Asian Journal of Applied Science and Technology (AJAST)
(Open Access Quarterly International Journal) Volume 2, Issue 2, Pages 665-673, April-June 2018

Network Intrusion Detection Using Deep Neural Networks

M.Ponkarthika[1] and Dr.V.R.Saraswathy[2]

[1]PG student, M.E Communication Systems. Kongu Engineering College, India. Email: ponkarthika6@gmail.com
[2]Assistant Professor, Department of ECE, Kongu Engineering College, India. Email: vrsaraswathy@kongu.ac.in

Article Received: 01 March 2018 *Article Accepted: 09 April 2018* *Article Published: 28 April 2018*

ABSTRACT

Due to the advance of information and communication techniques, sharing information through online has been increased. And this leads to creating the new added value. As a result, various online services were created. However, as increasing connection points to the internet, the threats of cyber security have also been increasing. Intrusion detection system (IDS) is one of the important security issues today. A Network Intrusion Detection System (NIDS) helps system administrators to detect network security breaches in their organization. However, many challenges arise while developing a flexible and effective NIDS for unforeseen and unpredictable attacks. In this work, a deep learning based approach is to implement such an effective and flexible NIDS. In this paper, model of an intrusion detection system is explored based on deep learning, and Long Short Term Memory (LSTM) architecture is applied to a Recurrent Neural Network (RNN) and train the IDS model using KDD Cup 1999 dataset. Through the performance test, it is confirmed that the deep neural network is effective for NIDS.

Keywords: Deep learning, Intrusion detection system, Long short term memory, Recurrent neural network, KDD.

1. INTRODUCTION

With the increasingly deep integration of the Internet and society, the Internet is changing the way in which people live, study and work, but the various security threats that we face are becoming more and more serious. With increasing the importance of cyber security, researches about Intrusion Detection System(IDS) have been actively studying. IDS protect a network system from malicious software attacks. There are two types Intrusion Detection Systems according to an object of observation [1]. The first one, Host-based IDS (HIDS), watches the host system operation or states. It detects system events such as unauthorized installation or access. Also, it checks the state of ram or file system whether the data is expected one or not, because the detection of HIDS is based on the system event log, a false alarm ratio is low. But it cannot analyze behaviors related to the network. The second one, Network-based IDS (NIDS) is placed on DMZ or choke point of the network edge. It observes a real-time network traffic and analyzes it for detecting unauthorized intrusions or the malicious attacks. The detection techniques are two types [2]. The first technique is a behavior-based intrusion detection called anomaly detection. It catches attacks by comparing an abnormal behavior to a normal behavior. The second technique is a knowledge based intrusion detection called misuse detection. This one detects the attacks based on the known knowledge. intrusion detection is usually equivalent to a classification problem, such as a binary or a multiclass classification problem, i.e., identifying whether network traffic behaviour is normal or anomalous, or a five-category classification problem, i.e., identifying whether it is normal or any one of the other four attack types: Denial of Service (DOS), User to Root (U2R), Probe (Probing) and Root to Local (R2L). In short, the main motivation of intrusion detection is to improve the accuracy of classifiers in effectively identifying the intrusive behaviour.

Machine learning methodologies have been widely used in IDS. However, most of the traditional machine learning methodologies belongs to shallow learning, they cannot effectively solve the massive intrusion data classification problem that arises in the face of a real network application environment. In addition, shallow learning is unsuited to intelligent analysis and the forecasting requirements of high-dimensional learning with massive data. In contrast, deep learners have the potential to extract better representations from the data to create much better models. As a result, intrusion detection technology has experienced rapid development after falling into a relatively slow period. In this paper deep learning based intrusion detection system model is proposed.

Deep learning achieves high level abstractions in data by using a complex architecture or composition of non-linear transformations. Therefore, we can acquire a high detection rate. In this paper, we apply Long Short Term Memory (LSTM) to Recurrent Neural Network (RNN) and use it for an IDS model [3] [4]. We train the model by using KDD Cup 1999 dataset and measure the performance.

The remainder of this paper is structured as follows. Section II presents a related works. Section III gives a brief description of RNN and LSTM. Results are discussed in section IV. Finally the paper concludes in Section V.

2. RELATED WORK

In previous studies, a number of approaches based on traditional machine learning, including SVM [5], [6], K-Nearest Neighbour (KNN) [7], ANN [8], Random Forest (RF) [9], [10] and others [11], [12], have been proposed and have achieved success for an intrusion detection system. There are two kinds of detection, including anomaly- based and misuse-based [13] [14]. In recent years, deep learning, a branch of machine learning, has become increasingly popular and has been applied for intrusion detection; studies have shown that deep learning completely surpasses traditional methods. In[15], the authors utilize a deep learning approach based on a deep neural network for flow-based anomaly detection, and the experimental results show that deep learning can be applied for anomaly detection in software defined networks. In [16], the authors propose a deep learning based approach using self-taught learning (STL) on the benchmark NSL-KDD dataset in a network intrusion detection system.

According to [17], RNNs are considered reduced-size neural networks. In that paper, the author proposes a three layer RNN architecture with 41 features as inputs and four intrusion categories as outputs, and for misuse-based IDS. However, the nodes of layers are partially connected, the reduced RNNs do not show the ability of deep learning to model high-dimensional features, and the authors do not study the performance of the model in the binary classification. With the continuous development of big data and computing power, deep learning methods have blossomed rapidly, and have been widely utilized in various fields. Following this line of thinking, a deep learning approach for intrusion detection using recurrent neural networks(RNN-IDS) is proposed in this paper.

Compared with previous works, we use the RNN-based model for classification rather than for pretraining. Besides, we use the NSL-KDD dataset with a separate training and testing set to evaluate their performances in detecting network intrusions in both binary and multiclass classification, and we compare it with J48, ANN, RF, SVM and other machine learning methods proposed by previous researchers.

3. PROPOSED WORK

Recurrent Neural Network is the effective model to training the sequence data. The conventional RNN has trouble when it is used to train with a long step size. The proposed Long Short Term Memory architecture is used to address this problem.

3.1. Recurrent Neural Network

Recurrent neural networks include input units, output units and hidden units, and the hidden unit completes the most important work. The RNN model essentially has a one-way flow of information from the input units to the hidden units, and the synthesis of the one-way information flow from the previous temporal concealment unit to the current timing hiding unit is shown in Fig. 1. We can regard hidden units as the storage of the whole network, which remember the end-to-end information. When we unfold the RNN, we can find that it embodies the deep learning. A RNNs approach can be used for supervised classification learning.

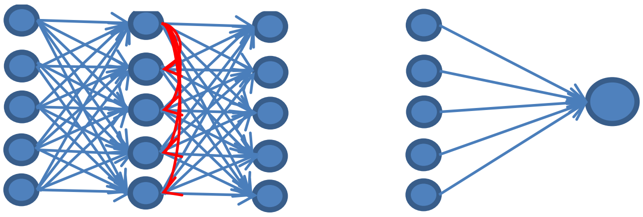


Fig.1 Recurrent Neural Networks (RNNs).

Recurrent neural networks have introduced a directional loop that can memorize the previous information and apply it to the current output, which is the essential difference from traditional Feed-forward Neural Networks(FNNs).The preceding output is also related to the current output of a sequence, and the nodes between the hidden layers are no longer connectionless; instead, they have connections. Not only the output of the input layer but also the output of the last hidden layer acts on the input of the hidden layers.

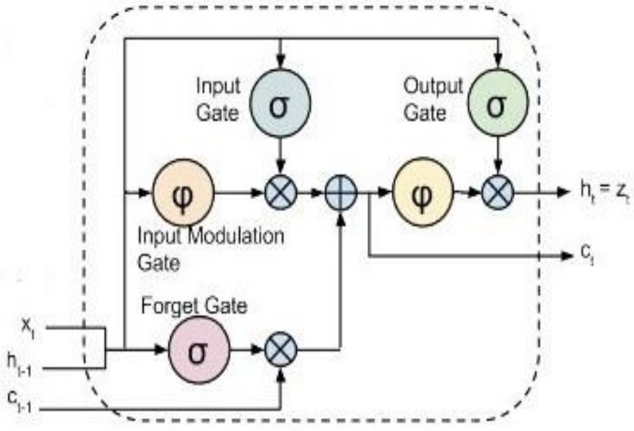


Fig.2 Single LSTM cell

3.2. Long Short Term Memory

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network capable of learning order dependence in sequence prediction problems. A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate as shown in Fig.2.

The cell is responsible for "remembering" values over arbitrary time intervals; hence the word "memory" in LSTM. The input gate decides the ratio of input. The forget gate passes the previous memory or not. The output gate determines whether passing the output of memory cell or not. By using LSTM, we can solve the vanishing and exploding gradient problems due to the three gates.

In LSTM RNN architecture, there current hidden layer is replaced by LSTM cell. Flow chart of this project is shown in Fig.3.Kdd cup 99 dataset is taken as an input, it has 42 features with 400 samples. The input dataset is trained using LSTM method. Test datasets are made with five features and 10 samples. Testing can be performed based on the three gate operations which are presented in the LSTM cell.

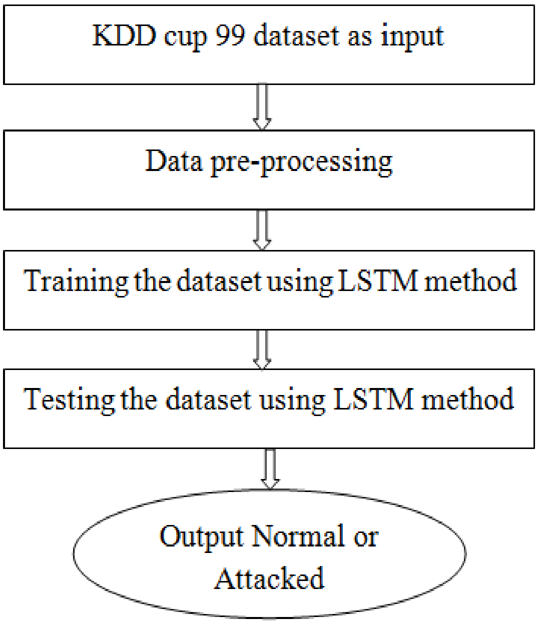
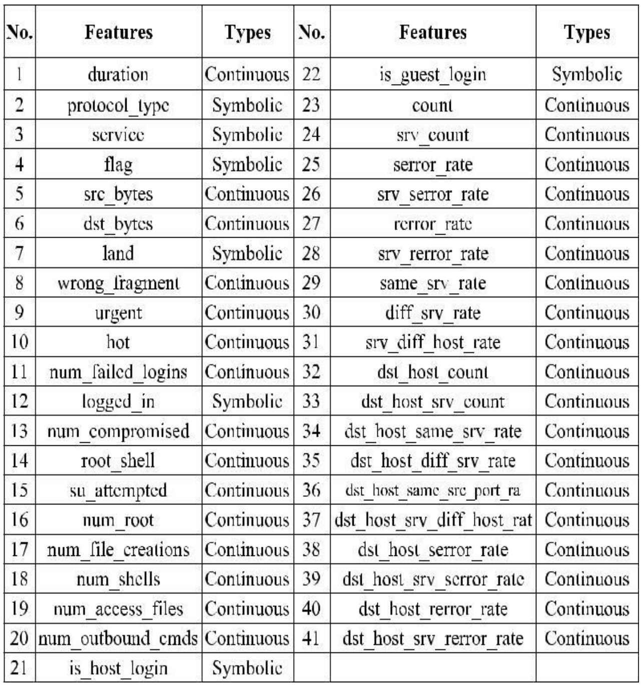

KDD cup 99 dataset as input
Data pre-processing
Training the dataset using LSTM method
Testing the dataset using LSTM method
Output Normal or Attacked

Fig.3 Approach used in proposed method

3.3. Dataset description

The NSL-KDD dataset [19],[20] generated in 2009 is widely used in intrusion detection experiments. In the prior writing [21]– [23], every one of the specialists utilize the NSL-KDD as the benchmark dataset, which not just successfully takes care of the natural excess records issues of the KDD Cup 1999 dataset yet additionally makes the quantity of records sensible in the preparation set and testing set, such that the classifier does not support more incessant records. Table.1 shows that the features of NSL-KDD dataset.

Table.1 Features of NSL-KDD dataset.



No.	Features	Types	No.	Features	Types
1	duration	Continuous	22	is_guest_login	Symbolic
2	protocol_type	Symbolic	23	count	Continuous
3	service	Symbolic	24	srv_count	Continuous
4	flag	Symbolic	25	serror_rate	Continuous
5	src_bytes	Continuous	26	srv_serror_rate	Continuous
6	dst_bytes	Continuous	27	rerror_rate	Continuous
7	land	Symbolic	28	srv_rerror_rate	Continuous
8	wrong_fragment	Continuous	29	same_srv_rate	Continuous
9	urgent	Continuous	30	diff_srv_rate	Continuous
10	hot	Continuous	31	srv_diff_host_rate	Continuous
11	num_failed_logins	Continuous	32	dst_host_count	Continuous
12	logged_in	Symbolic	33	dst_host_srv_count	Continuous
13	num_compromised	Continuous	34	dst_host_same_srv_rate	Continuous
14	root_shell	Continuous	35	dst_host_diff_srv_rate	Continuous
15	su_attempted	Continuous	36	dst_host_same_src_port_ra	Continuous
16	num_root	Continuous	37	dst_host_srv_diff_host_rat	Continuous
17	num_file_creations	Continuous	38	dst_host_serror_rate	Continuous
18	num_shells	Continuous	39	dst_host_srv_serror_rate	Continuous
19	num_access_files	Continuous	40	dst_host_rerror_rate	Continuous
20	num_outbound_cmds	Continuous	41	dst_host_srv_rerror_rate	Continuous
21	is_host_login	Symbolic			

3.4. Data Preprocessing

First, according to some features, such as‘ duration’, ‘src_bytes’ and ‘dst_bytes’, where the difference between the maximum and minimum values has a very large scope, we apply the logarithmic scaling method for scaling to obtain the ranges of ‘duration’, ‘src_bytes’ and ‘dst_bytes’. Second, the value of every feature is mapped to the [0,1]range linearly according to (1), where Max denotes the maximum value and Min denotes minimum value for each feature.

$$\text{xi} = \frac{xi - min}{max - min} \qquad (1)$$

3.5. Training LSTM with the dataset

To minimize LSTM's total error on a set of training sequences, iterative gradient descent such as back propagation through time can be used to change each weight in proportion to its derivative with respect to the error. An issue with utilizing inclination plummet for standard Recurrent Neural Network's is that blunder angles vanish exponentially rapidly with the extent of the time slack between critical occasion demonstrate ever, when error values are back- engendered from the output, the error remains in the unit's memory. This "error carousel" continuously feeds the error back to each of the gates until they learn to cut off the value. Thus, regular back propagation is effective at training an LSTM unit to remember values for long durations.

3.6. Testing the dataset to LSTM

A test dataset is independent of the training dataset, but follows the same probability distribution as the training dataset. After the dataset is trained more times, test the LSTM with the sample inputs. So the sample inputs were given by uploading the kddcup’99 dataset. Then the initialized LSTM network layer function is called and also all the default parameters were also called. The range, iteration and the count rate is mentioned as per the need. The gradient range is obtained by difference of height and width of the matrix. If the gradient value is less than 0.1 then the particular connection is normal and if it is greater than 0.1 then the particular connection is not attacked and the percentage of accuracy will be determined.

4. RESULT ANALYSIS

The preprocessed data's is given as input to the Long Short Term Memory network for training. The input data is trained using gates presented in the LSTM cell. Once the neural network is trained for more iterations, test data's were given to the network. The predicted results i.e. Normal or Anomaly, efficiency (%), error(%) were discussed below.

4.1. Parameter for evaluation

The performance of Network Intrusion Detection using LSTM-RNN Network is evaluated in terms of accuracy. Accuracy (AC) shows the percentage of true detection over total traffic trace:

$$AC=\frac{TP+TN}{TP+TN+FP+FN} \qquad (2)$$

$\frac{TP+TN}{TP+TN+FP+FN}$ True Positive (TP) is the number of attack records correctly classified. True Negative (TN) is the number of normal records correctly classified. False Positive (FP) is the quantity of typical records mistakenly grouped. False Negative (FN) is the quantity of assault records inaccurately arranged.

4.2. Results

The resulting table.2 shows the result analysis between RNN and LSTM network.

Table.2 Comparison of RNN and LSTM-RNN result analysis

LEARNING RATE	RNN	LSTM-RNN
10	78	80
50	78	80.5
100	81	82
150	81.5	82
200	82	83

Efficiency is calculated by finding the difference between expected detection rate and the error rate as shown in table.3

Table.3 Learning Rate Vs Efficiency

Learning Rate (Iteration)	Efficiency (%)	Error Rate (%)
1	70	30
2	70	30
3	71	29
4	73	27
5	73	27
6	75	25
7	78	22
8	80	20
9	80	20
10	81	19

% Efficiency η = (100- Error Rate) (3)

During the first iteration in the learning process we obtained the lower efficiency rate at a range of 70%.When training the dataset at more number of iteration the error rate will get decreased and the efficiency get increased. Efficiency has reached the maximum of 93%.

Table.4 Comparison between Expected and Obtained Output

Dimension	Expected Output	Obtained Output
10×5	0	ATTACKED
	0	NORMAL
	0	ATTACKED
	1	NORMAL
	0	ATTACKED
	0	NORMAL
	0	ATTACKED
	0	ATTACKED
	1	NORMAL
	1	NORMAL

Here ‘0’ denotes that the provided connection is Anomaly and ‘1’ denotes that the provided connection is not Attacked or Normal.

While testing with the sample of ten objects it predicts correctly at an efficiency of 93%.It has an error rate of 7%. Long Short Term Memory architecture is used to increasing the performance of higher dimension input data. Table.5 shows the efficiency value for different dimensions of test dataset.

Table.5 Testing Efficiency with Higher Dimension

S.No	Dimension	Efficiency (%)
1	50×5	93
2	90×5	91
3	130×5	90
4	160×5	90
5	200×5	89.5

5. CONCLUSION

In this paper, the Intrusion detection system classifier is implemented based on the Long Short Term Memory Recurrent Neural Network. A primary goal of the proposed method is to detect the network behaviour is normal or affected based on the past observations. For training phase, the dataset is generated by extracting instances from KDD Cup 1999 dataset. In order to find the proper learning rate and hidden layer size, the experiment is carried out by changing the values. For testing phase, five test datasets are used and measured the performance. Weights are initialized randomly. By comparing it to other IDS classifier, we found that the attacks are well detected by LSTM-RNN classifier. In future work, the weight can be initialized using different algorithms for increasing the performance of the proposed IDS model.

REFERENCES

[1] Bai, Yuebin, and Hidetsune Kobayashi, Intrusion detection systems: technology and development, AINA 2003. 17th International Conference on. IEEE, 2003.

[2] Depren, Ozgur, et al., An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks, Expert systems with Applications 29.4, pp.713-722, 2005.

[3] Hochreiter, Sepp, and Jrgen Schmidhuber, Long short-term memory, Neural computation 9.8, pp.1735-1780, 1997.

[4] Luko EviIus, Mantas, and Herbert Jaeger, Reservoir computing approaches to recurrent neural network training, Computer Science Review 3.3 pp.127-149, 2009.

[5] F.Kuang, W.Xu, and S.Zhang, “novel hybrid KPCA and SVM with GA model for intrusion detection,’’ Appl. Soft Comput., vol. 18, pp. 178–184, May 2014.

[6] R. R. Reddy, Y. Ramadevi, and K. V. N. Sunitha, ‘‘Effective discriminant function for intrusion detection using SVM,’’ in Proc. Int. Conf. Adv. Comput., Commun. Inform. (ICACCI), Sep. 2016, pp. 1148–1153.

[7] W. Li, P. Yi, Y. Wu, L. Pan, and J. Li, ‘‘A new intrusion detection system based on KNN classification algorithm in wireless sensor network,’’ J. Elect. Comput. Eng., vol. 2014, Jun. 2014, Art. no. 240217.

[8] B.Ingreand A.Yadav, ‘‘Performance analysis of NSL KDD data setusing ANN,’’ in Proc. Int. Conf. Signal Process. Commun. Eng. Syst., Jan.2015, pp. 92–96.

[9] N. Farnaaz and M. A. Jabbar, ‘‘Random forest modeling for network intrusion detection system,’’ Procedia Comput. Sci., vol. 89, pp. 213–217, Jan. 2016.

[10] J. Zhang, M. Zulkernine, and A. Haque, ‘‘Random-forests-based network intrusion detection systems,’’ IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 38, no. 5, pp. 649–659, Sep. 2008.

[11] J. A. Khan and N. Jain, ‘‘A survey on intrusion detection systems and classification techniques,’’ Int.J.Sci.Res.Sci., Eng.T echnol.,vol.2,no.5, pp. 202–208, 2016.

[12] S. Taeshik, M. Jongsub, A hybrid machine learning approach to network anomaly detection, ScienceDirect, Information Sciences 177, pp.37993821, 2007.

[13] C. Yehui, A. Ajith, Y. Bo, Hybrid flexible neural-tree-based intrusion detection systems, International Journal of Intelligent Systems, Volume 22 Issues 4, pp.337-352, April 2007.

[14] A.Javaid, Q.Niyaz, W.Sun, and M.Alam,‘‘A deep learning approach for network intrusion detection system,’’ presented at the 9th EAI Int. Conf. Bio-inspired Inf. Commun. Technol. (BIONETICS), NewYork, NY, USA, May 2016, pp. 21–26.

[15] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, ‘‘Deep learning approach for network intrusion detection in software defined networking,’’ in Proc. Int. Conf. Wireless Netw. Mobile Commun. (WINCOM), Oct. 2016, pp. 258–263.

[16] M. Sheikhan, Z. Jadidi, and A. Farrokhi, ‘‘Intrusion detection using reduced-size RNN based on feature grouping,’’ Neural Comput. Appl., vol. 21, no. 6, pp. 1185–1190, Sep. 2012.

[17] KDDCup99, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.htl.

[18] M. Tavallaee, E. Bagheri, W. Lu, and A. A. A. Ghorbani, ‘‘A detailed analysis of the KDD CUP 99 dataset,’’ in Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl., Jul. 2009, pp. 1–6.

[19] S.Revathiand A.Malathi, ‘‘A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection,’’ Int. J. Eng. Res. Technol., vol. 2, pp. 1848–1853, Dec. 2013.

[20] N.Paulauskasand J.Auskalnis, ‘‘Analysis of data pre-processing in fluence on intrusion detection using NSL-KDD dataset,’’ in Proc. Open Conf. Elect., Electron. Inf. Sci. (eStream), Apr. 2017, pp. 1–5.

[21] P.S.Bhattacharjee, A.K.M.Fujail, and S.A.Begum,‘‘Intrusion detection system for NSL-KDD data set using vectorised fitness function in genetic algorithm,’’ Adv. Comput. Sci. Technol., vol. 10, no. 2, pp. 235–246, 2017.

[22] R. A. R. Ashfaq, X.-Z. Wang, J. Z. Huang, H. Abbas, and Y.-L. He, ‘‘Fuzziness based semi-supervised learning approach for intrusion detection system,’’ Inf. Sci., vol. 378, pp. 484–497, Feb. 2017.