TECH-KNOWLEDGE JOURNAL

Anomaly Intrusion Detection Based on Recurrent Neural Networks

Bilal Mohammed*, Ekhlas K. Gbashi
Computer Science Department, University of Technology, Baghdad, Iraq
* cs.19.65@grad.uotechnology.edu.iq

Keywords: **Anomaly Intrusion Detection Systems (AIDS), Classification, Recurrent Neural Networks (RNNs), Recursive Feature Elimination (RFE).**

ABSTRACT

Security is the main issue within computer networks. Intrusion Detection Systems (IDS) are major ways to guarantee information security and to identify attacks before causing any harm. As a reasonable supplement of the firewall, intrusion detection technology can assist the system to deal with attacks and intrusions. There are numerous problems with the existing intrusion detection systems (IDSs) like the inability to detect unknown attacks and too many false positive rates. So, this work was suggested to implement IDS based on Recursive Feature Elimination (RFE) methodology to select features and to use Recurrent Neural Networks (RNN) for classification. RNN was used in the classifications for ten classes such as Fuzzers, DoS, Backdoors, Exploits, Analysis, Generic, Reconnaissance, Shellcode, Worms and Normal. The proposed work has achieved a high accuracy of 98%.

1.0. INTRODUCTION

The huge increasing in networks with increased devices and users endure from security sensitivity that can be almost hard and expensive to be fixed in industries, so the best solution that can be used is IDS for control in traffic. The IDS is a mechanism of security that performs the analysis of the network and computer systems in real-time to define the intrusions and takes fast and accurate measures as a response [1]. In addition, intrusion detection system is different from classic security [2]. Intrusion detection system (IDS) is split into network-level (NIDS) and host-level approach (HIDS) [3]. NIDS is the network action combined by using network devices like the switches, routers, and analyzed to set apart onslaughts and can attack removed within in-network. However, IDS system that utilizes the system actions in the way of different log files grouped via local sensors that executed on the localhost to expose offences which is known as HIDS. Different companies use a mix of both HIDS and NIDS such as Manage Engine Event Log Analyzer is a comprehensive security application with capabilities of both HIDS and NIDS [4].

Misuse (signature) Detection and Anomaly-based detection are two major ways utilized in the intrusion detection system. Misuse detection is based on human inputs where the misuse types of IDS operate based on the gathering and maintaining of known attack signatures. This way is careful in finding unknown treats but it cannot be recognized in case of unknown attack. Anomaly-based detection utilizes ways to find the activities of the unknown threat. In largely anomaly detection creates a high false-positive rate [5]. Anomaly-based depends upon statistical descriptions of the programs or users that have been ultimately aimed for detecting any activities which deviate from a normal behavior profile [6]. Anomaly-based offence exposes ways for cyber systems [7]. Some companies use hybrid

both misuse and Anomaly-based such as Sagan which is another free option using both anomaly- and signature-based detection strategies [4].

Attackers can reach the computer from the internet remotely and create intrusions that may be largely unusable. Detection of intrusion needs to split attacks into five stages for understanding the technique of attacker work such as reconnaissance, exploitation, reinforcement, consolidation, and pillage stages through the top of three stages which can identify an attack, but via fourth and fifth stages and this means that the system is disclosed to an attacker [8]. Through the reconnaissance phase, it needs to collect information to be ready to services, and within applications and operating systems. An attacker during the second exploitation phase, needs to reach to the system by a work on special services, such as taking the special number (passwords) and injection SQL. After internal to system, it will work to install tools that help in reinforcement phase. Based on the errors that user made and application that use, an attacker can access to all system and the last attacker can reach to all accounts that used by users where the attacker can control fully system during the phase of consolidation and the installed backdoor that has been utilized for communication purposes throughout the consolidation stage. Finally, the pillage stage attacker will start to steal the data network intrusion detection system by using some features such as much traffic to take it from remote places.

Recurrent Neural Networks this type of network is characterized by a dynamic neuron model, the so-called Dynamic Elementary Processor (DEP) which is structured as an Auto Regressive Moving Average (ARMA) filter, and is built into the network hidden layer. The paper revision for anomaly detection fully based on recurrent neural networks on different training and testing dataset. The suggested system was implemented by UNSW-NB-15 dataset.

This work proposes a network intrusion detection system which depends on recurrent neural network to classify the normal and the attacks. The paper is arranged as follows: section two illustrates some related works, section three shows Descriptions for the dataset as inclusive and substantive, section four explains the evaluation metrics, section five explain the steps of suggested system, section six illustrates recurrent neural networks, section seven illustrates the experimental results and discussions and finally conclusions and future work.

2.0. RELATED WORK

In the early work stages, many experts have been focused on normal machine learning techniques. Network intrusion detection system may be failed at alert that finds intrusion but no intrusion occurs. This means that the system NIDS suffers from the high rate of error positive. Self-learning system is the best solution and regards better than all solutions of commercial that be not active today such as supervised, semi-supervised and unsupervised may help organizations to classify and find Attacks [8]. Machine learning is used to find the best solutions that achieve better high false-positive which can be at high cost by using industries [9].

One public approach is to utilize Artificial Neural Network (ANN) to build intrusion detection systems such as feed-forward NN used applied to build a classifier, and the back-propagation algorithm has been utilized for training the network classifier [10]. Supervised ways are used largely in intrusion detection systems, e.g., Support Vector Machines (SVMs) [11,12], K-Nearest Neighbor (K-NN), and Random Forest (RF) [13]. Model can be designed to combine Genetic algorithm and KNN algorithm and it was the outcome from this combination for detection rate and accuracy [14].

Wrapper and filter were used by this way for feature selection and classification use J48 and Naive Bayes algorithms worked on UNSW-NB15 dataset [15, 16]. Built algorithm central spot with Association Rule Mining attribute were chosen to select best feature relevant based on KDD99 and UNSW-NB15 [15]. Test four algorithms (Naïve Bayesian, SVM, decision tree and RF) for classification and outcome sensitivity were 93.53%, specificity 7.75%, accuracy 97.49% based on UNSW-NB15 dataset [17].

Machine learning learns normalize the particulars of TCP/IP attributes, but deep learning is a part of machine learning being complex that consisting of many layers and transiting the TCP/IP in many layers. This model was designed to combine discretization and HNB classifier where this approach has focused on problems in intrusion detection and this model was based on a hidden layer in NB model for many classes than can get better accuracy and detection rate of the attack [18].

Newly, deep learning ways as a research hotspot have also mightily utilized in building IDSs. In [19], a proposal NN classifier depending on Self-taught Learning (STL) was suggested for intrusion detections. Based on [20], multi-channel LSTM has assembled many-features by utilizing various sources features, like the nominal-based, binary-based, and numeric-based. This work has detected how to establish an IDS based on Recurrent Neural Network algorithm [21, 22].

3.0. DATASET DESCRIPTION

The team of cyber security research of ACCS (i.e. the Australian Centre for Cyber Security) presented a new data which has been referred to as the UNSW-NB-15 Abbreviated to (University of New South Wale Network-Based 2015) for resolving the issues which have been found in NSL-KDD and the KDD-Cup99 data-sets [8]. This data has been produced in a hybrid way, which included the normal and threat ways of a live network traffic through the utilization of the IXIA Perfect Storm tool which had a storehouse of the modern threats and common vulnerability exposures (CVEs), a store-room include information about the security vulnerabilities and exposures that have been overtly known [23].

Two servers have been utilized in IXIA traffic creator tool where 1 server created the normal activities while the second one created malignant activities in network. Used Tcpdump tool to capture the packet traces from network that have taken many hours of compilation of all data of 100GBs that were split into 1000MB pcaps by utilizing tcpdump[20]. The time of the simulation that had this dataset in the first time was 16 hours on 22-Jan-2015 and was 1 attack per second while the second time was 15 hours on 17-Feb-2105 and 10 attacks per second, for cutting 100 GBs of raw data. From pcap files, the features have been obtained through the utilization of the Argus and Bro-IDS in Linux Ubuntu 14.0.4. This dataset was saved in format CSV file in the name (UNSW-NB15). The data was accessible in 2 ways as follows:

- Whole connection records which consist of two million connection records.
- A part of whole connection records that included 175,341 training connection records and 82,332 test connection records that have been confined with ten attacks. The divided data-set consisted of 42 features as well as their parallel class labels that were Normal and 9 distinctive threat types.

The information about the simulated threats classes and its detailed statistics have been presented in Tables 1 and 2.

Table 1: Explain of attacks of UNSW-NB15 dataset

Name attack	Describe
Fuzzers:	Use network or stall program to create random data for this attack.
Reconnaissance:	Attacker collects data and stimulates from the user system.
Shellcode:	Attack takes advantage of network by used code for taking network packet from the user.
Analysis:	This attack penetrations HTML files and mail random and spam.
Backdoors:	Attack can reach to the system by crossing safety way in silently.
Denial of Service:	In this attack use resources of the system to denial requests or user that authorize to the system.
Exploits:	The attacker monitoring the less point in the system by a known aperture of the system.
Generic:	This type of attack execution without passing on cryptographic.
Worms:	The attack propagates itself to expansion during the network.

Table 2: Train and test set for UNSW-NB15 dataset

Class	Training Sets	Percentages	Testing Sets	Percentages
Analyses	2000	1.141%	677	0.822%
Backdoor	1746	0.996%	583	0.708%
DoS	12264	6.994%	4,089	4.966%
Exploits	33393	19.045%	11,132	13.521%
Fuzzers	18184	10.371%	6,062	7.363%
Generic	40000	22.813%	18,871	22.921%
Normal	56000	31.938%	37,000	44.940%
Reconnaissance	10491	5.983%	3,496	4.246%
Shell Code	1133	0.646%	378	0.459%
Worms	130	0.074%	44	0.053%
Total	**175341**	**100%**	**82,332**	**100%**

4.0. EVALUATION METRIC

It was nominated for applying a similar validation process of each one of the classifier percentages. A data-set with about the same class distributions and sizes has been considered. The classifier has been trained for each one of the folds with the use of the percentage for class. In this part, a clarification has

been provided, which was based upon the measurement of the performance for the task of the machine learning classification, whereas the output may be comprising 2, or more classes. The 10 classes (i.e. the Backdoors, Fuzzers, Analysis, DoS, Exploits, Generic, Reconnaissance, Worms, Shellcode, and Normal) in current investigation, and the 6 distinct combinations of actual and predicted values [24,25], have been listed in Table3.

Table 3: The different metric such as accuracy, precession, recall

Metric	**Eq**
Accuracy	$\frac{TP+TN}{\text{Total Number of Samples}} \times 100$ (1)
Precision	$\frac{TP}{TP+FP}$ (2)
Recall	$\frac{TP}{TP+FN}$ (3)
F-score	$2 \times \frac{Precision \times Recall}{Precision+Recall}$ (4)
False Positive Rate	$\frac{FP}{FP+TN}$ (5)
Area Under the ROC Curve (AUC)	$\int_0^1 \frac{TP}{TP+FN} \, d \, \frac{FP}{TN+FP}$ (6)

5.0. Proposed Model

The model of intelligent ANIDS was proposed with the structure of the hierarchical progressive network where in this model design; approach based on UNSW-NB15 dataset was used. After preprocessing for dataset because this data is raw. Normalization was applied to make all features having values ranging from 0 to 1. The train set has been utilized to train model and the testing set has been utilized for the evaluation of the trained model. By applying the recursive feature, elimination technique was used to select important features from Train set and these important features have used it with the test set at predict. After selecting features of train set, it was built a model for classification by using RNN algorithm on the train set. Finally, evaluating the model by predicting with test set and compared results.

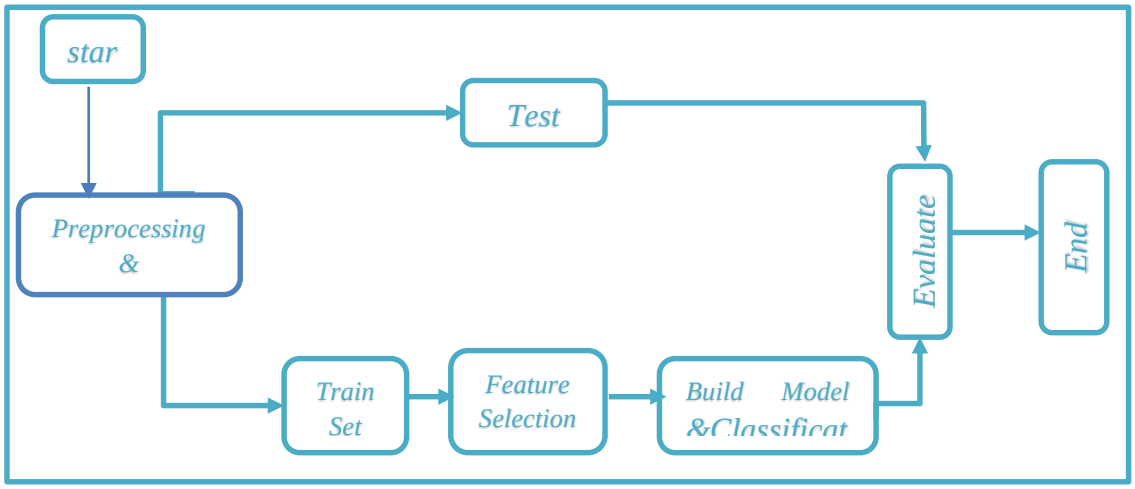


Fig. 1: Proposed model

5.1. Pre-processing

- UNSW-NB15 is dataset with three features which are proto, service and state and these features must convert into numerical values and make each variable has special value.
- UNSW-NB15 is dataset with ten attacks where each attack kind has special value such as Category 0 is assigned to normal,1 for Fizzers, 2 for Backdoors, 3 for DoS, 4 for Analysis, 5 for Exploits, 6 for Reconnaissance, 7 for Generic, 8 for Shellcode and 9 for Worms respectively.

5.2. Normalization

UNSW-NB15 datasets have featured either discrete or continuous. These feature however, have different values; so there is a need to bring these features to be closed together to make this algorithm works easily and the outcomes of work is better and improve the accuracy. Min-Max was used in normalization way to make scale data between (0,1) by using the following method for finding the new value [26]:

$$X_n = \frac{X - X_{min}}{X_{max} - X_{min}} \qquad (7)$$

5.3. Training set & Test set

1. The training set: In this set, our RNN architecture is trained using the known attacks.
2. The Testing set: In this set, we have verified how the same architecture is working in case of unknown attacks

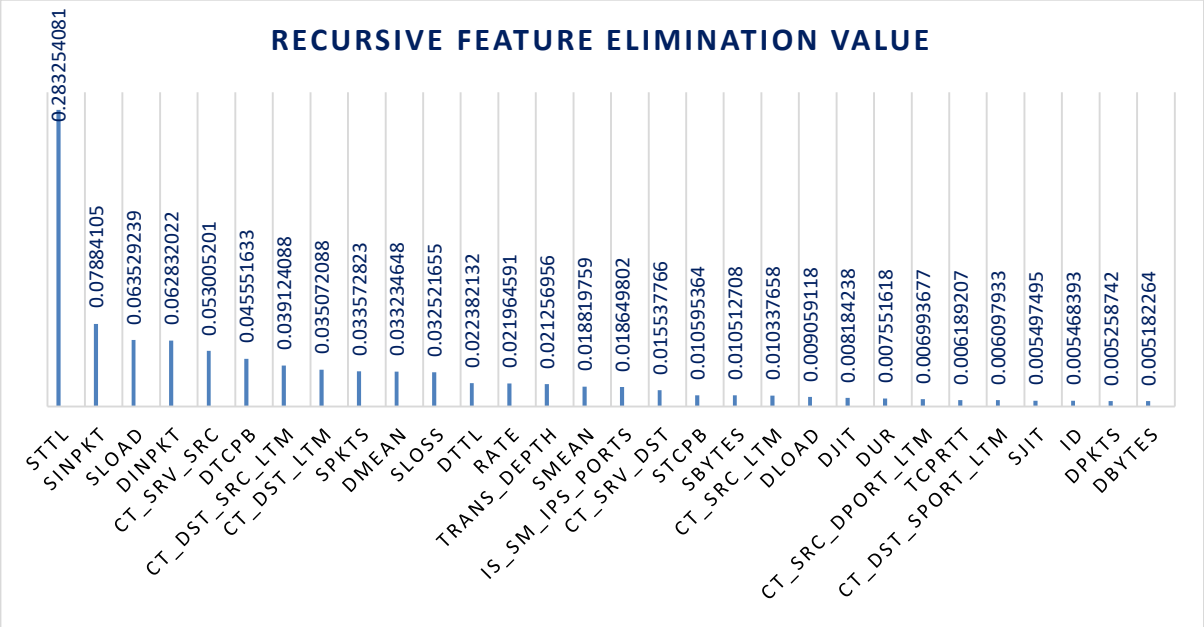


Fig. 2: Important Features from recursive feature elimination

5.4. Features Selection

Feature selection is one of the essential and a lot utilized ways in data preprocessing for the intrusion detection system. It decreases the number of attributes, split redundant, irrelevant, or noisy data, and fetch the essential effects for the intrusion detection system. Proposed the wrapper based Recursive

Feature Elimination is essentially a backward selection of predictors. This way was initiated by creating a model on the full set of predictors and calculating a significance score for every one of the predictors. The least significant predictor(s) were afterwards deleted, the model was re-built and significance scores were calculated again. In exercise, the analyst has assigned the number of predictor subsets to be evaluated as well as to every one of the subset's sizes. Therefore, the subset size was a tuning parameter for RFE. The subset size that optimized the rendering criteria was utilized to choose the predictors depending on the significance rankings. The optimum sub-set was afterwards utilized for the training of the ultimate model. RFE was used with random forest models. The scikit-learn achievement of RFE was utilized with random forest to come up with a feature ranking for the dataset. In the UNSW-NB15, it has chosen 30 features. Figure 2 shows these important features.

5.5. The Build Model

The suggested model was a fully-connected network structure with 3 layers (Input, Hidden and Output). The input layer can specify the number of neurons or nodes in the layer as the first argument, In RNN, there is an important feature which is return sequences, by default set to False, so there is a need to return sequences=True for adding more layers. Also, dropout as regularization technique is used to avoid over-fitting when training. Finally, in the output layer, the softmax function which takes output values between 0 and 1 as shown in Table 4 below:

Table 4: Adding each layer in RNN

The input layer of data with 30 variables since important features that select from RFE technique (the input_dim=30 argument)
The first hidden layer includes 64 nodes and utilizes return_sequences=True
The second hidden layer includes 64 nodes and utilizes return_sequences=True
The third hidden layer includes 64 nodes and utilizes return_sequences=True
The fourth hidden layer includes 64 nodes and utilizes return_sequences=False
The output layer has Dense= 10 nodes for multi-class classification and uses the softmax activation function.

5.6. Evaluate

In this final stage of the proposed system, the evaluation process works by applying the model that trains the training set on new groups and different from the training set, which is the test set by measuring the outputs, for example (Accuracy, Precision, Recall, F-score, False Positive Rate, to know the ability of the model to discover new states of the attacks.

6.0. RECURRENT NEURAL NETWORKS

RNNs are deep learning models made of artificial neurons with one or more returns loops. The returns loops are recurrent rotation over time or sequence [27]. A Recurrent neural network has been successfully used for text data, speech data, classification, regression, natural language processing and generative models [28]. Recurrent neural networks are not suitable for image data and tabular data. The sequential data features are contextual and the analysis of the private data from a sequence does not make any sense. To take the contextual information, every one of the units in the RNN welcomes not

only the present state but preceding ones as well. The standard RNN types are dealing with specific-length sequences only. For the purpose of solving the issue of the long-term dependence, a wide range of the RNNs like the long short-term memory (LSTM), bi-RNN gated recurrent unit (GRU), were proposed.

Let it considers the input structure x= (X_0, X_1, X_2,..., X_{T-1}), the recurrent layer hidden states has been represented as: h= (h_0, h_1, h_2,..., h_{T-1}).

The output layer values y= (y_0, y_1, y_2,..., y_{T-1}). The hidden and the output values are regulated by [29]:

$$h_T = \mathrm{F}(\mathrm{W}_{xh}\, x_T + \mathrm{W}_{hh} h_{T-1} - 1 + \mathrm{b}_h) \qquad (8)$$

$$y_T = \mathrm{O}(W_{ho}\, h_T + b_0) \qquad (9)$$

Where W_{xh}, W_{hh} & W_{ho} represent respectively the input hidden, hidden-hidden and hidden-output weight values. (F) and (O) are the squashing functions for the hidden and the output layers.

7.0. EXPERIMENTAL RESULTS

In early 2015, Keras had the 1st reusable open-source Python implementations for developing and evaluating deep learning models.The system was implemented by UNSW-NB15 dataset has been worked by the Cyber Range Lab of the Australian Centre for Cyber, which was very efficient for offline analyses systems for the Intrusion detection system. This work was implemented by the Python language. The UNSW-NB15 composed of 175,341 train set and 82,332 test set confined with 10 attacks.

In general, obtainable Network Intrusion Detection System by UNSW-NB15 dataset to make an Offline system and was to evaluate of work by recurrent neural network. The UNSW-NB15 was isolated automatically into train and test sets. For train set, most of the RNN network topology showed train accuracy reached to 99%.

Table 5: Results of train set

Attack Type	Precision	Recall	F1-Score	Accuracy
Normal	94%	92%	94%	99%
Generic	99%	98%	97%	99%
Exploits	87%	75%	89%	93%
Fuzzers	89%	96%	93%	88%
DOS	87%	93%	90%	93%
Reconnaissence	93%	83%	88%	99%
Analysis	99%	99%	99%	99%
Backdoor	98%	99%	99%	96%
Shellcode	99%	99%	99%	99%

Worms	99%	99%	99%	99%

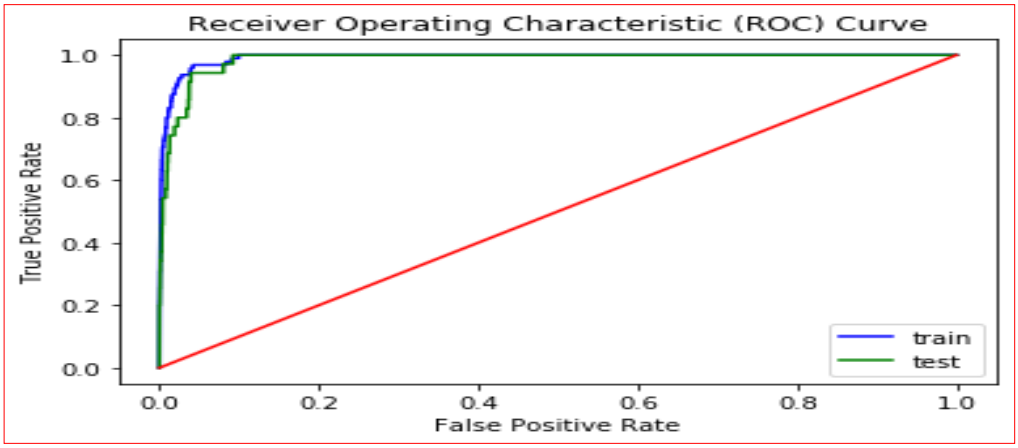
Table 5 shows the training set and the experiments display that after examining the outcomes of the data. It was found that the accuracy of each class in train set was very good. Normal was 99%, Generic was 99%, Exploits was 99%, Fuzzers was 88% and DoS was 93% Reconnaissance was 99%, Analysis was 99% Backdoor was 95% Shellcode was 96% and Worms was 100%. Detection rate was 100%, false alarm rate was 0.09% for Normal. Detection rate was 100%, false alarm rate was 0.00 % for Generic. Detection rate was 100%, false alarm rate was 0.00 % for Exploits. Detection rate was 97%, false alarm rate was 0.030% for Fizzers. Detection rate was 95%, false alarm rate was 0.040% for Dos. Detection rate has been 100%, false alarm rate was 0.02 % for Reconnaissance. Detection rate was 97%, false alarm rate was 0.032 % for Analysis. Detection rate has been 99 %, false alarm rate has been 0.32 % for Backdoor. Detection rate has been 99%, false alarm rate was 0.68% for Shellcode. Detection rate was 100%, false alarm rate was 0.00% for Worms.

Table 6: Results of test set

Attack Type	Precision	Recall	F1-Score	Accuracy
Normal	86%	84%	83%	84%
Generic	99%	99%	99%	99%
Exploits	91%	92%	91%	92%
Fuzzers	91%	83%	86%	83%
DOS	90%	95%	93%	95%
Reconnaissence	98%	98%	97%	98%
Analysis	96%	98%	97%	93%
Backdoor	98%	98%	97%	98%
Shellcode	99%	99%	99%	99%

Worms	99%	99%	99%	99%

For test set shown in Table 6, the results of accuracy on test set were for Normal was 99%, Generic was 99%, Exploits was 95%, Fuzzers was 92% and DoS was 83%



Reconnaissance was 99%, Analysis was 95% Backdoor was 98% Shellcode was 100% Worms was 100%. Detection rate was (100) %, false alarm rate was (0.00) % for Normal. Detection rate was (100) %, false alarm rate was (0.00) % for Generic. Detection rate was (100) %, false alarm rate was (0.00) % for Exploits. Detection rate was (96) %, false alarm rate was (0.04) % for Fuzzers. Detection rate was (85) %, false alarm rate was (0.04) % for DoS. Detection rate was (100) %, false alarm rate was (0.03) % for Reconnaissance. Detection rate was (97) %, false alarm rate was (0.032) % for Analysis. Detection rate was (99) %, false alarm rate has been (0.5) % for Backdoor. Detection rate has been (99) %, false alarm rate was (0.77) % for Shellcode. Detection rate was (100) %, false alarm rate was (0.00) % for Worms.

The operating characteristics of the receiver can be used to provide explanations for the required work to extract the results by providing the graphics shown in Figure 3. Blue color indicated to the evaluation process of training and Green color indicated the evaluation process of testing. The ROC curve of UNSW-NB15 is shown in Figure 3. In most of the cases, RNN was performed well with AUC utilized as standard metric. Which is indicating the fact that the RNN has obtained the maximal true positive rate and lowest false positive rate in some cases approximate to 0.

Fig. 3: ROC Curve for Results of train and test by RNN

8.0 CONCLUSIONS AND FUTURE WORK

The IDS is a technique which may be used for discovering the known and unknown intrusions before the attacker harms the devices of the network. In this paper, proposed Network intrusion has detected alert system by using a Recurrent Neural Network based on by UNSW-NB15 dataset.

For building a flexible and effective Network IDS by UNSW-NB15 dataset. After preprocessing and normalize and feature selection by Recursive Feature Elimination. The proposed Recurrent Neural Network model for the detection of the attacks and threats. The RNN model has been chosen by the comprehensive evaluation of their efficiency compared with the traditional machine learning types.

In addition, network-based features can be used in real-time and employed the suggested RNN model for the detection of the attacks and intrusions. The current proposed model can perform better than previously implemented conventional machine learning types in network intrusion detection system.

Overall, the performance of training and testing by using RNNs architecture was good. It was observed that the IDS anomaly detection accuracy has shown a very high percentage of detecting where the accuracy has reached more than 98% while Detection rate reached to 99% and false alarm rate reached to 0.7%. Overall model performance has been good, particularly in the anomaly detection.

Current work can be extended in 3 directions where firstly, it is possible to apply the system on another intrusion dataset such as Kyoto, WSN-DS and CICIDS2017. Secondly, it may use another model to feature selection such as LDA and rough set and other. Thirdly, it tries to perform the offer approach online.

REFERENCES

[1] Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. Journal of Network and Computer Applications, 60, 19-31.
[2] Abd, D. H., & Obaida, T. H. (2016). A Robust Approach for Mixed Technique of Data Encryption Between DES and RC4 Algorithm. Journal of Kufa for Mathematics and Computer, 3(2), 48-54.
[3] Mishra, P., Pilli, E. S., Varadharajan, V., & Tupakula, U. (2017). Intrusion detection techniques in cloud environment: A survey. Journal of Network and Computer Applications, 77, 18-47.
[4] https://www.dnsstuff.com/host-based-intrusion-detection-systems
[5] Bakshi, A., & Dujodwala, Y. B. (2010, February). Securing cloud from ddos attacks using
intrusion detection system in virtual machine. In 2010 Second International Conference on Communication Software and Networks (pp. 260-264). IEEE.
[6] Bhuyan, M. H., Bhattacharyya, D. K., & Kalita, J. K. (2013). Network anomaly detection: methods, systems and tools. Ieee communications surveys & tutorials, 16(1), 303-336.
[7] Khan, R. U., Kumar, R., Alazab, M., & Zhang, X. (2019, May). A Hybrid Technique To Detect Botnets, Based on P2P Traffic Similarity. In 2019 Cybersecurity and Cyberforensics Conference (CCC) (pp. 136-142). IEEE.
[8] Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., Al-Nemrat, A., & Venkatraman, S. (2019). Deep learning approach for intelligent intrusion detection system. IEEE Access, 7, 41525-41550.
[9] Kanagalakshmi, R., & Raj, V. N. (2014). Network Intrusion Detection Using Hidden Naive Bayes Multiclass Classifier Model. International Journal of Science, Technology & Management, 3(12), 76-84.
[10] Li, L., Yu, Y., Bai, S., Hou, Y., & Chen, X. (2017). An Effective Two-Step Intrusion Detection Approach Based on Binary Classification and $ k $-NN. IEEE Access, 6, 12060-12073.
[11] Yin, C., Zhu, Y., Fei, J., & He, X. (2017). A deep learning approach for intrusion detection using recurrent neural networks. Ieee Access, 5, 21954-21961.

[12] Al-Mejibli, I. S., Abd, D. H., Alwan, J. K., & Rabash, A. J. (2018, November). Performance evaluation of kernels in support vector machine. In 2018 1st Annual International Conference on Information and Sciences (AiCIS) (pp. 96-101). IEEE.
[13] Farnaaz, N., & Jabbar, M. A. (2016). Random forest modeling for network intrusion detection system. Procedia Computer Science, 89(1), 213-217.
[14] Canbay, Y., & Sagiroglu, S. (2015, December). A hybrid method for intrusion detection. In 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA) (pp. 156-161). IEEE.
[15] Anwer, H. M., Farouk, M., & Abdel-Hamid, A. (2018, April). A framework for efficient network anomaly intrusion detection with features selection. In 2018 9th International Conference on Information and Communication Systems (ICICS) (pp. 157-162). IEEE.
[16] Abd, D. H., Sadiq, A. T., & Abbas, A. R. (2019, September). Political Articles Categorization Based on Different Naïve Bayes Models. In International Conference on Applied Computing to Support Industry: Innovation and Technology (pp. 286-301). Springer, Cham.
[17] Khan, N. M., Negi, A., & Thaseen, I. S. (2018, December). Analysis on Improving the Performance of Machine Learning Models Using Feature Selection Technique. In International Conference on Intelligent Systems Design and Applications (pp. 69-77). Springer, Cham.
[18] Canbay, Y., & Sagiroglu, S. (2015, December). A hybrid method for intrusion detection. In 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA) (pp. 156-161). IEEE.
[19] Javaid, A., Niyaz, Q., Sun, W., & Alam, M. (2016, May). A deep learning approach for network intrusion detection system. In Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS) (pp. 21-26).
[20] Jiang, F., Fu, Y., Gupta, B. B., Lou, F., Rho, S., Meng, F., & Tian, Z. (2018). Deep learning based multi-channel intelligent attack detection for data security. IEEE transactions on Sustainable Computing.
[21] Yin, C., Zhu, Y., Fei, J., & He, X. (2017). A deep learning approach for intrusion detection using recurrent neural networks. Ieee Access, 5, 21954-21961.
[22] Khalaf, M., Hussain, A. J., Keight, R., Al-Jumeily, D., Keenan, R., Chalmers, C., ... & Idowu, I. O. (2017, June). Recurrent neural network architectures for analysing biomedical data sets. In 2017 10th International Conference on Developments in eSystems Engineering (DeSE) (pp. 232-237). IEEE.
[23] Moustafa, N., Creech, G., & Slay, J. (2017). Big data analytics for intrusion detection system: Statistical decision-making using finite dirichlet mixture models. In Data analytics and decision support for cybersecurity (pp. 127-156). Springer, Cham.
[24] Abd, D. H., Sadiq, A. T., & Abbas, A. R. (2019, September). Classifying Political Arabic Articles Using Support Vector Machine with Different Feature Extraction. In International Conference on Applied Computing to Support Industry: Innovation and Technology (pp. 79-94). Springer, Cham.
[25] Abd, D. H., & Al-Mejibli, I. S. (2017, December). Monitoring System for Sickle Cell Disease Patients by Using Supervised Machine Learning. In 2017 Second Al-Sadiq International Conference on Multidisciplinary in IT and Communication Science and Applications (AIC-MITCSA) (pp. 119-124). IEEE.
[26] Mehibs, S. M., & Hashim, S. H. (2018). Proposed network intrusion detection system in cloud environment based on back propagation neural network. Journal of University of Babylon for Pure and Applied Sciences, 26(1), 29-40.
[27] Mikolov, T., Karafiát, M., & Burget, L. (2010). Jan ˇCernocky, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In Eleventh annual conference of the international speech communication association (pp. 1045-1048).
[28] Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Advances in neural information processing systems (pp. 3104-3112).
[29] Alom, M. Z., Bontupalli, V., & Taha, T. M. (2015, June). Intrusion detection using deep belief networks. In 2015 National Aerospace and Electronics Conference (NAECON) (pp. 339-344). IEEE.