrusion detection system over big data. IEEE Access **2018**, 6, 11897–11906. [CrossRef]

[32] Hongyu, L. & Bo, L.(2019). Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey. State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China; doi:10.3390/app9204396. www.mdpi.com/journal/applsci

[33] Anish, H. A. & Sundarakantham, K. (2019). Machine Learning Based Intrusion Detection System. Proceedings of the Third International Conference on Trends in Electronics and Informatics (ICOEI 2019) IEEE Xplore Part Number: CFP19J32-ART; ISBN: 978-1-5386-9439-8.