JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Review on Advancements in Intrusion Detection Systems: A Comprehensive Review Unveiling the Complexity of Network Security

[1] Anvesha Mahto, [2]Ravikant Soni

[1]Research Scholar, [2]Assistant Professor
[1]Department of CSE Engineering,
[1]SSTC, Bhilai, India.

Abstract: Over the past decade, network security has emerged as a critical area of research due to the rapid evolution and proliferation of internet and communication technologies. The dynamic landscape of network security has propelled research in the past decade, fuelled by the rapid advancements in internet and communication technologies. Safeguarding networks and their digital assets within the vast cyberspace realm has become an intricate and perplexing challenge. Intrusion Detection Systems (IDS), including firewalls, anti-virus software, and network-based IDS (NIDS), have emerged as indispensable tools for countering malicious activities. However, the sheer magnitude of technological progress has led to sprawling networks, handling an ever-increasing array of applications, resulting in a deluge of critical data shared across nodes. This data's security has been compromised by the incessant emergence of new attacks, evolving mutations of existing threats, and the vulnerability of nearly every network node. Breaches in data integrity can have devastating consequences, inflicting irreparable damage to an organization's market reputation and incurring substantial financial losses. Despite continuous development and refinement, conventional IDSs struggle to detect complex attacks like zero-day exploits, while grappling with the challenge of minimizing false alarms. This comprehensive literature review critically assesses a broad range of cutting-edge research studies aimed at advancing IDS capabilities and mitigating existing limitations.

IndexTerms **–IDS, Network Security**

I. INTRODUCTION

In the past decade, network security has emerged as a crucial area of research due to the recent interest and advancements in the development of internet and communication technologies. It employs tools such as firewalls, anti-virus software, and intrusion detection systems (IDS) to protect the network and all of its assets within cyberspace. Among these, network-based intrusion detection system (NIDS) is the attack detection mechanism that provides the desired level of security by continuously monitoring network traffic for malicious or suspicious activity. Jim Anderson first proposed the concept of IDS in 1980. Since then, numerous IDS products have been developed and refined to meet network security requirements. However, the tremendous technological advancements of the past decade have resulted in a significant increase in network size and the number of applications handled by network nodes. As a result, a vast quantity of vital data is generated and shared across various network nodes. The security of these data and network nodes has become difficult due to the emergence of a large number of new attacks, either as a result of the mutation of an existing attack or the introduction of a new attack. Nearly every network node is susceptible to security threats. For example, the data node may be crucial to an organisation. Any breach of the node's information could have a devastating effect on the organization's market reputation and financial losses. Existing IDSs have demonstrated inadequacy in detecting a variety of attacks, such as zero-day attacks, and lowering false alarm rates (FAR). This eventually necessitates an NIDS that is efficient, accurate, and cost-effective in order to provide strong network security.

LITERATURE REVIEW

Lashkari et al. [1] employed padding as a means to reduce the dimensions of the tendon. The model was developed through a process of adjusting the hyperparameters until a point was reached where the performance of the model began to deteriorate. To avoid overfitting, the authors constructed a final model comprising of several dual convolutional layers, a pooling layer, and a dropout layer. This model consists of 10 classes, with nine for attacks and one for regular traffic. However, the model exhibits a significant disparity between the upper and lower classes, thereby necessitating the implementation of bootstrapping techniques. The authors employed the pre-partitioned UNSWNB15 dataset and a user-defined dataset, constituting 30% of the entire dataset, to evaluate their approach. Both datasets exhibited accuracy rates of 94.4% and 95.6%, respectively.

Chen et. al. [2] utilized of Convolutional Neural Networks (CNN) for the development of a secondary Intrusion Detection System (IDS). The response provided comprises of a dual component. One approach involves utilising Convolutional Neural Networks (CNNs) for offline training. The model commences with an initial layer of dimensions 9 by 9, followed by subsequent convolutional layers and a maximum pooling layer, resulting in a final output layer of dimensions 1 by 1. During the live monitoring phase of their system, the researchers utilise Suricata, an open-source Intrusion Detection System (IDS), to impede network traffic. Upon the completion of packet cleaning, the acquired model is subsequently employed to process network data, thereby yielding the identification outcome. The model was evaluated using the CICIDS2017 dataset. Both the feature dataset and the raw flow dataset were utilised for testing purposes. The model's precision scores of 96.55% and 99.56% indicate superior performance on actual traffic data compared to a feature-extracted dataset.

Gautam et al. [3] proposed a strategy termed "ensemble" for detecting invasion. Three tests were conducted to demonstrate the efficacy of the proposed approach in enhancing outcomes. Following the normalisation of the KDD Cup99 dataset, feature selection was performed using a correlation-based approach. Finally, the researchers employ an ensemble technique that incorporates three distinct algorithms (Naive Bayes, PART, and Adaptive Boost) to facilitate the feature selection procedure. The selection was determined by the amount of acquired information in the aforementioned procedure. The outcome is subsequently determined either by aggregating the outcomes of various formulas or by selecting the option that garners the highest number of adherents. Packing is utilised as a means to reduce the occurrence of arbitrary errors. The method employed by the researchers yielded a success rate of 99.9732% when applied to the KDD Cup99 dataset.

Al-Yaseen et al. [4] presented a novel approach for detecting attacks in systems through the utilisation of a mixed multi-level model that integrates Support Vector Machine (SVM) and Extreme Learning Machine (ELM) techniques. The proposed concept comprises a stratification of five distinct layers, whereby the initial layer serves to isolate Denial-of-Service (DoS) operations from any other network traffic. The subsequent tier categorises novel data into either Probe or Other. The third tier of classification categorises unidentified network traffic as either User to Root (U2R) attacks or Other, while the fourth tier sorts unidentified traffic as Remote to Local (R2L) attacks or Other. During the fifth stage, traffic that has an unknown predecessor is segregated from regular traffic. The reason for R2L and U2R being positioned at the lower end of the spectrum is due to their similarity to conventional links. At each stage, a forecaster is utilised. The model comprises of a single Extreme Learning Machine (ELM) classifier at level 2 and four Support Vector Machine (SVM) classifiers at levels 1, 3, 4, and 5. Given that ELM has demonstrated superior performance compared to SVM, the decision was made to employ an ELM predictor for the purpose of identifying the individual known as Probe. Following the preparation of the training set from the KDD dataset, a modified K-means algorithm was employed to extract five distinct clusters that can be identified by the proposed method. The approach employed by the researchers yielded a precision rate of 95.75%, slightly surpassing the precision rate of 95.57% that would have been achieved through the use of multi-level SVM in isolation. Furthermore, the mixed model exhibited a comparatively lower rate of false alarms in contrast to the multi-level support vector machine (SVM) approach, with a value of 2.17 percent.

Kanimozhi et al. [5] cloud attacks can be detected through the application of the opposing tunicate fuzzy C-means algorithm. The data underwent pre-processing and normalisation prior to partitioning into two distinct sets, one for training and the other for testing. Logistic regression was employed alongside the OPTSA and FCM grouping models to retain the salient features. The data set is partitioned into C clusters utilising the fuzzy C-means algorithm. Following the grouping of data, a cluster extension and merging procedure was conducted to eliminate redundant groups. The method was applied to multiple datasets, including CICIDS2017, and yielded an accuracy rate of 80%.

Chen et al. [6] employed the random forest algorithm to develop an intrusion detection system for cellular networks. Initially, a model was developed to identify aggressive nonlinear scrambling entry signals, which was designed as a signal recognition model to capture the salient features of the signals. The study employed an improved random forest approach to extract the spectral characteristics of a hazardous signal. Additionally, a combination of reinforcement learning and static feature fusion techniques were utilised to identify the most detrimental traffic in a wireless network. The average accuracy rate was 96.93%.
Jabez et al. [7] developed a system that employs the technique of "outlier detection" to identify potential hazards that are not yet known. The methodology of outlier discovery relies on identifying individual data points that deviate from the normative distribution of a given dataset. This approach employs the "neighbourhood misfit factor" to identify distant locations. The KDDcup99 datasets were utilised to evaluate the efficacy of their approach. The primary advantage of their system lies in its rapidity, surpassing alternative techniques such as the computationally demanding back propagation neural network.

Kurniawan et al. [8] proposed a novel approach to enhance Intrusion Detection Systems (IDS) through the utilisation of Support Vector Machines (SVM). The method employed by the researchers involved the utilisation of the Naive Bayes algorithm for feature identification. Subsequently, the model underwent training using the modified data resulting from the feature selection process. The researchers employed the UNSW-NB15 and CICIDS2017 datasets to evaluate their proposed concept. The utilisation of Naive Bayes as a feature identifier prior to employing the SVM classifier yields superior outcomes compared to the utilisation of the SVM classifier in isolation. The accuracy rate on the UNSW-NB15 dataset was 93.75%. The CICIDS2017 dataset exhibited an accuracy rate of approximately 98.92 percent. However, the approach employed by them solely enables the identification of the occurrence of an invasion, without providing any insight into the nature or type of the attack.

Chauhan et al. [9] proposed a methodology for detecting intrusions in wireless networks. The system was constructed utilising cloud technology to optimise computational resources. The utilisation of sink nodes within the fog infrastructure has been implemented to alleviate a portion of the computational burden from the cloud computing component. The researchers employed Polymorphic Mutation (PM) and Compact SCA (CSCA) techniques to develop a solution that is optimised for minimal weight. The utilisation of probability by CSCA resulted in a reduction of data density, thereby alleviating the computational burden. The incorporation of polymorphic evolution was implemented as a measure to mitigate the reduction in precision that is associated with the use of CSCA.

The Particle Swarm Optimisation Monte Carlo Simulated Annealing (PMCSCA) technique was employed to determine the optimal configuration for the parameters of the K-Nearest Neighbours (KNN) algorithm. The authors employed the NSL-KDD and UNSW-NB15 datasets to evaluate their approach. They exhibited a level of accuracy of 99.327% and 98.27%, respectively.

Gu et al. [10] was founded on the use of Convolutional Neural Networks (CNN). Initially, Auto-Encoder (AE) and Principal Component Analysis (PCA) were employed to extract features. The AutoEncoder technique involves the utilisation of multiple layers of neural networks to eliminate redundant data and reduce the dimensionality of the input data. Subsequently, the data was transformed from a unidimensional format to a bidimensional format, and the resultant training matrix was transmitted to the Convolutional Neural Network (CNN) model. Backpropagation techniques are employed to instruct and enhance the model. The KDDcup99 was employed to evaluate the efficacy of their model, and the findings indicated a 94% accuracy rate. Upon comparison with DNN and RNN models, it was observed that the outcomes of their model exhibited a slight improvement. However, the model's efficacy in detecting U2R and R2L, which are infrequent occurrences within the cohort, is limited.

Pan et al. [11] introduced a multi-layer model for the purpose of detecting attacks. The GcForest and CNN machine learning techniques were employed in their methodology. The GcForest algorithm is a type of random forest technique that involves the construction of decision trees in a hierarchical manner. The model comprises two primary components. The initial segment of the study employs a CNN algorithm to analyse the incoming data and differentiate between regular flow and hazardous activity. The CNN approach employed by the authors is founded on a modified iteration of GoogLeNet, denoted as GoogLeNetNP. The subsequent phase involves the expansion of attack subclasses through the utilisation of a deep forest as a reference model. The response provided by the individual is deemed more precise as their approach involves the segregation of odd categories into N-1 subclasses within the second layer. The subsequent stratum employs the gcForest cascade methodology, albeit with XGBoost as opposed to random forest. XGBoost shares similarities with a random forest algorithm, however, it differs in that the trees are constructed sequentially until the objective function is optimised. The authors employed the UNSW-NB15 and CICIDS2017 datasets to evaluate their proposed approach. The combined accuracy of all the methods utilised was 99.24%, in contrast to the individual accuracies of each method.

Xiao et al. [12] have proposed FSL technique. The FSL technique is a form of deep learning that is capable of acquiring knowledge with minimal or absent prior information. The respondent employed CNN and DNN embedding models to extract features in their response. The utilisation of these models facilitates the reduction of data dimensions while preserving crucial information. The researchers employed the UNSW-NB15 and NSL-KDD datasets for the purpose of evaluating their model. The accuracy rate of their responses was 92.34% and 92% correspondingly.

Zhang et al. [13], proposed the utilization of an intrusion detection system (IDS) based on machine learning ensemble. The technique of principal component analysis was employed to extract characteristics. The ensemble technique employed by the authors incorporates a combination of various machine learning algorithms, namely Decision Tree, Random Forest, K-Nearest Neighbours, Deep Neural Network, and MultiTree. This is predicated on numerous evaluations conducted on the NSL-KDD datasets. The ensemble algorithm employs a weighted approach to enhance accuracy, and the outcomes are determined by a majority vote. The accuracy rate of their predictions was 85.2%, surpassing the accuracy rate that would have been achieved by utilizing a single programmer. However, the efficacy of their approach is limited when it comes to analyzing dispersed threats.

Yu et al. [14] proposed a methodology that employs a Deep Belief Network (DBN) in conjunction with a multi-SVM technique. A Deep Belief Network (DBN) is formed by combining multiple unregulated networks, such as Restricted Boltzmann Machines (RBM). A Restricted Boltzmann Machine (RBM) is a type of neural network architecture that comprises an input layer and a hidden layer, where the nodes in the latter are sparsely connected to the levels immediately above and below them. The approach employed by DBN involves the utilisation of a greedy layer-wise architecture for the purpose of unstructured pre-training. The acquisition of the most significant characteristics is accomplished via a technique known as "supervised fine-tuning." The utilisation of Deep Belief Networks (DBN) is employed to identify advantages in support of their response. Subsequently, the identified traits are incorporated into a Support Vector Machine (SVM) that comprises multiple layers. The outcome will be determined by a voting mechanism. The researchers employed the KDDcup99, NSL-KDD, UNSW-NB15, and CICIDS2017 datasets to evaluate the efficacy of their proposed approach. The obtained accuracy rates were 94.76%, 97.27%, 90.47%, and 90.40%, in that order. Research has demonstrated that employing a greater number of levels results in expedited response times.

Gao et al. [15], employed an optimization technique to enhance the efficiency of Deep Belief Networks (DBN) for Intrusion Detection Systems (IDS). The authors employed the Artificial Fish Swam Algorithm (AFSA), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO) methodologies to enhance their model. Initially, the utilization of AFSA is aimed at enhancing the performance of PSO. The optimal solution for the initial particle search can be obtained through the utilization of Genetic Algorithm (GA). Subsequently, the DBN model's precision is enhanced through utilization of the optimal solution. The NSL-KDD dataset was utilized to evaluate the efficacy of their approach, yielding a success rate of 82.36 percent.

Marir et al. [16], proposed a strategy for an intrusion detection system (IDS) that employs deep neural networks. The DNN architecture comprises of a single input layer, five hidden layers, and a sole output layer. A flexible approach is provided for deep neural network models, allowing for the utilisation of a range of one to five hidden layers. The IT system utilised in the study was Apache Spark. The response provided is applicable in scenarios involving both Host-based Intrusion Detection Systems (HIDS) and Network-based Intrusion Detection Systems (NIDS). The NIDS employed various datasets, namely KDDcup99, NSL-KDD, Kyoto, UNSW-NB15, and CICIDS2017, to evaluate their approach. The accuracy rates for each number of deep neural network (DNN) layers were 93%, 79.42%, 87.78%, 76.48%, and 94.55%.

Wei et al. [17] proposed a solution combining Non-Symmetric Deep Auto-Encoder (NDAE) and Random Forest. Usually, an auto-encoder uses the symmetric scheme from encoder-decoder, however, in their solution, they only used the encoding phase. It reduces the computational time without impacting too much on the accuracy of the IDS. To handle complex datasets, they choose to stack their NDAE. However, they discovered that using only NDAE was not enough to have an accurate classification. Therefore, they added Random Forest as their classifier after performing feature extraction using two NDAE with three hidden layers each. They tested their solution on the KDDcup99 and NSL-KDD datasets and compared it to a DBN solution. They obtained, respectively a total accuracy of 97.85% and 85.42%. However, their solution struggles to detect small classes such as R2L and U2R.

Vinayakumar et al. [18] demonstrated the effect of feature extraction using a Stacked Sparse Auto Encoder (SSAE). An auto encoder that utilizes a sparsity penalty is referred known as a sparse auto-encoder. Typically, the penalty is applied when hidden nodes are employed. Because of this, adopting a sparse auto-encoder minimizes the utilization of hidden nodes. A stacked sparse auto-encoder is one that has additional sparse auto-encoders added to it. It enables dimensional reduction of the supplied data without considerable information loss. The NSL-KDD dataset was used to test their SSAE model, and the error back propagation approach was utilized to optimize it. To demonstrate the extent to which using SSAE for feature extraction increases accuracy, they employed several classifiers both with and without their SSAE model. Combining the SSAE and SVM classifiers yielded the highest accuracy. They achieved a 99.35% total accuracy. One of the key benefits of choosing their solution is the significant time savings for training and testing, which takes around one-tenth as long as alternative solutions. R2L and U2R, however, have a lower detection rate when compared to the other classes.

Shone et al. [19], suggested a two-stage deep learning model (TSDL). They initially assign a probability value to the traffic and categorise it as either normal or abnormal. They employed a DNN technique for both phases, using a Deep stacked auto-encoder (DSAE) for feature extraction and Soft-max as a classifier. In the second step, they used this value as an additional feature to train the classifier. For multi-class classification issues, neural networks often adopt soft-max. They used the KDDcup99 and UNSW-NB15 datasets to test their solution. They each achieved a total accuracy of 99.996% and 89.134%, respectively.

Yan et al. [20] combined an unsupervised technique with two auto-encoders and a supervised stage. They individually trained the two autoencoders using both regular and attack traffic. The samples are then rebuilt by the auto-encoders and added to the dataset used to train the model. The dataset is processed using a CNN in one dimension. This is done to better distinguish between the two classes—normal and attack—by observing the effects of one channel on the other. Lastly, they used a Soft-max classifier to determine if the data represented an assault or was regular. On the KDDcup99, UNSW-NB15, and CICIDS2017 datasets, they tested their model. They each achieved total accuracy results of 92.49%, 93.40%, and 97.90%. Their method lacks information about the various attack kinds, which is one of its flaws.

Khan et al. [21], proposed a model using Fast Learning Network (FLN) based on particle swarm optimization (PSO). Due to the inefficient nature of the neural network's weights, PSO was used to improve FLN's accuracy. On the KDDcup99 dataset, their solution was compared to other FLN methods. They were more precise than other solutions at detecting the distinct classes. Overall, they achieved 89.23% accuracy. However, their weak accuracy when identifying one of the few assault classes (R2L) lowers their overall accuracy.

V. CONCLUSION

Several research studies have explored different approaches to improve the performance of Intrusion Detection Systems (IDS). These include solutions like Deep Belief Networks (DBNs) combined with ensemble SVMs for feature extraction and classification, optimizing algorithms like Particle Swarm Optimization (PSO) applied to DBNs, scalable Deep Neural Network (DNN) architectures with flexible hidden layer configurations, combinations of Non-symmetric Deep Auto-Encoders (NDAEs) and Random Forest classifiers, feature extraction using Stacked Sparse Auto-Encoders (SSAEs) leading to reduced training time, Two-Stage Deep Learning (TSDL) models incorporating Deep stacked auto-encoders (DSAEs) and Soft-max classifiers, unsupervised approaches utilizing auto-encoders and one-dimensional CNNs, and Fast Learning Networks (FLNs) optimized with particle swarm optimization (PSO). These solutions have demonstrated varying degrees of success on different datasets, showing improved accuracy and reduced computational time. However, challenges remain, such as the detection of small attack classes and providing detailed information on attack types. Continued research is necessary to develop robust and efficient IDS capable of effectively detecting and mitigating diverse cyber threats.

REFERENCES

[1]. Sharafaldin, I.; Lashkari, A.H.; Ghorbani, A. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization; Canadian Institute for Cybersecurity (CIC): Fredericton, NB, Canada, 2018; pp. 108–116.

[2]. Chen, L.; Kuang, X.; Xu, A.; Suo, S.; Yang, Y. A Novel Network Intrusion Detection System Based on CNN. In Proceedings of the 2020 Eighth International Conference on Advanced Cloud and Big Data (CBD), Taiyuan, China, 5–6 December 2020; pp. 243–247. [CrossRef]

[3]. Gautam, R.K.S.; Doegar, E.A. An Ensemble Approach for Intrusion Detection System Using Machine Learning Algorithms. In Proceedings of the 2018 8th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 11–12 January 2018; pp. 14–15. [CrossRef]

[4]. Al-Yaseen, W.L.; Othman, Z.A.; Nazri, M.Z.A. Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system. Expert Syst. Appl. 2017, 67, 296–303. [CrossRef]

[5]. Kanimozhi, P.; Victoire, T.A.A. Oppositional tunicate fuzzy C-means algorithm and logistic regression for intrusion detection on cloud. Concurr. Comput. Pract. Exp. 2022, 34, e6624. [CrossRef]

[6]. Chen, Y.; Yuan, F. Dynamic detection of malicious intrusion in wireless network based on improved random forest algorithm. In Proceedings of the 2022 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC), Dalian, China, 14–16 April 2022; pp. 27–32. [CrossRef]

[7]. Jabez, J.; Muthukumar, B. Intrusion Detection System (IDS): Anomaly Detection Using Outlier Detection Approach. Procedia Comput. Sci. 2015, 48, 338–346. [CrossRef]

[8]. Kurniawan, Y.; Razi, F.; Nofiyati, N.; Wijayanto, B.; Hidayat, M. Naive Bayes modification for intrusion detection system classification with zero probability. Bull. Electr. Eng. Inform. 2021, 10, 2751–2758. [CrossRef]

[9]. Chauhan, N. Naïve Bayes Algorithm: Everything You Need to Know. Available online: https://www.kdnuggets.com/2020/06/ naive-bayes-algorithm-everything.html#:~{}:text=One%20of%20the%20disadvantages%20of,all%20the%20probabilities%20 are%20multiplied (accessed on 9 September 2022).

[10]. Gu, J.; Lu, S. An effective intrusion detection approach using SVM with naïve Bayes feature embedding. Comput. Secur. 2021, 103, 102158. [CrossRef]

[11]. Pan, J.-S.; Fan, F.; Chu, S.C.; Zhao, H.; Liu, G. A Lightweight Intelligent Intrusion Detection Model for Wireless Sensor Networks. Secur. Commun. Networks 2021, 2021, 1–15. [CrossRef]

[12]. Xiao, Y.; Xing, C.; Zhang, T.; Zhao, Z. An Intrusion Detection Model Based on Feature Reduction and Convolutional Neural Networks. IEEE Access 2019, 7, 42210–42219. [CrossRef]

[13]. Zhang, X.; Chen, J.; Zhou, Y.; Han, L.; Lin, J. A Multiple-Layer Representation Learning Model for Network-Based Attack Detection. IEEE Access 2019, 7, 91992–92008. [CrossRef]

[14]. Yu, Y.; Bian, N. An Intrusion Detection Method Using Few-Shot Learning. IEEE Access 2020, 8, 49730–49740. [CrossRef]

[15]. Gao, X.; Shan, C.; Hu, C.; Niu, Z.; Liu, Z. An Adaptive Ensemble Machine Learning Model for Intrusion Detection. IEEE Access 2019, 7, 82512–82521. [CrossRef]

[16]. Marir, N.; Wang, H.; Feng, G.; Li, B.; Jia, M. Distributed Abnormal Behavior Detection Approach Based on Deep Belief Network and Ensemble SVM Using Spark. IEEE Access 2018, 6, 59657–59671. [CrossRef]

[17]. Wei, P.; Li, Y.; Zhang, Z.; Hu, T.; Li, Z.; Liu, D. An Optimization Method for Intrusion Detection Classification Model Based on Deep Belief Network. IEEE Access 2019, 7, 87593–87605. [CrossRef]

[18]. Vinayakumar, R.; Alazab, M.; Soman, K.P.; Poornachandran, P.; Al-Nemrat, A.; Venkatraman, S. Deep Learning Approach for Intelligent Intrusion Detection System. IEEE Access 2019, 7, 41525–41550. [CrossRef]

[19]. Shone, N.; Ngoc, T.N.; Phai, V.D.; Shi, Q. A Deep Learning Approach to Network Intrusion Detection. IEEE Trans. Emerg. Top. Comput. Intell. 2018, 2, 41–50. [CrossRef]

[20]. Yan, B.; Han, G. Effective Feature Extraction via Stacked Sparse Autoencoder to Improve Intrusion Detection System. IEEE Access 2018, 6, 41238–41248. [CrossRef]

[21]. Khan, F.A.; Gumaei, A.; Derhab, A.; Hussain, A. A Novel Two-Stage Deep Learning Model for Efficient Network Intrusion Detection. IEEE Access 2019, 7, 30373–30385. [CrossRef]