Intrusion Detection System for NSL-KDD Dataset

K. Saranya\textsuperscript{1}, D. Sathya\textsuperscript{2}

M.E. Scholar\textsuperscript{1}, Assistant Professor II\textsuperscript{2}

Department of CSE
Kumaraguru College of Technology, Coimbatore, India

Corresponding Authors: saranyask23@gmail.com\textsuperscript{1}

Abstract

During the last decade, the analysis of intrusion detection has attracted the attention of many researchers focus on different dataset to improve system accuracy and to reduce false positive rate based on DAPRA 98 and later the updated version as KDD cup 99 dataset which has some statistical issues, it degrades the system accuracy and evaluation of anomaly detection that affects the performance of the security analysis which leads to the replacement of KDD dataset to NSL-KDD dataset. In the paper focus on detailed study on NSL- KDD dataset that contains only selected record. The NSL-KDD dataset provide a good analysis on various machine learning techniques for intrusion detection.

Keywords: NSL-KDD, Data Mining Technique and KDD Cup 99

I. INTRODUCTION

The feed forward neural networks have been extensively used in many fields due to their ability: to approximate complex nonlinear mappings directly from the input samples and to provide models for a large class of natural and artificial phenomena that are difficult to handle using classical parametric techniques. On the other hand, there lack faster learning algorithms for neural networks. The traditional learning algorithms are slower than required process. It is not surprising to see that it can take several hours, several days, and even more
time to train neural networks by using traditional methods.

In real applications, the feed forward neural networks are trained in finite training set. For function approximation in a finite training set, shows that a single-hidden layer feed forward neural network (SLFN) with at most N hidden nodes and with almost any nonlinear activation function can exactly learn N distinct observations. It should be noted that input weights and hidden layer biases need to be adjusted in all these previous theoretical research works as well as in almost all practical learning algorithms of feed forward neural networks.

Traditionally, all the parameters of the feed forward networks are needed to be tuned and thus there exists the dependency between different layers of parameters (weights and biases). During the past decades, the gradient descent-based methods have mainly been used in various learning algorithms of feed forward neural networks. It is clear that gradient descent-based learning methods are generally very slow due to improper learning steps or may easily converge to local minima and many iterative learning steps may be required by learning algorithms in order to obtain better learning performance.

SLFNs (N hidden nodes) with randomly chosen input weights and hidden layer biases (and such hidden nodes can thus be called random hidden nodes) can exactly learn N distinct observations. Unlike the popular thinking and most practical implementations that all parameters of the feed forward networks need to be tuned, one may not necessarily adjust the input weights and first hidden layer biases in applications. In fact, some simulation results on artificial and real large applications in our work [7] have shown that the method not only makes learning extremely fast but also produces good generalization performance.

In the paper, first rigorously prove that the input weights and hidden layer biases of SLFNs can be randomly assigned if the activation functions in the hidden layer are infinitely differentiable. After the input weights and hidden layer biases are chosen randomly, SLFNs can be simply considered as a linear system and the output weights (linking the hidden layer to the output layer) of SLFNs can be analytically determined through simple generalized inverse
operation of the hidden layer output matrices. Based on the above concept, the paper proposes a simple learning algorithm for SLFNs called extreme learning machine (ELM) whose learning speed can be thousands of times faster than traditional feed forward network learning algorithms like back-propagation (BP) algorithm while obtaining better generalization performance.

Different from the traditional learning algorithms the proposed learning algorithm not only tends to reach the smallest training error but also the smallest norm of weights. Bartlett’s theory on the generalization performance of feed forward neural networks states to reaching smaller training error, the smaller the norm of weights is, the better generalization performance the networks tend to have [7]. Therefore, the proposed learning algorithm tends to have good generalization performance for feed forward neural networks.

The new proposed learning algorithm can be easily implemented, and tends to reach the smallest training error, obtains the smallest norm of weights and the good generalization performance, and runs extremely fast, in order to differentiate it from the other popular SLFN learning algorithms, it is called the extreme learning machine in the context of the paper.

The rest of the paper is structured as follows: section II present some related works based on intrusion detection research. Section III explains the detailed description of the attacks present in NSL-KDD dataset. Section IV summarize in detail about analysis of NSL KDD dataset on various data mining technique. Section V explains the experimental analyses on various attacks using different machine learning techniques. The conclusion and future works is summarized in section VI.

II. RELATED WORK
KDD’99 [3] has been the most widely used data set for the evaluation of anomaly detection methods. The data set is prepared by Stolfo et al [5] and it is built based on the data captured in DARPA’98 IDS evaluation program [6]. DARPA’98 is about 4 gigabytes of compressed raw (binary) tcp dump data of 7 weeks of network traffic, which can be processed into about 5 million connection records, each with about 100 bytes. The two weeks of the test data have around 2 million connection records. KDD
training dataset consists of approximately 4,900,000 single connection vectors each of which contains 41 features and is labelled as either normal or an attack, with exactly one specific attack type [8]. The simulated attacks fall in the following four categories:

1) Denial of Service Attack (DoS):
It is an attack in which the attacker can makes some computing or memory resource too busy or too full to handle legitimate requests, or denies legitimate users access to a machine.

2) User to Root Attack (U2R):
It is a class of exploit in which the attacker starts out with access to a normal user account on the system (perhaps gained by sniffing passwords, a dictionary attack, or social engineering) and it is able to exploit some vulnerability to gain root access to the system.

3) Remote to Local Attack (R2L):
It occurs when an attacker who has the ability to send packets to a machine over the network but who does not have an account on that machine exploits some vulnerability to gain local access as a user of that machine.

4) Probing Attack:
It is an attempt to gather information about a network of computers for the apparent purpose of circumventing its security controls.

The analysis showed that there are two important issues in the data set which highly affects the performance of evaluated systems, and results in a very poor evaluation of anomaly detection approaches. To solve these issues, a new data set, NSL-KDD [24], which consists of the selected records of the complete KDD99 dataset [16]. The data set is publicly available for researchers through our website and has the following advantages over the original KDD data set:

- It does not include redundant records in the train set, so the classifiers will not be biased towards more frequent records.
- There are no duplicate records in the proposed test sets. The performance of learners is not biased by the methods which have better detection rates on the frequent records.
• The number of the selected records from each difficulty level group is inversely proportional to the percentage of records in the original KDD data set. As a result, the classification rates of distinct machine learning methods vary in a wider range, which makes it more efficient to have an accurate evaluation of different learning techniques.

The number of the records in the train set and test set are reasonable, which makes it affordable to run the experiments on the complete set without the need to randomly select a small portion. The evaluation results of different research works will be consistent and comparable.

III. DATA SOURCE AND SELECTION
The NSL-KDD training data set is collected from secondary sources (http://nsl.cs.unb.ca/NSL-KDD/) [5]. There are 43 numbers of columns including the class label and 12,5773 numbers of rows. This data set is originated from KDD’99 data set which is the popularly used training data set (Olusola 2010). NSL-KDD is updated from KDD 99 data set by removing some redundant records and the number of columns or feature in both the data set is same (Tavallaee 2009). Among the 43 columns there are many features or columns which are irrelevant or weakly relevant. Therefore it is required to remove or delete those features from the data set, which will reduce the size of the data set and make the analysis simpler (Han 2006).

The training dataset is made up of the 21 different attacks out of the 37 present in the test dataset. The known attack types are those present in the training dataset while the novel attacks are the additional attacks in the test dataset i.e. not available in the training datasets. The attack types are grouped into four categories: DoS, Probe, U2R and R2L. NSL-KDD data set [3] is a refined version of its predecessor. It contains essential records of the complete KDD dataset [18]. There are a collection of downloadable files at the disposal for the researchers. They are listed in the Table 1

1) The redundant records are removed to enable the classifiers to produce an un-biased result.

2) Sufficient number of records is available in the train and test data
sets, which is reasonably rational and enables to execute experiments on the complete set.

3) The number of selected records from each difficult level group is inversely proportional to the percentage of records in the original KDD data set.

Table 1: Mapping of Attack Class with Attack Type

<table>
<thead>
<tr>
<th>Attack Class</th>
<th>Attack Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>DoS</td>
<td>Back, Land, Neptune, Pod, Smurf, Teardrop, Apache2, Udpstorm, Processtable, Worm (10)</td>
</tr>
<tr>
<td>Probe</td>
<td>Satan, Ipsweep, Nmap, Portsweep, Mscan, Saint (6)</td>
</tr>
<tr>
<td>R2L</td>
<td>Guess_Password, Ftp_write, Imap, Phf, Multihop, Warezmaster, Warezclient, Spy, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Httptunnel, Sendmail, Named (16)</td>
</tr>
<tr>
<td>U2R</td>
<td>Buffer_overflow, Loadmodule, Rootkit, Perl, Sqlattack, Xterm, Ps (7)</td>
</tr>
</tbody>
</table>

In each record there are 41 attributes unfolding different features of the flow and a label assigned to each either as an attack type or as normal. The 42nd attribute contains data about the various 5 classes of network connection vectors and they are categorized as one normal class and four attack class. The 4 attack classes are further grouped as DoS, Probe, R2L and U2R.

IV. DATA MINING TECHNIQUES
Using the data processing techniques, it can perceive and extrapolate knowledge that may scale back the probabilities of fraud detection[9], improve audit reactions to potential business changes, and make sure that risks area unit managed in exceedingly a lot of timely and active manner.

Additionally to employing a specific data processing tool, internal auditors will choose between the ranges of knowledge mining techniques [13]. The foremost unremarkably used techniques embody artificial neural networks, support vector machine, and extreme learning machine. Each of the techniques are analysed the knowledge in numerous ways:

The Feed forward neural networks [1] have been extensively used in many fields due to their ability:

1) To approximate complex nonlinear mappings directly from the input samples; and

2) To provide models for a large class of natural and artificial phenomena that are difficult to handle using classical parametric techniques. all the parameters of the feed forward networks need to be tuned and thus there exists the dependency between different layers of parameters (weights and biases).
Figure 4.1 shows workflow of NSL-KDD dataset

The SVM embodies many important principles. It solves the problem of classification directly without trying to solve the much harder problem of estimating the distribution of data samples [3]. The SVM uses two main ideas. First, kernel functions are used to transform the problem from the original input space into a highly dimensional one, called the feature space, where linear separation of training samples belonging to different classes is possible. Second, to the best separating hyperplane, the concept of maximum margin is introduced. Finally, the optimization problem which defines the SVM is convex and quadratic, and therefore it can be solved efficiently.

In proposed paper, a simple learning algorithm for SLFNs called extreme learning machine (ELM) whose learning speed can be thousands of times faster than traditional feedforward network learning algorithms [1] like back-propagation (BP) algorithm while obtaining better generalization performance. The input weights and hidden layer biases of SLFNs can be randomly assigned if the activation
functions in the hidden layer are infinitely differentiable. After the input weights and the hidden layer biases are chosen randomly that are linking the input layer to the hidden layer, SLFNs can be simply considered as a linear system and the output weights (linking the hidden layer to the output layer) of SLFNs can be analytically determined through simple generalized inverse operation of the hidden layer output matrices.

V. EXPERIMENTAL RESULT AND ANALYSIS

The experimental steps are as follows:

1) Select and preprocess the dataset.

2) Run the classifier algorithm.

3) Compare the classifier result.

The first step, to perform the discretization as preprocess. Discretization is the process of turning numeric attributes into nominal attributes. The main benefit is that some classifiers can take only nominal attributes as input, not numeric attributes. The other advantage is that some classifiers can take numeric attributes can achieve improved accuracy if the data is discretized prior to learning.

In the performance of the proposed ELM learning algorithm is compared with the popular algorithms of feedforward neural networks like the conventional BP algorithm and support vector machines (SVMs) on quite a few benchmark real problems in the function approximation and classification areas.

The simulations for BP and ELM algorithms are carried out in MATLAB environment running in a Pentium 4, 1.9 GHZ CPU. Although there are many variants of BP algorithm, the faster Back propagation algorithm called Levenberg–Marquardt algorithm is used in our simulations.

The HELP of MATLAB package and tested on many benchmark applications among all traditional BP learning algorithms, the Levenberg–Marquardt algorithm appears to be the fastest method for training moderate sized feedforward neural networks (up to several hundred weights). It has a very efficient implementation of Levenberg–Marquardt algorithm provided by
MATLAB package, that has been used in our simulations for BP. The simulations for SVM are carried out using compiled C-coded SVM packages: LIBSVM running in the same PC. The kernel function used in SVM is radial basis function whereas the activation function used in the proposed algorithms is a simple sigmoidal function.

In the experiments, ELM algorithm that the learning time of ELM is mainly spent on calculating the Moore–Penrose generalized inverse \( H \) of the hidden layer output matrix \( H \).

**Accuracy**
The accuracy is the proportion of true results (both true positives and true negatives) among the total number of cases examined. Accuracy can be calculated from formula given as follows:

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}
\]

*Figure 6.1 shows comparison of Accuracy*
Figure 6.1 shows that the comparison of BPN Classification, SVM Classification, ELM Classification in terms of accuracy values. The result shows that the ELM Classification approach provides higher accuracy value than other classification.

B. Precision

Precision value is evaluated according to the feature classification at true positive prediction; false positive. It is expressed as follows:

\[
\text{Precision} = \frac{\text{True positive}}{\text{True positive} + \text{False positive}}
\]

Figure 6.2 shows that the comparison of BPN Classification, SVM Classification, ELM Classification in terms of precision values. The result shows that the ELM Classification approach provides higher precision value than other classification.

Figure 6.2 shows comparison of Precision
C. Recall
Recall value is evaluated according to the feature classification at true positive prediction, false negative. It is given as,

\[
\text{Recall} = \frac{\text{Truepositive}}{\text{Truepositive} + \text{Falsenegative}}
\]

Figure 6.3 show that the comparisons of BPN Classification, SVM Classification, and ELM Classification in terms of recall values. The result shows that the ELM Classification approach provides higher recall value than other classification.

D. F-Measure
F-measure is calculated from the precision and recall value. It is calculated as:

\[
f - \text{measure} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}
\]

Figure 6.4 show that the comparisons of BPN Classification, SVM Classification, and ELM Classification in terms of F-Measure values. The result shows that the ELM Classification approach provides higher F-Measure value than other classification.

\textit{Figure 6.3 shows comparison of Recall}
Figure 6.4 shows comparison of F-Measure

Table 2: Comparison Table for BPN Classification, SVM Classification, ELM Classification
CONCLUSION AND FUTURE WORK
The proposed paper a simple and efficient learning algorithm for single-hidden layer feedforward neural net-works (SLFNs) called extreme learning machine (ELM), which has also been rigorously proved in this paper. The proposed ELM has several interesting and significant features different from traditional popular gradient-based learning algorithms for feedforward neural networks:

- The proposed ELM has better generalization performance than the gradient-based learning such as back-propagation in most cases.

- The learning speed of ELM is extremely fast.

- It should be worth pointing out that gradient-based learning algorithms like back-propagation can be used for feedforward neural networks which have more than one hidden layers while the proposed ELM algorithm at it is present form still only valid for SLENs.

The SVM algorithm to external new unknown observations is much slower than feedforward neural networks since SVM algorithms normally generate much larger number of support vectors (computation units) while feedfor-ward neural networks require very few hidden nodes (computation units) for same applications. It is not easy for SVMs to make real-time predication in this application since several hours may be spent for such prediction (testing) set, while the ELM appears to be suitable in applications which request fast prediction and response capability.

The paper has demonstrated that ELM can be used efficiently in many applications, however, two more interesting aspects are still open: the universal approximation capability of ELM and the performance of ELM in sparse high-dimensional applications, which are currently under our investigation.

REFERENCES
1) L.Dhanabal, Dr. S.P. Shantharajah, “A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms”, International Journal of Advanced


18) Shahaboddin Shamshirband, Nor Badrul Anuar, Miss Laiha Mat Kiaha, Vala Ali Rohani, Dalibor Petković, Sanjay Misrac, Abdul Nasir Khana “Co-FAIS: Cooperative fuzzy artificial immune system for detecting intrusion in wireless sensor networks”, Journal of

