INTRUSION DETECTION SYSTEM USING GATED RECURRENT NEURAL NETWORKS

MRS. G PRANITHA
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh, India
pranitha.cse@anits.edu.in

D. KIRAN MAHESH REDDY, B. DEEPIKA, G. ALEKHYA, CH.N.VENNELA
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh, India
dkmaheshreddy.16.cse@anits.edu.in, bdeepika.16.cse@anits.edu.in, glekhya.16.cse@anits.edu.in, chnagavennela.16.cse@anits.edu.in.

Abstract- As use of the net and related technologies which are spreading around the world, the use of those networks now creates new threats for organizations. An Intrusion detection system(IDS) plays a major role in preserving network security. During this paper, we propose a deep learning-based Intrusion Detection System using recurrent neural networks with gated recurrent units(GRU-IDS). The dataset used for evaluating the GRU-IDS is that the NSL-KDD dataset. To chop back the dimensionality of the NSL-KDD dataset we used a Random Forest classifier for feature selection. The experimental result suggests that the performance of GRU-IDS is superior compared to traditional machine learning classification methods.

Keywords- Intrusion detection, Recurrent Neural Network, Gated Recurrent Unit, GRU-IDS, machine learning, deep learning.

I. INTRODUCTION

The Intrusion Detection System(IDS) assists in preserving the network free from various kinds of attacks by using it as a software in various computer or network systems. An intrusion detection system(IDS) inspects all outbound and inbound network actions and finds out the doubtful patterns which will point to network or system intrusion or attack from someone trying to crack into or conciliate a system[1]. The type of detection techniques seen in Intrusion detection system are misuse detection and anomaly detection[2]. A misuse detection must know the attributes or signatures of intrusion. The most drawback of misuse detection is it’s going to be unsuccessful to detect new attacks. In Anomaly-based IDS, this IDS system first defines the conventional behavior of the network and so checks whether the particular behavior deviates from the conventional behavior or not, supported that comparison it defines unknown attacks.

The traditional machine learning technologies like SVMs[3], ANNs[10], Random Forest[4], Naive Bayes[5], KNN[6] and J48[7] are examined that they show lower accuracy rate in intrusion detection. So we’ve decided to create up an IDS model that may detect abnormal behavior within the network and generate more accuracy rate in intrusion detection.

1.1 Intrusion Detection System

In the modern network, IDS has become an important part of all-over network security architecture. Firstly we’d like to grasp the Intrusion before Intrusion Detection System. Intrusion refers to unauthorized access to a system or a service by compromising the system to enter into an insecure state. An Intrusion will be featured in terms of Confidentiality, Integrity, Availability. Confidentiality indicates protecting information from unauthorized users. Integrity ensures that the information is accurate and safeguarded even after an intruder’s modification. Availability brings up the power of the user to access information incorrectly format. The user who does intrusion is termed an intruder, who leaves some traces which are being detected by an Intrusion detection system. The intrusion detection system monitors the network to seek out any malicious activity and issues conscious of the administrator. Modern network-based environments need IDS for safe communication between the organizations. Some IDS are capable of responding to detected intrusion upon discovery. Those are called IPS(Intrusion Prevention System).

1.2 Random forest classifier

Random forest classifier falls under supervised learning and it's an ensemble algorithm. Ensemble methods use multiple learning algorithms to get higher predictive performance than usually compared to any of the constituent learning algorithms. Random forest classifiers are used for feature selection where it creates decision trees from a randomly selected subset of the training dataset. Each tree within the random forest has its own predicted class value and also the class with most of the votes becomes the prediction class for our model. The primary choice of selecting this classifier is that it doesn't overfit. The study on this classifier shows that it generates more accuracy on the nsl-kdd test dataset[4]. Hence after a change in some trees, they have an inclination to own a continuing performance. Using this classifier we have an opportunity of getting an accurate value of 99.13%.

1.3 Recurrent Neural Network(RNN)

Neural networks are a gaggle of algorithms, modelled supported the working of the human brain, that are designed to acknowledge patterns. All real-world data, images, sound must be translated into numerical series because neural networks recognize numerical patterns, contained in vectors. Recurrent Neural Network usually process sequences where the output from the preceding step is fed as input to the current step. An RNN consists of the input layer (x_t), a hidden layer (h_t), and an output layer (o^t). RNNs are different from the normal feedforward neural networks because it consists of a directional loop that acts as a memory for storing the previous state's information and for all the inputs they use the same parameters which reduce the complexity of RNNs. Hidden layers will be quite one depending upon the complexity of the project.

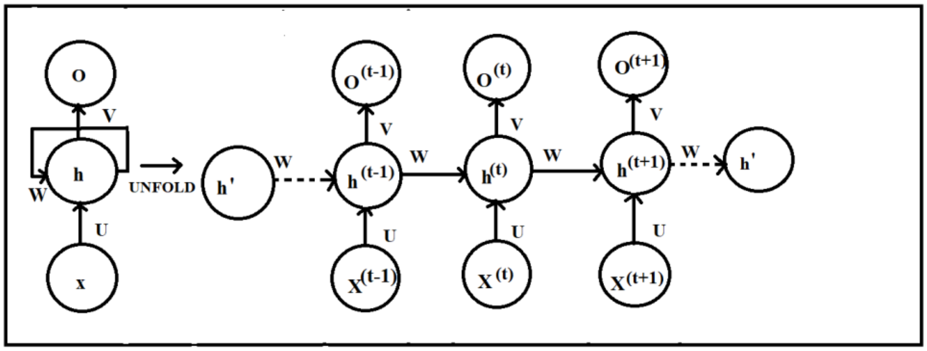


FIGURE 1.Unfolded Structure of Recurrent Neural Networks

As shown in Figure 1 U,V, and W are used as weight matrices. U matrix is used between the input to hidden layer units, W matrix is used between the hidden to hidden layer units, and the V matrix is used between hidden and the output layer units.

1.4 Gated Recurrent Unit

Gated Recurrent Unit (GRU) came into existence to overcome the vanishing gradient hassle that is seen in the regular RNN. GRU was build by using two gates the update and the reset gates. The update gate helps the model to determine how much past information is needed to be passed along the future. The reset gate is mainly used to select how much of the past information needs to be forgotten. The reset gate helps the GRU-IDS model to remove unwanted information in the future.

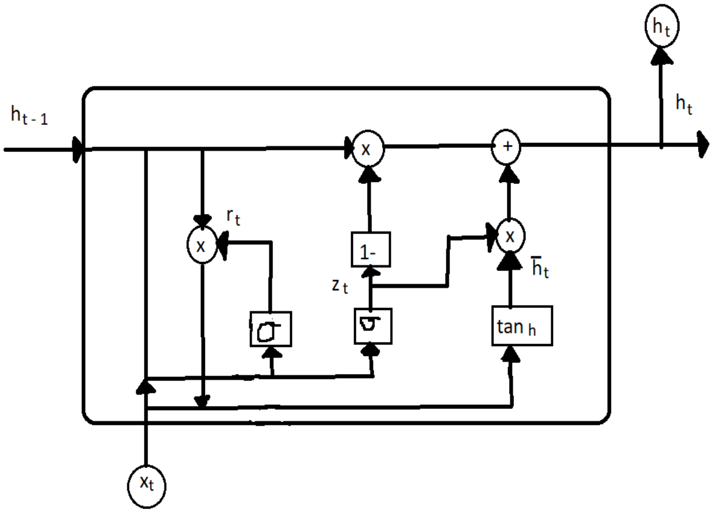


FIGURE 2. Structure of GRU

$$r_t = \sigma(W_r x_t + U_r h_{t-1})$$

$$z_t = \sigma(W_z x_t + U_z h_{t-1})$$

$$\bar{h}_t = \tanh(W_h x_t + U(r_t \odot h_{t-1}))$$

$$h_t = (1 - z_t)h_{t-1} + z_t \bar{h}_t$$

Where,
x_t = input at time step t.
h_t = hidden layer input at time step t.
z_t = update gate output at time step t.
r_t = reset gate output at time step t.

II. Related Work

S.Revathi, A Malathi (2013)[8], done a detailed study on NSL-KDD dataset. They found out that the NSL-KDD dataset consists of four classes of attacks and one normal class. They have used data mining techniques like J48, Random forest, Naïve Bayes, CART and SVM to find attack classes from the normal class. The Random Forest Classifier shows good results on the test dataset accuracy.

Bhupendra Ingre, Anamika Yadav (2015)[9], proposed an Intrusion Detection System using ANN and calculated various performance measures like Accuracy, Detection Rate, and False Positive Rate. This model shows an accuracy rate of 81.2% and 79.9% on the train and test NSL-KDD datasets.

AK Shrivas, AK Dewangan (2014)[10], proposed an Intrusion Detection System which is a combination of ANN and Bayesian net classifier and uses the Gain ratio for reducing the feature vector. This model gave an accuracy of 99.42% with KDD99 and 98.07% with the NSL-KDD data set. So we are considering this and providing result which is similar to this model.

H Chae, B jo, SH Choi, T Park (2013)[11], proposed a new feature selection method using feature average of total and each class. They also used a feature reduction algorithm called Decision tree classifier to reduce the dimensionality of the input vector.

C Yin, Y Zhu, J Fei, X He (2017)[12], developed an Intrusion Detection System using Recurrent neural networks. This Intrusion Detection System is trained and tested using the benchmarked NSL-KDD dataset. This model was then compared with the traditional machine learning classifiers like Support Vector Machines, Random Forest, Naive Bayes, and J48. The metrics used for evaluating the RNN-IDS was the detection rate and accuracy. This model shows an accuracy rate of 99.81% and 83.3% on the train and test NSL-KDD datasets.

SM Kasongo, Y Sun (2019)[13], proposed an Intrusion Detection System using the technique of Deep Long Short-Term Memory(DLSTM) for storing the past information without losing it with time. This model outperforms over the methods such as Deep Feed-forward Neural Networks, Support Vector Machines, k-Nearest Neighbors, Random Forests and Naive Bayes. A feature selection algorithm based on information gain was used to reduce the feature vector. To achieve better results Information gain feature selection method was used. The accuracy of this model on the training and testing datasets was 99.51% and 86.99%.

SM Kasongo, Y Sun (2019)[14], a Deep Learning method using feed-forward deep neural networks(FFDNN) besides a feature selection algorithm using information gain(IG) was used. In this work, the FFDNN with IG was evaluated on the nsl-kdd intrusion detection dataset. This model FFDNN-IDS outperforms over various other models like k-Nearest Neighbors(KNN), Naive Bayes, Support Vector Machine(SVM), Random Forest (RF) and Decision Trees(DT). This model shows an accuracy rate of 99.37% and 86.76% on the train and test NSL-KDD datasets.

III. Dataset Description

In our work to deal with the detection of intrusions we have taken the standard NSL-KDD dataset which is an updated version of kdd cup 99. The advantages of the nsl-kdd dataset are
I. The dataset consists of distinct records so that the classifiers will not produce any biased result.
II. No overfitting of the result.
The NSL-KDD dataset is composed of 41 attributes and one categorized attribute. The training is performed on the nsl-kdd train dataset which contains 22 attack types and testing is performed on the nsl-kdd test dataset which contains additional 17 attack types. The attack classes present in nsl-kdd dataset are grouped into four categories
1.Denial of service(DoS): The authorized users will be blocked by intruders from using their service.
2.Probe: This attack collects information about potential vulnerabilities of the target system that can be later used to launch attacks on that system
3.Remote to Local(R2L): Unauthorized users gain privileges as a root user by dumping the data packets to remote systems over a network and do unauthorized activities.
4.User to Root(U2R): Intruders access the administrative privileges by entering into the network as normal users.

IV. Proposed System

We have developed an Intrusion Detection System using a Recurrent Neural Network with the gated recurrent units. The recurrent neural network comprises the input unit, hidden unit, and output units. The hidden unit consists of all mathematical computations. We are taking nsl-kdd dataset as input and it consists of the training and the testing datasets. First the input data has to be pre-processed to remove any irrelevant data and then we applied Feature Selection on the target data to reduce the dimensionality of the input data. Then, we fed this input data to the Recurrent Neural Networks with GRU units to train the GRU-IDS and finally test the proposed model with the nsl-kdd test dataset.

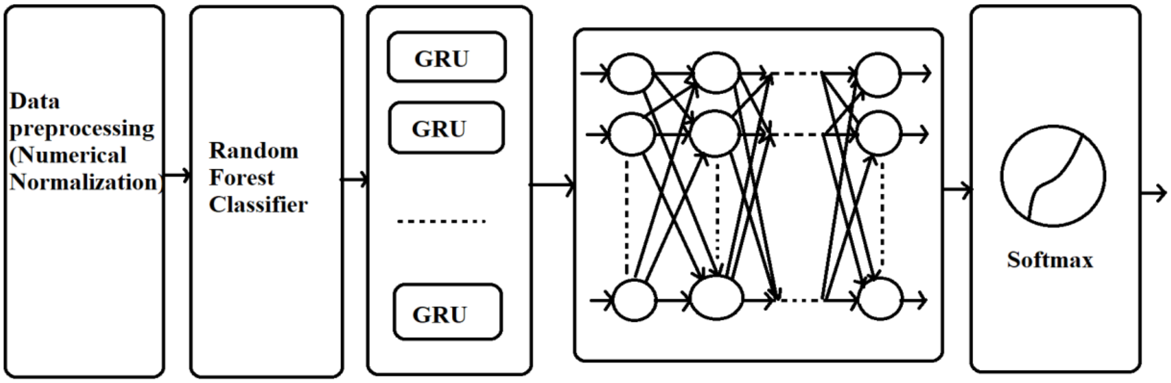


FIGURE 3.Proposed System

4.1 DATA PREPROCESSING

1) Conversion of Non-Numeric values to Numeric values

The GRU-IDS can accept only numeric values as input. The NSL-KDD dataset consists of 41 features out of which 38 are in numerical form and 3 are of string datatype. The non-numeric features are labelled as 'protocol_type', 'service' and 'flag' which are of string type to be converted into numeric form. To do this we used the Hot encoding technique to convert the non-numeric features to numeric features.

2) Normalization

The GRU-IDS works with the input which is only in the range of 0 to 1. As the input data we get is not in the specific range[0-1]. So here we applied a Min-max scaling technique to scale the input data in the range between 0 to 1. The below equation was applied to each input feature in the nsl-kdd dataset.

$$I' = (I - min_j) / (max_j - min_j)$$

In the above equation, I is the unnormalized value of a particular attribute, I' is the changed value of the attribute which is in the normalized form and max_j and min_j are the maximum and minimum values of the j^{th} attribute.

4.2 Feature Selection

The NSL-KDD dataset has 41 attributes and one class attribute. From those 41 attributes, some of the attributes will not be useful in the detection of intrusion. So, we are using the random forest classifier to remove some of the unimportant attributes of the train and test datasets that resolves the problem of overfitting and decrease the training time of the GRU-IDS model.

4.3 Designing of the Gated Recurrent Unit

In our work, the proposed system GRU-IDS takes the nsl-kdd train dataset as a input vector (X_t) and multiplies it with the weight matrix(W_z). From the hidden layer of the previous time step, we take ht-1 as input which gives past information and then it is also multiplied by the weight matrix (U_z). W_z*X_t and U_z*X_t were added together and passed to the SoftMax function to get the update gate's output(Z_t) in the range between 0 to 1. This operation will be useful to prevent the vanishing gradient problem because the model keeps track of all the past information without any loss. In the same way, the reset gate(r_t) is constructed. Now we make use of the reset and update gates in the GRU cell as shown in Figure 2. To store the relevant information from the past we use the reset gate(r_t). First, we multiply X_t with a weight matrix W_h. Secondly, we apply Hadamard product between the reset gate r_t and h_{t-1} and sum the result of Hadamard product with W_h*X_t and apply tanh activation function to the obtained result and store the result in $h_t^{'}$ which stores only the relevant information from the past called as current memory content. Finally we calculate the final memory at time step t. Now we make use of the update gate(Z_t) which consists of the information to be passed at time step t. Calculate element-wise multiplication between Z_t and $h_t^{'}$ and between 1-Z_t and h_{t-1} then sum up both of them and store the result in h_t. The h_t will tell the GRU model how much of the past information to be useful; this will make the GRU model train perfectly without any loss of the past information

V. EVALUATION METRICS

To examine the performance of the GRU-IDS we specifically used Accuracy(AC) as a performance indicator. The other performance measures used are Detection Rate, and False Positive Rate. The output of the GRU-IDS model is categorized based on the following four conditions:

True Positive (TP): The number of anomaly records that are correctly classified as anomaly.
False Positive(FP): The number of normal records that are incorrectly classified as anomaly.
True Negative(TN): The number of normal records that are correctly classified as normal.
False Positive(FN): The number of anomaly records that are incorrectly classified as normal.
From the above-defined TP, FP, TN, FN metrics we can define Accuracy, Detection Rate, and False Positive Rate.

Accuracy(AC): It is the percentage of the number of records that are correctly classified out of the total number of records.

$$\text{Accuracy} = (TP+TN) / (TP+TN+FP+FN)$$

Detection Rate(DR): It is the percentage of the number of records that are classified correctly out of the total number of anomaly records.

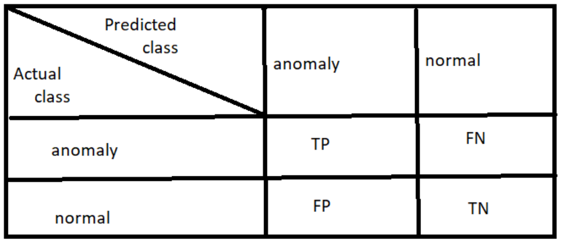
$$\text{Detection Rate(DR)} = TP / TP+FN$$

False Positive Rate(FPR): It is the percentage of the number of records that are incorrectly classified out of the total number of normal records.

$$\text{False Positive Rate(FPR)} = FP / FP + TN$$

The Confusion matrix visualizes the performance of the GRU-IDS model as shown below.

Table 1. Confusion Matrix



Predicted class Actual class	anomaly	normal
anomaly	TP	FN
normal	FP	TN

VI. EXPERIMENTALRESULTS

The experimental results show that our proposed system GRU-IDS gives better accuracy on the test dataset compared to various traditional machine learning classifiers as shown in Table 2. The GRU-IDS also gives more accuracy rate compared to simple RNN and LSTM based techniques. From Table 3, we observe that our proposed system's accuracy varies with the number of hidden nodes present in the hidden layer of recurrent neural networks.

Table 2. Performance of the existing systems.

IDS SYSTEM	Validation Accuracy	Test Accuracy
SVM	99.55%	78.32%
KNN	99.42%	73.26%
NB	89.32%	75.62%
RF	99.73%	83.92%
ANN	99.49%	84.17%
RNN	97.53%	82.74%
LSTM	98.12%	85.42%

Table 3. Performance of our proposed system GRU-IDS.

Hidden Nodes	Validation Accuracy	Test Accuracy
40	95.15%	76.78%
80	99.42%	85.34%
120	99.13%	89.22%
160	96.18%	82.17%
200	97.35%	79.19%

VII. Conclusion and future work

This model mainly focused on Intrusion detection with a high accuracy rate using RNN and feature selection algorithm Random forest classifier. The experimental results shows an accuracy rate of 99.13%.on the training dataset and 89.22% on the test data. This model outperforms all the other existing Intrusion Detection Systems. In our future research, we would like to focus on decreasing the time complexity and increasing the accuracy rate in detecting intrusions in a network system.

VIII. References

[1]. Sharma S, Gupta RK. Intrusion detection system: A review. International Journal of Security and its Applications.2015;9(5):69-76.

[2]. Allen J, Christie A, Fithen W, McHugh J, Picket J. State of the practice of intrusion detection technologies. CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST; 2000 Jan.

[3].Reddy RR, Ramadevi Y, Sunitha KN. Effective discriminant function for intrusion detection using SVM.in 2016 International Conference on Advances in Computing. Communications and informatics (ICACCI) 2016 Sep 21(pp.1148-1153). IEEE.

[4].Farnaaz N, Jabbar MA.Random forest modeling for network intrusion detection system. Procedia Computer Science.2016 Jan 1;89(1):213-7.

[5].Selvakumar B, Muneeswaran K. Firefly algorithm based feature selection for network intrusion detection.Computers & Security. 2019 Mar 1;81:148-55.

[6].Li W, Yi P, Wu Y, Pan L, Li J.A new intrusion detection system based on KNN classification algorithm in the wireless sensor network. Journal of Electrical and Computer Engineering 2014;2014.

[7].Sahu S, Mehtre BM. Network intrusion detection system using J48 Detection Tree. In 2015 International Conference on Advances in Computing, Communications, and Informatics (ICACCI) 2015 Aug 10(pp. 2023-2026). IEEE.

[8].Revathi S, Malathi A. A detailed analysis of NSL-KDD dataset using various machine learning techniques for intrusion detection. International Journal of Engineering Research & Technology (IJERT). 2013 Dec;2(12):1848-53.

[9]In GRE B, Yadav A. Performance analysis of NSL-KDD dataset using ANN. In 2015 international conference on signal processing and communication engineering systems 2015 Jan 2(pp. 92-96).IEEE.

[10]. Shrivas AK, Dewangan AK. An ensemble model for classification of attacks with feature selection based on KDD99 and NSL-KDD data set. International Journal of Computer Applications.2014;99(15):8-13.

[11]. Chae HS, Jo BO, Choi SH, Park TK. Feature selection for intrusion detection using NSL-KDD.Recent advances in computer science.2013 Nov:184-7.

[12]. Yin C, Zhu Y, Fei J, He X. A deep learning approach for intrusion detection using recurrent neural networks. Ieee Access.2017 Oct 12;5:21954-61.

[13]. Kasongo SM, Sun Y.A Deep Long Short-Term Memory based classifier for Wireless Intrusion Detection System. ICT Express. 2019 Aug 22.

[14]. Kasongo SM, Sun Y.A deep learning method with a filter-based feature engineering for the wireless intrusion detection system. IEEE Access. 2019 Mar 18;7:38597-607.