JOURNAL OF CRITICAL REVIEWS

ISSN- 2394-5125 VOL 08, ISSUE 05, 2021

Intelligent intrusion Detection system using Deep Learning Approach

Nusrath Mohammad. Pavankumar Thummeti , Mahesh Kumar Singirikonda

Department of Computer Science and Engineering

Sree Dattha Group of Institutions, Hyderabad, Telangana, India.

ABSTRACT

Machine learning techniques are being widely used to develop an intrusion detection system (IDS) for detecting and classifying cyber-attacks at the network-level and host-level in a timely and automatic manner. However, many challenges arise since malicious attacks are continually changing and are occurring in very large volumes requiring a scalable solution. There are different malware datasets available publicly for further research by cyber security community. However, no existing study has shown the detailed analysis of the performance of various machine learning algorithms on various publicly available datasets. Due to the dynamic nature of malware with continuously changing attacking methods, the malware datasets available publicly are to be updated systematically and benchmarked. In this paper, deep neural network (DNN), a type of deep learning model is explored to develop a flexible and effective IDS to detect and classify unforeseen and unpredictable cyber-attacks. The continuous change in network behaviour and rapid evolution of attacks makes it necessary to evaluate various datasets which are generated over the years through static and dynamic approaches. This type of study facilitates to identify the best algorithm which can effectively work in detecting future cyber-attacks. A comprehensive evaluation of experiments of DNNs and other classical machine learning classifiers are shown on various publicly available benchmark malware datasets. Our DNN model learns the abstract and high dimensional feature representation of the IDS data by passing them into many hidden layers. Through a rigorous experimental testing it is confirmed that DNNs perform well in comparison to the classical machine learning classifiers. Finally, we propose a highly scalable and hybrid DNNs framework called Scale-Hybrid-IDS-AlertNet (SHIA) which can be used in real time to effectively monitor the network traffic and host-level events to proactively alert possible cyber-attacks.

Keywords: Machine Learning, Cyber Attacks, Cyber security, Deep Neural Network.

1. INTRODUCTION

Information and communications technology (ICT) systems and networks handle various sensitive user data that are prone by various attacks from both internal and external intruders. These attacks can be manual, and machine generated, diverse and are gradually advancing in obfuscations resulting in undetected data breaches. For instance, the Yahoo data breach had caused a loss of $350M and Bitcoin breach resulted in a rough estimate of $70M loss. Such cyberattacks are constantly evolving with very sophisticated algorithms with the advancement of hardware, software, and network topologies including the recent developments in the Internet of Things (IoT). Malicious cyber-attacks pose serious security issues that demand the need for a novel, flexible and more reliable intrusion detection system (IDS). An IDS is a proactive intrusion detection tool used to detect and classify intrusions, attacks, or violations of the security policies automatically at network-level and host-level infrastructure in a timely manner. Based on intrusive behaviors, intrusion detection is classified into network-based intrusion detection system (NIDS) and host-based intrusion detection system (HIDS). An IDS system which uses network behaviour is called as NIDS. The network behaviors are collected using network equipment via mirroring by networking devices, such as switches, routers, and network taps and analyzed to identify attacks and possible threats concealed within in network traffic. An IDS system which uses system activities in the form of various log files running on the local host computer to detect attacks is called as HIDS. The log files are collected via local sensors. While NIDS inspects each packet contents in network traffic flows, HIDS relies on the information of log files which includes sensors logs, system logs, software logs, file systems, disk resources, users account information and others of each system.

Many organizations use a hybrid of both NIDS and HIDS. Analysis of network traffic flows is done using misuse detection, anomaly detection and stateful protocol analysis. Misuse detection uses predefined signatures and filters to detect the attacks. It relies on human inputs to constantly update the signature database. This method is accurate in finding the known attacks but is completely ineffective in the case of unknown attacks. Anomaly detection uses heuristic mechanisms to find the unknown malicious activities. In most of the scenarios, anomaly detection produces a high false positive rate. To combat this problem, most organizations use the combination of both the misuse and anomaly detection in their commercial solution systems. Stateful protocol analysis is most powerful in comparison to the detection methods due to the fact that stateful protocol analysis acts on the network layer, application layer and transport layer. This uses the predefined vendors specification settings to detect the deviations of appropriate protocols and applications.

Though deep learning approaches are being considered more recently to enhance the intelligence of such intrusion detection techniques, there is a lack of study to benchmark such machine learning algorithms with publicly available datasets. The most common issues in the existing solutions based on machine learning models are: firstly, the models produce high false positive rate with wider range of attacks; secondly, the models are not generalizable as existing studies have mainly used only a single dataset to report the performance of the machine learning model; thirdly, the models studied so far have completely unseen today's huge network traffic; and finally the solutions are required to persevere today's rapidly increasing high-speed network size, speed and dynamics. These challenges form the prime motivation for this work with a research focus on evaluating the efficacy of various classical machine learning classifiers and deep neural networks (DNNs) applied to NIDS and HIDS.

This work assumes the following

- An attacker aims at pretense as normal user to remain hidden from the IDS. However, the patterns of intrusive behaviors differ in some aspect. This is due to the specific objective of an attacker for example getting an unauthorized access to computer and network resources.
- The usage pattern of network resources can be captured; however, the existing methods ends up in high false positive rate.
- The patterns of intrusions exist in normal traffic with a very low profile over long-time interval.

Overall, this work has made the following contributions to the cyber security domain

- By combining both NIDS and HIDS collaboratively, an effective deep learning approach is proposed by modelling a deep neural network (DNN) to detect cyberattacks proactively. In this study, the efficacy of various classical machine learning algorithms and DNNs are evaluated on various NIDS and HIDS datasets in identifying whether network traffic behaviour is either normal or abnormal due to an attack that can be classified into corresponding attack categories.
- The advanced text representation methods of natural language processing (NLP) are explored with host-level events, i.e., system calls with the aim to capture the contextual and semantic

similarity and to preserve the sequence information of system calls. The comparative performance of these methods is conducted with the ADFA-LD and ADFA-WD datasets.

- This study uses various benchmark datasets to conduct a comparative experimentation. This is mainly due to the reason that each dataset suffers from various issues such as data corruptions, traffic variety, inconsistencies, out of date and contemporary attacks.
- A scalable hybrid intrusion detection framework called SHIA is introduced to process large amount of network level and host-level events to automatically identify malicious characteristics to provide appropriate alerts to the network admin. The proposed framework is highly scalable on commodity hardware server and by joining additional computing resources to the existing framework, the performance can be further enhanced to handle big data in real-time systems.

The code and detailed results are made publicly available [7] for further research. The remainder of the chapter is organized as follows. Section II discusses various stages of compromise according to attackers' perspective. Section III discusses the related works of similar research work done to NIDS and HIDS. Information of scalable framework, the mathematical details of DNNs and text representation methods for intrusion detection is placed in Section IV. Section V includes information related to major shortcomings of IDS datasets, problem formulation and statistical measures. Section VI includes description of datasets. Section VII and Section VIII includes experimental analysis and a brief overview of proposed system and architecture design. Section IX presents the experimental results. Conclusion, future work directions and discussions are placed in Section X.

2. LITERATURE SURVEY

The research on security issues relating to NIDS and HIDS exists since the birth of computer architectures. In recent days, applying machine learning based solutions to NIDS and HIDS is of prime interest among security researchers and specialists. A detailed survey on existing machine learning based solutions is discussed in detail by [5]. This section discusses the panorama of largest study to date that explores the field of machine learning and deep learning approaches applied to enhance NIDS and HIDS.

A. NETWORK-BASED INTRUSION DETECTION SYSTEMS (NIDS)

Commercial NIDS primarily use either statistical measures or computed thresholds on feature sets such as packet length, inter-arrival time, flow size and other network traffic parameters to effectively model them within a specific time window [6]. They suffer from high rate of false positive and false negative alerts. A high rate of false negative alerts indicates that the NIDS could fail to detect attacks more frequently, and a high rate of false positive alerts means the NIDS could unnecessarily alert when no attack is taking place. Hence, these commercial solutions are ineffective for present day attacks.

Self-learning system is one of the effective methods to deal with the present-day attacks. This uses supervised, semi-supervised and unsupervised mechanisms of machine learning to learn the patterns of various normal and malicious activities with a large corpus of Normal and Attack network and host-level events. Though various machine learning based solutions are found in the literature, the applicability to commercial systems is in early stages [9]. The existing machine learning based solutions outputs high false positive rate with high computational cost [3]. This is because machine learning classifiers learn the characteristic of simple TCP/IP features locally. Deep learning is a complex subnet of machine learning that learns hierarchical feature representations and hidden sequential relationships by passing the TCP/IP information on several hidden layers. Deep learning has achieved significant results in long standing Artificial intelligence (AI) tasks in the field of image processing, speech recognition, natural language processing (NLP) and many others [10]. Additionally, these performances have been transformed to various cyber security tasks such as intrusion detection, android malware classification, traffic analysis, network traffic prediction, ransomware detection, encrypted text categorization, malicious URL detection, anomaly detection, and malicious domain name detection [11]. This work focuses towards analyzing the effectiveness of various classical machine learning and deep neural networks (DNNs) for NIDS with the publicly available network-based intrusion datasets such as KDDCup 99, NSL-KDD, Kyoto, UNSW-NB15, WSN-DS and CICIDS 2017.

A large study of academic research used the de facto standard benchmark data, KDDCup 99 to improve the efficacy of intrusion detection rate. KDDCup 99 was used for the third International Knowledge Discovery and Data Mining Tools Competition and the data was created as the processed form of tcpdump data of the 1998 DARPA intrusion detection (ID) evaluation network. The aim of the contest was to create a predictive model to classify the network connections into two classes: Normal or Attack. Attacks were categorized into denial of service ('DoS'), 'Probe', remote-to-local ('R2L'), user-to-root ('U2R') categories. The mining audit data for automated models for ID (MADAMID) was used as feature construction framework in KDDCup 99 competition [17]. MADAMID outputs 41 features: first 9 features are basic features of a packet, 10-22 are content features, 23-31 are traffic features, and 32-41 are host-based features. The choices of available datasets are: (1) full dataset and (2) complementary 10% data. The detailed evaluation results of KDDCup 98 and KDDCup 99 challenge was published in [3]. Totally, 24 entries were submitted in the KDDCup 98, in that 3 winning entries used variants of decision tree to whom they showed only the marginal statistics significance in performance. The 9th winning entry in the contest used the 1-nearest neighbor classifier. The first significant difference in performance was found between 17th and 18th entries. This inferred that the first 17 submissions methods were robust and were profiled by [3]. The Third International Knowledge Discovery and Data Mining Tools Competition task remained as a baseline work and after this contest many machine learning solutions have been found. Most of the published results took only the 10% data of training and testing and few of them used custom-built datasets. Recently, a comprehensive literature survey on machine learning based ID with KDDCup 99 dataset was conducted [18]. After the challenge, most of the published results of KDDCup 99 have used several feature engineering methods for dimensionality reduction [18]. While few studies employed custom-built datasets, majority used the same dataset for newly available machine learning classifiers [18]. These published results are partially comparable to the results of the KDDCup 99 contest.

In [19], the classification model consists of two-stages: i) P-rules stage to predict the presence of the class, and ii) N-rules stage to predict the absence of the class. This performed well in comparison with the KDDCup 99 results except for the user-to-root ('U2R') category. In [20], the significance of feature relevance analysis was investigated for IDS with the most widely used dataset, KDDCup 99. For each feature they were able to express the feature relevance in terms of information gain. In addition, they presented the most relevant features for each class label. Reference [21] discussed random forest techniques in misuse detection by learning patterns of intrusions, anomaly detection with outlier detection mechanism, and hybrid detection by combining both the misuse and anomaly detection. They reported that the misuse approach worked better than winning entries of KDDCup 99 challenge results, and in addition anomaly detection worked better compared to other published unsupervised anomaly detection methods. Overall, it was concluded that the hybrid system enhances the performance with the advantage of combining both the misuse and anomaly detection approaches [22], [23], [72]. In [24], an ID algorithm using AdaBoost technique was proposed that used decision stumps as weak classifiers. Their system performed better than other published results with a lower false alarm rate, a higher detection rate, and a computationally faster algorithm. However, the drawback is that it failed to adopt the incremental learning approach. In [25], the performance of the shared nearest neighbor (SNN) based model in ID was studied and reported as the best algorithm with a high detection rate. With the reduced dataset they were able to conclude that SNN performed well in comparison to the K-means for ’U2R’ attack category. However, their work failed to show the results on the entire testing dataset.

In [26], Bayesian networks for ID was explored using Naive Bayesian networks with a root node to represent a class of a connection and leaf nodes to represent features of a connection. Later, [27] investigated the application of Naive Bayes network to ID and through detailed experimental analysis, they showed that Bayesian networks performed equally well and sometimes even better in ’U2R’ and ’Probe’ categories in comparison with the winning entries of KDDCup 99 challenge. In [28], a non-parametric density estimation method based on Parzen-window estimators was studied with Gaussian kernels and Normal distribution. Without the intrusion data, their system was comparatively favorable to the existing winning entries that was based on ensemble of decision trees. In [29], a genetic algorithm-based NIDS was proposed that facilitates to model both temporal and spatial information to identify complex anomalous behavior. An overview of ensemble learning techniques for ID was given in [30], and swarm intelligence techniques for ID using ant colony optimization, ant colony clustering and particle swarm optimization of systems were studied in [31]. A comparative study in such research works show that the descriptive statistics was predominantly used.

Overall, a comprehensive literature review shows very few studies use modern deep learning approaches for NIDS and the commonly used benchmark datasets for experimental analysis are KDDCup 99 and NSL-KDD [3], [32]– [34]. The IDS based on recurrent neural network (RNN) outperformed other classical machine learning classifiers in identifying intrusion and intrusion type on the NSL-KDD dataset [32]. Two level approach proposed for IDS in which the first level extracts the optimal features using sparse autoencoder in an unsupervised way and classified using softmax regression [33]. The application of stacked autoencoder was proposed for optimal feature extraction in an unsupervised way where the proposed method is completely non-symmetric, and classification was done using Random Forest. Novel long short-term memory (LSTM) architecture was proposed and by modeling the network traffic information in time series obtained better performance. The proposed method performed well compared to all the existing methods and as well as KDDCup 98 and 99 challenge entries [3]. The performance of various RNN types were evaluated by [34]. Various deep learning architectures and classical machine learning algorithms were evaluated for anomaly-based ID on NSL-KDD dataset [74]. The configuration of SVM was formulated as bi-objective optimization problem and solved using hyper-heuristic framework. The performance was evaluated for malware and anomaly ID. The proposed framework is very suitable for big data cyber security problems [75]. To enhance the anomaly-based ID rate, the spatial and temporal features were extracted using convolutional neural network and long short-term memory architecture. The performance was shown on both KDDCup 99 and ISCX 2012 datasets [76]. Two step attack detection method was proposed along with a secure communication protocol for big data systems to identify insider attack. In the first step, process profiling was done independently at each node and in second step using hash matching and consensus, process matching was done [77]. An online detection and estimation method was proposed for smart grid system [78]. The method specifically designed for identifying false data injection and jamming attacks in real-time and additionally provides online estimates of the unknown and time-varying attack parameters and recovered state estimates [78]. A scalable framework for ID over vehicular ad hoc network was proposed. The framework uses distributed machine learning i.e. alternating direction method of multipliers (ADMM) to train a machine learning model in a distributed way to learn whether an activity normal or attack [79].

B. HOST-BASED INTRUSION DETECTION SYSTEMS (HIDS)

Various software tools such as Metasploit, Sqlmap, Nmap, Browser Exploitation provide the necessary framework to examine and gather information from target system vulnerabilities. Malicious attackers use such information to launch attacks to various applications like FTP server, web server, SSH server, etc. Existing methods such as firewall, cryptography methods and authentications aim to defend host systems against such attacks. However, these solutions have limitations and malicious attackers are able to gain unauthorized access to the system. To address this, a typical HIDS operates at host-level by analyzing and monitoring all traffic activities on the system application files, system calls and operating system [73]. These types of traffic activities are typically called as audit trials. A system call of an operating system is a key feature that interacts between the core kernel functions and low-level system applications. Since an application makes communication with the operating system via system calls, their behavior, ordering, type and length generates a unique trace. This can be used to distinguish between the known and unknown applications [12]. System calls of normal and intrusive process are entirely different. Thus, analysis of those system calls provides significant information about the processes of a system. Various feature engineering approaches have been used for system call-based process classification. They are N-gram [12], [13], sequence gram [14] and pair gram [15]. An important advantage of HIDS is that it provides detailed information about the attacks. The three main components of HIDS, namely the data source, the sensor, and the decision engine play an important role in detecting security intrusions. The sensor component monitors the changes in data source, and the decision engine uses the machine learning module to implement to the intrusion detection mechanism. However, the benchmarking the data source component requires much investigation. Compiling the KDDCup 99 dataset involved the data source component with system calls and Sequence Time-Delay Embedding (STIDE) approach used to analyze the fixed length pattern of system calls to distinguish between normal and anomalous behaviors [13]. Many decision engines have been used to analyze patterns of system calls to detect intrusions. Such a data source is most commonly used among cyber security research community. Apart from system calls, since Windows operating system (OS) does not provide a direct access to system calls, log entries [35] and registry entry manipulations [36] form the other two most used data sources.

This work focuses on the decision engine component to benchmark the data source. Classical methods aim to find information about the nature of the host activity by analyzing the patterns in the sequence of system calls. While STIDE was most used simple algorithm, Support Vector Machines (SVMs), Hidden Markov Models (HMMs) and Artificial Neural Networks (ANNs) are more recently adopted complex methods. In [37], N-gram feature extraction approach was used for compiling the ADFA-LD system call data and N-gram features were passed to different classical machine learning classifiers to identify and categorize attacks. In [39], in order to reduce the dimensions of system calls, K-means and KNN were experimented using a frequency-based model. A revised version of N-gram model was used in [38] to represent system calls with various classical machine learning classifiers for both Binary and Multi-class categories. An approach for HIDS based on N-gram system call representations with various classical machine learning classifiers was proposed in [40]. To reduce the dimensions of N-gram, dimensionality reduction methods were employed. In [41], frequency distribution-based feature engineering approach with machine learning algorithms was explored to handle the zero-day and stealth attacks in Windows OS. In [42], an ensemble approach for HIDS was proposed using language modeling to reduce the false alarm rates which is a drawback in classical methods.

This method leveraged the semantic meaning and communications of system call. The effectiveness of their methods was evaluated on three different publicly available datasets. Overall, the published results are limited in detecting the intrusions and cyberattacks using HIDS. Studies that show an increase in detection rate of intrusions and cyberattacks also show an increase in false alarm rate. The pros and cons of NIDS and HIDS with its efficacy are discussed in detail by [16]. Major advantages of HIDSs are: HIDSs facilitate to detect local attacks and are unaffected by the encryption of network traffic. Major disadvantage is that they need all the configuration files to identify attack, but it is a daunting task due to the huge amount of data. Allowing access to big data technology in the domain of cyber security is of paramount importance, particularly IDS. The motivation of this research is to develop a novel scalable platform with hybrid framework of NIDS and HIDS, which is capable of handling large amount of data with the aim to detect the intrusions and cyberattacks more accurately.

3. EXISTING SYSTEM

many challenges arise since malicious attacks are continually changing and are occurring in very large volumes requiring a scalable solution. There are different malware datasets available publicly for further research by cyber security community. However, no existing study has shown the detailed analysis of the performance of various machine learning algorithms on various publicly available datasets. Due to the dynamic nature of malware with continuously changing attacking methods, the malware datasets available publicly are to be updated systematically and benchmarked.

4. PROPOSED SYSTEM

In this paper, deep neural network (DNN), a type of deep learning model is explored to develop a flexible and effective IDS to detect and classify unforeseen and unpredictable cyber-attacks. The continuous change in network behaviour and rapid evolution of attacks makes it necessary to evaluate various datasets which are generated over the years through static and dynamic approaches. This type of study facilitates to identify the best algorithm which can effectively work in detecting future cyber-attacks.

We propose a highly scalable and hybrid DNNs framework called Scale-Hybrid-IDS-AlertNet (SHIA) which can be used in real time to effectively monitor the network traffic and host-level events to proactively alert possible cyber-attacks.

Today's ICT system is considerably more complex, connected and involved in generating extremely large volume of data, typically called as big data. This is primarily due to the advancement in technologies and rapid deployments of large number of applications. Big data is a buzzword which contains techniques to extract important information from large volume of data. Allowing access to big data technology in the domain cyber security particularly IDS is of paramount importance. The advancement in big data technology facilitates to extract various patterns of legitimate and malicious activities from large volume of network and system activities data in a timely manner that in turn facilitates to improve the performance of IDS. However, processing of big data by using the conventional technologies is often difficult [43]. The purpose of this section is to describe the computing architecture and the advanced methods adopted in the proposed framework, such as text representation methods, deep neural networks (DNNs) and the training mechanisms employed in DNNs.

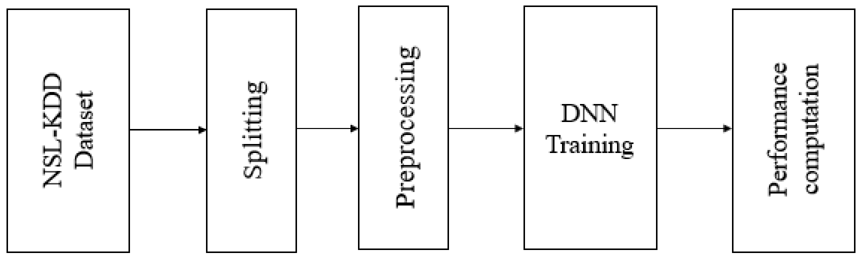


Fig. 1: Block diagram of Proposed system

4.1 Scalable Computing Architecture

The technologies such as Hadoop Map reduce and Apache Spark in the field of high-performance computing is found to be an effective solution to process the big data and to provide timely actions. We have developed scalable framework based on big data techniques, Apache Spark cluster computing platform [45]. Due to the confidential nature of the research, the scalable framework details cannot be disclosed. The Apache spark cluster computing framework is setup over Apache Hadoop Yet Another Resource Negotiator (YARN). This framework facilitates to efficiently distribute, execute, and harvest tasks. Each system has specifications (32 GB RAM, 2 TB hard disk, Intel(R) Xeon(R) CPU E3-1220 v3 @ 3.10GHz) running over 1 Gbps Ethernet network. The proposed scalable architecture employs distributed and parallel machine learning algorithms with various optimization techniques that makes it capable of handling very high volume of network and host-level events. The scalable architecture also leverages the processing capability of the general-purpose graphical processing unit (GPGPU) cores for faster and parallel analysis of network and host-level events. The framework contains two types of analytic engines, they are real-time and non-real time. The purpose of analytic engine is to monitor network and host-level events to generate an alert for an attack. The developed framework can be scaled out to analyze even larger volumes of network event data by adding additional computing resources. The scalability and real-time detection of malicious activities from early warning signals makes the developed framework stand out from any system of similar kind.

4.2 Text Representation Methods

System calls are essential in any operating system depicting the computer processes and they constitute a humongous amount of unstructured and fragmented texts that a typical HIDS uses to detect intrusions and cyber-attacks. In this research we consider text representation methods to classify the process behaviours using system call trace. Classical machine learning approaches adopt feature extraction, feature engineering and feature representation methods. However, with advanced machine learning embedded approach such as deep learning, the necessity of the feature engineering and feature extraction steps can be completely avoided. We adopt such advanced deep learning along with text representation methods to capture the contextual and sequence related information from system calls. The following feature representation methods in the field of NLP are used to convert the system calls into feature vectors in this study.

ISSN- 2394-5125 VOL 08, ISSUE 05, 2021

Bag-of-Words (BoW): This classical and most used representation method is used to form a dictionary by assigning a unique number for each system call. Term document matrix (TDM) and term frequency inverse document frequency (TF-IDF) are employed to estimate the feature vectors. The drawback is that it cannot capture the sequence information of system calls [46].

N-grams: An N-gram text representation method has the capability to preserve the sequence information of system calls. The size of N can be 1 (unit-gram), 2 (bigram), 3 (trigram), 4 (four-gram), etc., which can be employed appropriately depending on the context.

Keras Embedding: This follows a sequential representation method to convert the system calls into a numeric form of vocabulary by simply assigning a unique number for each system call.

4.3 Deep Neural Network

We employ an artificial neural network (ANN) approach as the computational model since it is influenced by the characteristics of biological neural networks to incorporate intelligence in our proposed method. Feed forward neural network (FFN), a type of ANN is represented as a directed graph to pass various system information along edges from one node to another without forming a cycle. We adopt a multilayer perceptron (MLP) model which is a type of FFN having three or more layers with one input layer, one or more hidden layers and an output layer in which each layer has many neurons or units in mathematical notation. We select the number of hidden layers by following a hyper parameter selection method. The information is transformed from one layer to another layer in a forward direction with neurons in each layer being fully connected.

4.4 Statistical Measures

In evaluation to estimate the various statistical measures the ground truth value is required. The ground truth composed of set of connection records labeled either Normal or Attack in the case of Binary classification. Let L and A be the number of Normal and Attack connection records in the test dataset, respectively and the following terms are used for determining the quality of the classification models:

- True Positive (T P) - the number of connection records correctly classified to the Normal class.
- True Negative (T N) - the number of connection records correctly classified to the Attack class.
- False Positive (F P) - the number of Normal connection records wrongly classified to the Attack connection record.
- False Negative (F N) - the number of Attack connection records wrongly classified to the Normal connection record.

5. RESULTS

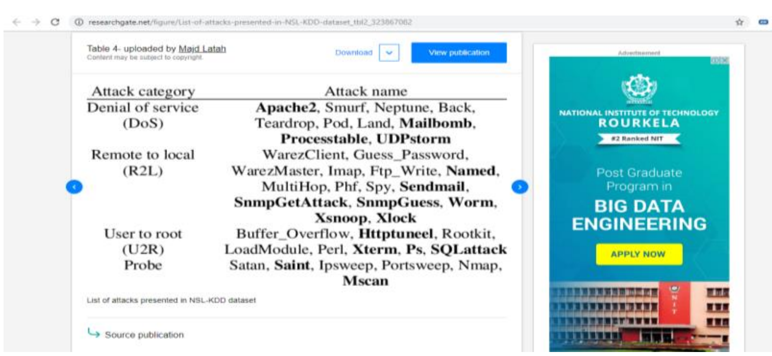

researchgate.net/figure/List-of-attacks-presented-in-NSL-KDD-dataset_tbl2_323867082

Table 4- uploaded by Majd Latah
Content may be subject to copyright.

Download View publication

Attack category	Attack name
Denial of service (DoS)	**Apache2**, Smurf, Neptune, Back, Teardrop, Pod, Land, **Mailbomb**, **Processtable**, **UDPstorm**
Remote to local (R2L)	WarezClient, Guess_Password, WarezMaster, Imap, Ftp_Write, **Named**, MultiHop, Phf, Spy, **Sendmail**, **SnmpGetAttack**, **SnmpGuess**, **Worm**, **Xsnoop**, **Xlock**
User to root (U2R)	Buffer_Overflow, **Httptuneel**, Rootkit, LoadModule, Perl, **Xterm**, **Ps**, **SQLattack**
Probe	Satan, **Saint**, Ipsweep, Portsweep, Nmap, **Mscan**

List of attacks presented in NSL-KDD dataset

Source publication

From above screen shots we can understand that Neptune attack belongs to DOS category. Similarly other attacks belong to different categories

Before running code execute below two commands

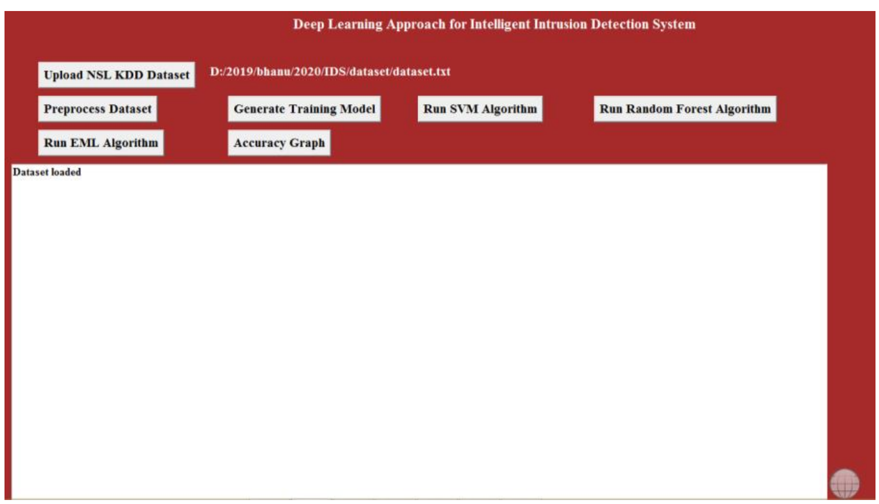
pip install elm

pip install sklearn-extensions

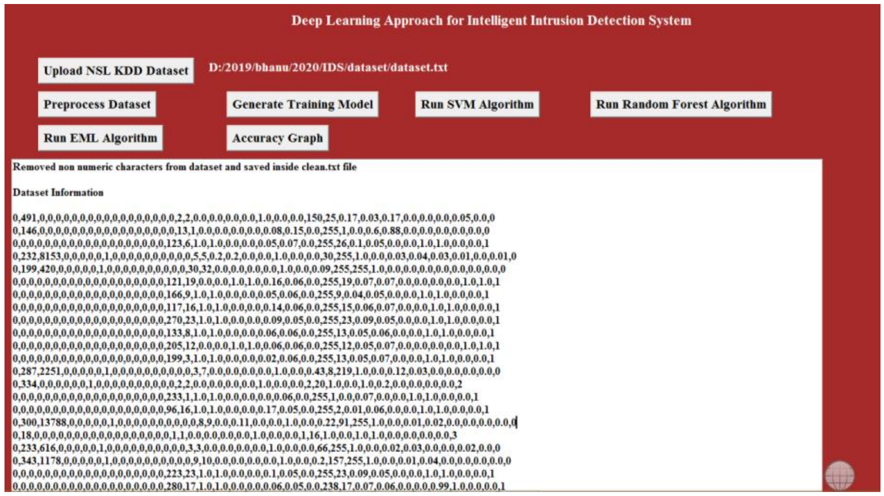
Screen shots

Double click on ‘run.bat’ file to get below screen

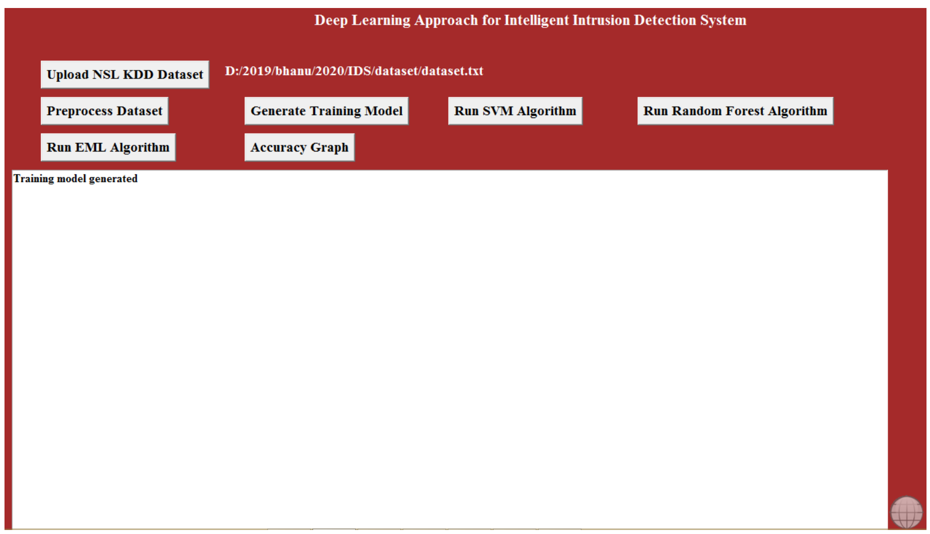
In above screen click on ‘Upload NSL KDD Dataset’ button to upload dataset



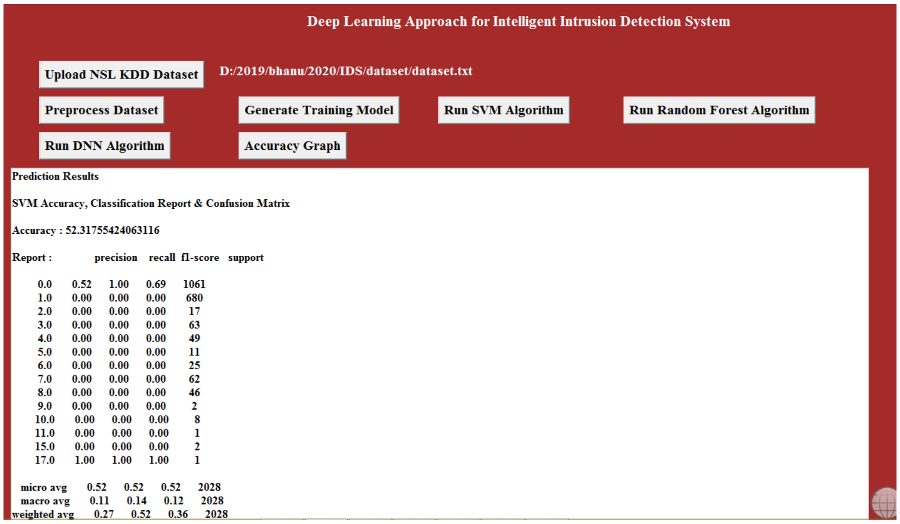
Now click on ‘Preprocess Dataset’ button to assign numeric values to each attack names as algorithms will not understand string names



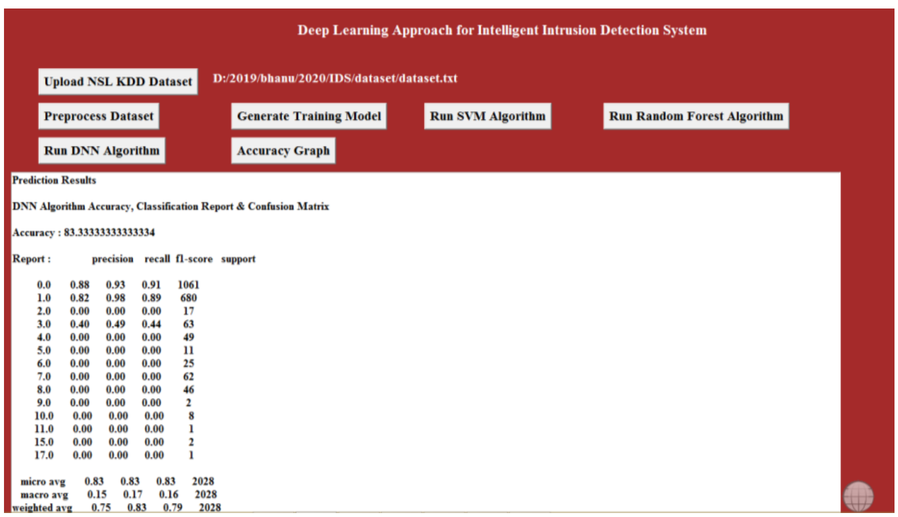
In above screen we can see we assign numeric id to each attack. Now click on ‘Generate Training Model’ button to generate model for training purpose



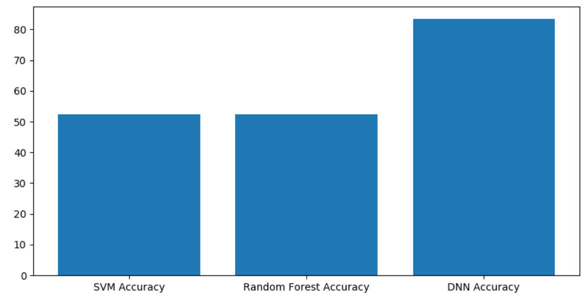
In above screen we can see dataset arrange in such a format so algorithms can build training and test set for prediction and accuracy result. Now click on ‘Run SVM Algorithm’ to get its prediction accuracy



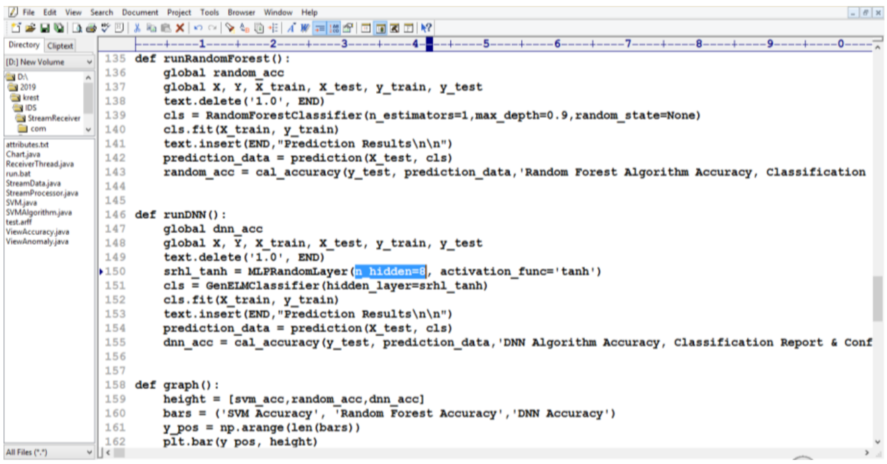
In above screen we can see SVM prediction accuracy is 52%. Now click on 'Run Random Forest Algorithm' button to get its accuracy



In above screen we can see DNN accuracy is better than other two algorithms. DNN algorithm accuracy may be vary different times as it hidden layer will be chosen randomly from dataset. Now click on 'Accuracy Graph' button to get below graph



In above graph x-axis represents algorithm name and y-axis represents accuracy and DNN is the proposed technique. In below code screen you can see i specify DNN hidden layer as 8



4. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a hybrid intrusion detection alert system using a highly scalable framework on commodity hardware server which has the capability to analyze the network and host-level activities. The framework employed distributed deep learning model with DNNs for handling and analyzing very large-scale data in real time. The DNN model was chosen by comprehensively evaluating their performance in comparison to classical machine learning classifiers on various benchmark IDS datasets. In addition, we collected host-based and network-based features in real-time and employed the proposed DNN model for detecting attacks and intrusions. In all the cases, we observed that DNNs exceeded in performance when compared to the classical machine learning classifiers. Our proposed architecture can perform better than previously implemented classical machine learning classifiers in both HIDS and NIDS. To the best of our knowledge this is the only framework which has the capability to collect network-level and host-level activities in a distributed manner using DNNs to detect attack more accurately. The performance of the proposed framework can be further enhanced by adding a module for monitoring the DNS and BGP events in the networks. The execution time of the proposed system can be enhanced by adding more nodes to the existing cluster.

In addition, the proposed system does not give detailed information on the structure and characteristics of the malware. Overall, the performance can be further improved by training complex DNNs architectures on advanced hardware through distributed approach. Due to extensive computational cost associated with complex DNNs architectures, they were not trained in this research using the benchmark IDS datasets.

ISSN- 2394-5125 VOL 08, ISSUE 05, 2021

This will be an important task in an adversarial environment and is considered as one of the significant directions for future work.

REFERENCES

[1] Mukherjee, B., Heberlein, L. T., & Levitt, K. N. (1994). Network intrusion detection. IEEE network, 8(3), 26-41.

[2] Larson, D. (2016). Distributed denial of service attacks-holding back the flood. Network Security, 2016(3), 5-7.

[3] Staudemeyer, R. C. (2015). Applying long short-term memory recurrent neural networks to intrusion detection. South African Computer Journal, 56(1), 136-154.

[4] Venkatraman, S., Alazab, M. "Use of Data Visualisation for Zero-Day Malware Detection," Security and Communication Networks, vol. 2018, Article ID 1728303, 13 pages, 2018. https://doi.org/10.1155/2018/1728303.

[5] Mishra, P., Varadharajan, V., Tupakula, U., & Pilli, E. S. (2018). A detailed investigation and analysis of using machine learning techniques for intrusion detection. IEEE Communications Surveys & Tutorials.

[6] Azab, A., Alazab, M. & Aiash, M. (2016) "Machine Learning Based Botnet Identification Traffic" The 15th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (Trustcom 2016), Tianjin, China, 23-26 August, pp. 1788-1794.

[7] Vinayakumar R. (2019, January 19). vinayakumarr/Intrusion-detection v1 (Version v1). Zenodo. http://doi.org/10.5281/zenodo.2544036.

[8] Tang, M., Alazab, M., Luo, Y., Donlon, M. (2018) Disclosure of cyber security vulnerabilities: time series modelling, International Journal of Electronic Security and Digital Forensics. Vol. 10, No.3, pp 255 - 275.

[9] V. Paxson. Bro: A system for detecting network intruders in realtime. Computer networks, vol. 31, no. 23, pp. 24352463, 1999. DOI http://dx.doi. org/10.1016/S1389-1286(99)00112-7

[10] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436.