# Improvising an Intrusion Detection Precision of ANN Based Hybrid NIDS by incorporating Various Data Normalization Techniques - A Performance Appraisal

**A.M.Chandrashekhar[1], K. Raghuveer[2]**

[1]**Department of Computer Science, Sri Jayachamarajendra College of Engineering (SJCE), Mysore - 570006, Karnataka, India**

[2]**Department of Information Science, The National Institute of Engineering (NIE), Mysore-570008, Karnataka, India**

*Abstract* — **Intrusion Detection Systems (IDS) are becoming an essential component usually in network and data security weapon store. In recent days hybrid network IDS is trend in IDS development, Data mining techniques, predominantly supervised and unsupervised machine learning techniques play a major role in IDS development. Most of the researchers involved in IDS are using Offline dataset called KDD Cup 99 dataset. This dataset consists of symbolic, binary, numeric, and continuous data types scattered in different range of values. Machine learning techniques used in IDS development can't process the data as it is: clustering algorithms work only with numeric data and also in clustering algorithms. Such disadvantages could be conquered by Normalization, ensuring homogeneity to the dataset while preserving the correctness of the features mapped**

**In this paper, different linear and non-linear data normalization methods are applied independently on intrusion detection data set so as to get normalized dataset. This normalized dataset data set is then given as input to network IDS model developed using machine learning techniques to check the intrusion detection accuracy. The output results are compared so as to find the more relevant normalization technique for intrusion detection dataset. From the analysis, it was found that different normalization techniques are suitable for IDS dataset. From the experimentation results it is proved that the Z-score (98.46%), Logarithmic (97.84%) and Decimal Scaling (97.08%) normalization techniques result in better detection rate. Among these three, Logarithmic technique takes less time to detect intrusion.**

***Keywords - Data normalization, Intrusion detection, min-max, Z-score, Decimal scaling, Logarithmic, MAD, KDD cup 99 dataset.***

## I. INTRODUCTION

In large real-world databases, inconsistent, incomplete and noisy data are universal. Element of interest may not always be existing and extra data was incorporated just because it was deem to be important during data entity. In knowledge discovery process, prior to data mining itself, data preprocessing plays a crucial role. During data mining process, data need to be prepared or preprocessed before applying mining algorithms or building models. The main goal of data preparation is to promise the quality of the data prior to apply any learning algorithms. Preprocessing will ease the task of mining algorithms and also influence algorithm's performance. There is a bundle of data preprocessing techniques; among them data transformation is predominant. Through data transformation, the data are renovated or consolidated into forms appropriates for mining. Data transformation can involve activities such as smoothing, aggregation, generalization, normalization and attribute construction. Data transformation operations such as normalization and aggregation can contribute toward the success of mining algorithms. Normalization is the most widely used data transformation technique.

Data normalization is a fundamental preprocessing step for mining and learning from data [1].The term normalization is used in various contexts, with distinct, but connected, meanings. In principle normalizing means renovating so as to render normal. If data is noticed as vectors, normalization means changing the vector so that it has unit norm. When data is thought of as random variables, normalization means changing it into normal distribution. When data is hypothesizing to be normal, normalizing means change it into unit variance.

In simple words, intrusion is an illegitimate act of entering, get hold of, or taking possession of another's property (computer system). An intrusion detection system (IDS) is a piece of equipment or software that supervises network or system's activities for malicious actions or policy violations and generates reports to administrator. With the rapid improvement in the network technology including higher bandwidth and simplicity in connecting wireless and mobile devices, Intrusion detection protection systems have become a necessary addition to the security infrastructure of nearly every organization. . Intrusion detection has been an active field of research for about three decades, starting in1980. This is primarily because there has been ever-increasing concern to safeguard the immense data stored in a network from malicious amendments and disclosure to unauthorized folks. In recent days, upon encountering huge network traffic leads to an outsized datasets. In view of this, lots of data mining techniques have been introduced to solve the problem of data analysis. In recent years, more intelligence is brought into IDS by means of machine learning [2].Artificial neural network (ANN) is proved to be successful in solving many complex practical problems due to encounter of large traffic data set. Based on a study of latest research articles, there are numerous researches that attempts to link data mining and machine learning techniques to IDS so as to devise more intelligent IDS model. Currently the support vector learning technique is featuring superior [3].

The rest of the text is ordered as follows: The effect of normalization on Machine learning is presented in section 2. Different Normalization techniques are discussed in section 3.The detailed experimental setup, data set description, evaluation criterion; results and comparative study are discussed in section 4. The conclusions are summed up in section 5.

## II. Normalization effect on Machine learning

Normalization is a course of action followed to bring the data nearer to the requisite of the algorithms. A data element is normalized by scaling its values with the intention that they fall within small specific intervals, such as 0 to 1 or -1 to +1 and so on. Generally normalization will be applied on data elements when all their attributes have the same domain. Normalization may boost the accuracy and efficiency of mining algorithms involving distance measurements. For an un-normalized data, when squared distance between two data instances are calculated using Euclidean distance calculation we can observe large deviation between instances which are statistically in the same category. This is due to wide range of values of the attributes, which necessitates the need of normalization. Normalization is mainly useful for classification algorithms relating neural networks, or distance measurements such as clustering and nearest-neighbor classification. If back-propagation algorithm is used in neural network for classification mining, normalizing the input values will help to speed up the learning phase [4].

Neural network training becomes more efficient if normalization step is performed on the network inputs and targets. For example, in multilayer networks, generally sigmoid transfer functions are used in the hidden layers. These functions turn out to be essentially saturated as soon as the net input is greater than three (exp (−3) ≅ 0.05). In the beginning of training process if this happens, the gradients will be very small, and the network training will be very slow. The net input is a product of the input times the weight plus the bias in the first layer of the network. If the input is outsized, then the weight must be very small so as to avoid the transfer function from becoming saturated. Before applying to the network, it is customary practice to normalize the inputs. Commonly normalization step is applied to both the input and target vectors in the data set. In this fashion, the network output constantly falls into a normalized range. The output of network can then be reverse transformed back into the units of the original target data when the network is put to use in the field [5].

## III. Normalization techniques

At the outset, the data normalization Techniques have been classified into two feature based and Vector based [6]. Since KDD data is feature based, our spotlight is on feature based normalization techniques. Feature based schemes are further divided into Linear and Non linear techniques. This division is based on distribution of data around the mean. Figure 1 gives the list of relevant Linear and non linear normalization Techniques useful for the data set considered in this paper.

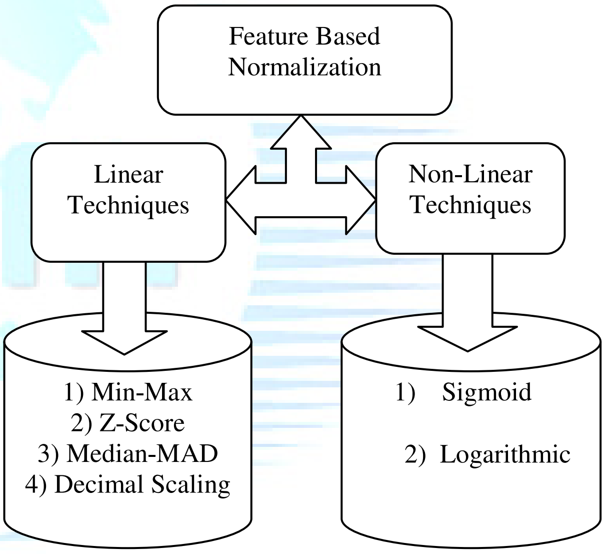

Fig. 1: Classification of Normalization Techniques.

### A. Min Max Normalization Technique

Basically, linear transformation on the original data is been done in Min max normalization. Through this relationship among original data are preserved. Min-Max normalization transforms a value P to Q which fits in the range [M,N]. It is formulated as shown below

$$Q = \left(\frac{(\mathrm{P} - \mathrm{Min(P)})}{Max(P) - \ Min(P)}\right) * (\mathrm{N} - \mathrm{M}) + \mathrm{M} \qquad (1)$$

In the equation (1), Value of Feature P needs to be Normalized into value Q. Min(P) and Max(P) represents Minimum and Maximum values in the Feature P respectively. M and N corresponds to Lower value and upper Value in the new range.

### B. Z-Score Normalization

Z-score normalization is also known zero-mean normalization. In Z-score normalization, The value of an attribute P is normalized into Q based on the mean and standard deviation of attribute of by using equation (2)

$$Q = \left(\frac{(\mathrm{P} - \mathrm{Mean}(\mathrm{P})}{Standard\ Deveation(P)}\right) \qquad (2)$$

The z-score method, in contrast, is useful when the maximum and minimum value of an attribute is unknown or when there are outliers that dominate the min-max normalization.

### C. Median & Median absolute Deviation Technique

This Normalization Technique is a revised form of Z-Score normalization technique and is appropriate if the feature contains type of values in the given range. Normalization of P into Q is done by using following equation (3).

$$Q = \left(\frac{(\mathrm{P} - \mathrm{Median}(\mathrm{P})}{Median(|P - Median(P)|)}\right) \qquad (3)$$

When data are skewed, then median is a premium measure of central tendency. The traditional method for computing the median involves sorting the data first. In case, if there is odd number of values, then median is the middle value. Otherwise, the median is the average of the two middle records. Median (|P - Mean(P)|) is called as Median Absolute Deviation(MAD) and also referred as mean deviation. In other words, it is the average distance of the data set from its mean.

### D. Decimal Scaling Normalization Technique

This technique normalizes the data by shifting the decimal point of value of an attribute P. The number of decimal points shifted depends on the maximum absolute value of P. Normalization of P into Q is computed by using following equation (4).

$$Q = \left(\frac{\mathrm{P}}{10^k}\right) \qquad (4)$$

In this technique, the computation is generally scaled in terms of decimals. It means that the result is generally scaled by dividing it with $10^k$ where, k is the smallest integer such that k= $\log_{10}$Max(Q).

### E. Sigmoid Technique of Normalization

This non-linear technique ensures proper mapping of larger data values in to the range of 0 and 1 along with normalization. The transformation is approximately linear in the middle range around mean value, and has a smooth nonlinearity at the end which guarantees that all values are within the range. Values away from the mean are squashed exponentially [7].

$$Q = \frac{1}{1 + \mathrm{e}^{-k}} \ , \mathrm{Where} \quad \mathrm{k} = \frac{\mathrm{P} - \mathrm{Mean}(\mathrm{P})}{\lambda * \mathrm{Std.dev}(\mathrm{P})} \qquad (5)$$

Normalization of P into Q is done by using above equation (5), in which λ is user defined value.

### F. Logarithmic Normalization technique

This non-linear normalization technique gives more promise to lower feature values. If the minimum values are known well in advance it is preferable to use Logarithmic normalization [7].

$$Q = Log(P - Min(P) + 1) \qquad (6)$$

Logarithmic normalization of P into Q is done by using above equation (6).

## IV. Experimental Setup and Results Evaluation

In this section, we elaborate our experimental setup and dataset description. Framework used in this paper for experimentation is shown in figure 2. We considered corrected KDD cup 99 data set as input data and its detailed explanation is available in the section IV. KDD dataset is pre-processed as it includes continuous, discrete, and symbolic attributes. These attributes can't be applied directly for classification and another reason is most of clustering algorithms work with continuous (numerical) data. During pre-processing, all the symbolic attributes are removed and only continuous attributes are extracted. Hence, the number of attributes in each of the data vector is reduced from 41 to 34. In intern helps the intrusion detection process becomes easier and less complex and also yields a better result. These vectors of data records are fed into normalization process so as to get normalized data set, which can be given as input to Hybrid network Intrusion detection system (NIDS) model which is explained in section IV. This engine detects intrusions if any in the data supplied to it.

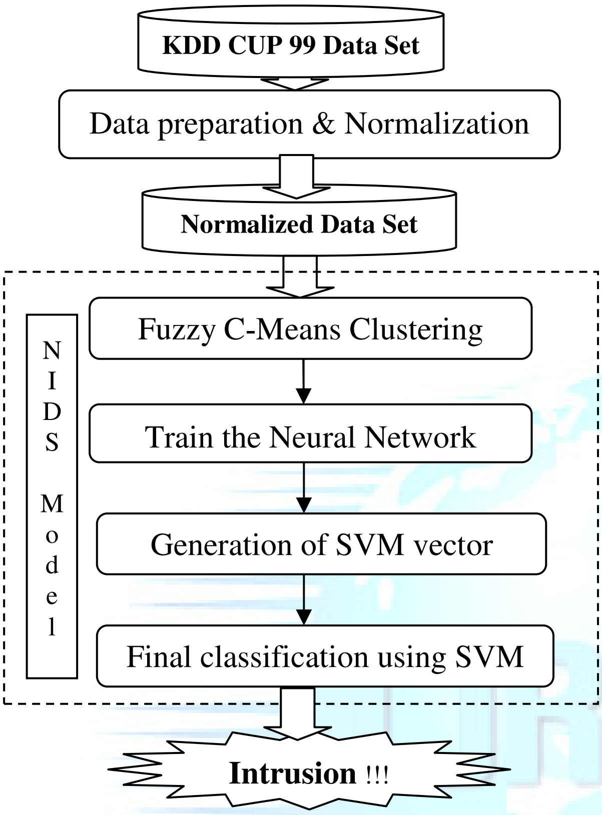

**Fig 2.** Block Diagram of Experimental framework

## A. KDD Cup 99 Data Set

KDD CUP 99 Data set is offline network data based on original 1998 DARPA intrusion detection evaluation program, which is prepared and managed my MIT Lincoln Laboratory**.** This dataset is one of the most rational publicly available data set that includes actual attacks [8]. It provides benchmark to the researchers to evaluate intrusion detection using offline data. This dataset has 48,98,430 single connection records with each connection record has 41 features/attributes and one class attribute. Class attributes labels connection as normal or anomaly with exactly one specific attack types. Features numbered 1-9 are basic features, 10-22 are content features, 23-31 are traffic features and 32-41 stands for host based features. There are totally 37 different types attacks which fall underneath four main categories: PROBE (Probing), denial of service (DOS), user to root(U2R) and remote to local(R2L) [9]. A complete listing of the set of features given in KDD Cup 99 dataset defined for the connection records and types of attacks falling into four major categories are given in [10]. Since corrected KDD cup 99 dataset is of enormous size, it requires high-end machines to perform experimentation. Because of this, a subset of 10% of corrected KDD Cup dataset is utilized for our experimentation. The sample of single connection record of attack type Probe is given below (Un-Normalized record)

5059,1,14,1,5133876,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.0,0.00,0.0
0,0.0,1.00,0.00,0.00,0,0,1.00,0.00,1.00,0.17,0.00,0.0,0.00,0.00,Probe.

## B. Hybrid NIDS Model

In this paper we are employing hybrid network intrusion detection (NIDS) model as a tool to validate the effect of Normalization Techniques considered. This NIDS model is build by assimilation of competent data mining techniques such as appropriate clustering technique, Multilayer perception (MLP) neural network and support vector machine (SVM), which is significantly improvises the prediction of network intrusions. Clustering is valuable in intrusion detection as malevolent activity should group together, unraveling itself from non-malicious activities. Since the number of clusters desired for intrusion detection problem is previously known and is fixed, we employed FCM clustering technique to segment different attack data present in input dataset. The key idea of using neural network for IDS is to take the advantage of classification skill of supervised learning based neural network and clustering skill of unsupervised learning based NN. The detailed justification for selecting these data mining techniques for NIDS model is available in our research papers [11].

The Hybrid NIDS model comprises of Four phases namely: (1) Clustering using Fuzzy C-Means Clustering, where the input data set is grouped into ‘k’ clusters, where ’k’ is the number of clusters preferred, (2) Neural network training, in which all the data in a particular cluster is trained with the respective neural network associated with each of the cluster, (3) SVM vector generation for SVM classification, which contains attribute values obtained by passing each of the data through all of the trained neural networks, and an added attribute which is a membership value of each of the data and (4) final classification using SVM to detect intrusion.

## C. Experiments and Evaluation Metrics

The experiments and evaluations were performed with 10% KDD cup 99 intrusion detection dataset, by using MATLAB version R2013a on a Windows PC with 3.20 GHz CPU and 4GB RAM. We performed the experiments in two phases: training phase and testing phase. The number of data records taken for training and testing phase is given in table 1. Totally, in training and testing , we considered 26114 and 27112 data records respectively.

Table 1. Data Size taken for the experiment.

| | Normal | DOS | PROBE | R2L | U2R |
|---|---|---|---|---|---|
| Train | 12500 | 12500 | 2053 | 38 | 21 |
| Test | 12500 | 12500 | 2054 | 39 | 21 |

Standard parameters Accuracy or detection rate is used to estimate the performance of NIDS. Accuracy measures the degree of faithfulness; it is a proposition of true results. It can be computed by using equation (7)

$$Accuarcy = \frac{(\text{TN+TP})}{\text{TN+TP+FN+FP}} \qquad (7)$$

**Table 2:** Accuracy obtained in 6 Experiments

| | *Attacks* | *Training* | *Testing* | *Average* |
|---|---|---|---|---|
| **Decimal Scaling** | DOS | 0.95424 | 0.95736 | **0.9558** |
| | PROBE | 0.94345 | 0.95108 | **0.94726** |
| | R2L | 0.99832 | 0.98142 | **0.98987** |
| | U2R | 0.99792 | 0.98275 | **0.99034** |
| **MAD** | DOS | 0.95112 | 0.48600 | **0.71866** |
| | PROBE | 0.92744 | 0.95533 | **0.94139** |
| | R2L | 0.98277 | 0.96650 | **0.97464** |
| | U2R | 0.98251 | 0.96726 | **0.97488** |
| **Min- -Max** | DOS | 0.50000 | 0.50000 | **0.50000** |
| | PROBE | 0.14113 | 0.14107 | **0.1411** |
| | R2L | 0.00311 | 0.00303 | **0.00307** |
| | U2R | 0.00168 | 0.00168 | **0.00170** |
| **Z-score** | DOS | 0.99308 | 0.99084 | **0.99196** |
| | PROBE | 0.98667 | 0.96956 | **0.97812** |
| | R2L | 0.98628 | 0.98206 | **0.98417** |
| | U2R | 0.98610 | 0.98219 | **0.98415** |
| **Logari- -thmic** | DOS | 0.97964 | 0.97100 | **0.97532** |
| | PROBE | 0.98811 | 0.97087 | **0.97949** |
| | R2L | 0.98756 | 0.97153 | **0.97954** |
| | U2R | 0.98746 | 0.97229 | **0.97984** |
| **Sigmoid** | DOS | 0.54164 | 0.99010 | **0.76588** |
| | PROBE | 0.99127 | 0.97430 | **0.98282** |
| | R2L | 0.99338 | 0.97991 | **0.98661** |
| | U2R | 0.99329 | 0.98033 | **0.98682** |

The overall performance of our model is determined by taking the average of accuracy results obtained in training and testing phases. A confusion matrix [12] shows the number of proper and improper predictions made by the model compared with the concrete classifications in the test data. It is a table with two rows and two columns that reports the number of false positives (FN), false negatives (FP), true positives (TP), and true negatives (TN).

## D. Experimental Results

As mentioned earlier, we conducted 6 major experiments by employing 6 different Normalization Techniques to obtain related Normalized dataset. These datasets are then fed into NIDS model to check the accuracy of Detection. The related confusion matrix values obtained and are used to compute performance measures accuracy. These results are tabulated in table 2. The time taken by 6 experiments and Consolidated accuracy by considering all types of attacks as one attack are tabulated in table 3.

**Table 3:** Time Taken for Execution of NIDS model

| | **Normalization Algorithms** | **Time taken Sec*e+04** | **Average Accuracy (%)** |
|---|---|---|---|
| 1 | decimal-scaling | 3.1115 | 97.08 |
| 2 | MAD | 1.9016 | 90.24 |
| 3 | min-max | 1.8375 | 64.59 |
| 4 | z-score | 3.355 | 98.46 |
| 5 | Logarithm | 2.7029 | 97.84 |
| 6 | Sigmoid | 1.9130 | 93.05 |

## E. Result Analysis

From Table 3 we can observe that data set generated by Z-score normalization achieved highest accuracy (98.46%).Dataset Obtained from Logarithmic and decimal scaling techniques gives close results i.e 97.84% and 97.08% respectively. Min-Ma Algorithm generated data set produces less accuracy (64.59%) compare to its counterparts but time taken to produce the result is best compare to other techniques. Mean median and sigmoid techniques datasets takes almost same time to produce results. Both the Decimal scaling and Z-score technique generated datasets takes more time to produce the results. The graphical representation of comparisons of Accuracy and Time taken are given in Fig 3 and 4 respectively.

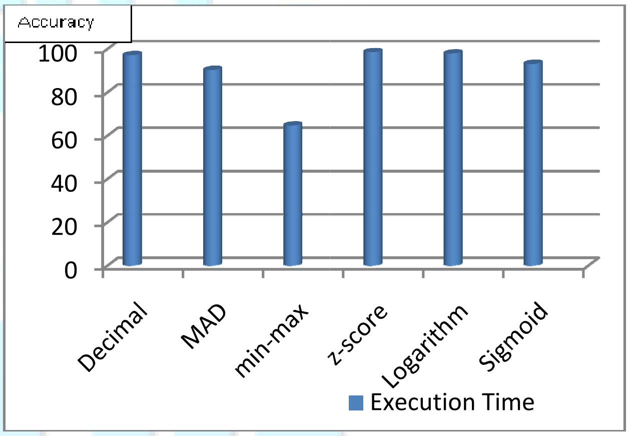

**Fig. 3:** Intrusion Detection Accuracy Comparison

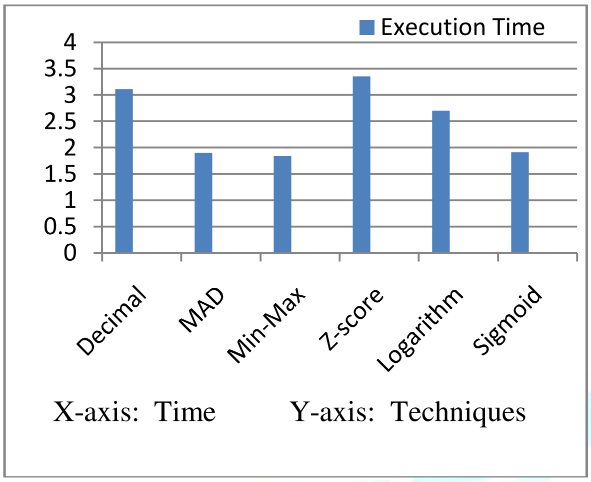

Fig. : Detection time Comparison

We compared the efficiency of data normalization techniques with respect to individual Attack types such as DOS, Probe, R2L and U2R to know which algorithm performs better to detect a particular type of attack. Table 4 depicts the Accuracy values obtained in all the six algorithms for types of attacks. Figure 5 shows the graphical representation of the comparison table 4.

**Table 4:** Attack wise performance comparison of algorithms

| **ALGORITHMS** | **DOS** | **Probe** | **R2L** | **U2R** |
|---|---|---|---|---|
| **Decimal-scaling** | 95.58 | 94.73 | **98.99** | **99.03** |
| **MAD** | 71.87 | 94.14 | 97.46 | 97.49 |
| **min-max** | 50.00 | 14.11 | 00.31 | 00.17 |
| **z-score** | **99.20** | 97.81 | 98.42 | 98.42 |
| **Logarithm** | 97.53 | 97.95 | 97.95 | 97.98 |
| **Sigmoid** | 76.59 | **98.28** | 98.66 | 98.68 |

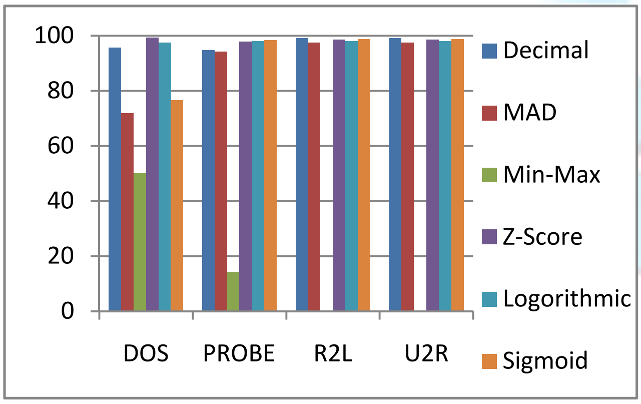

**Fig. 5:** Attack wise performance of algorithms.

From Table 4, we can conclude that high Detection accuracy can be obtained for DOS Intrusions (99.20%) if Z-score algorithm is used in Normalization. For Probe attacks, an application of Sigmoid Normalization technique improves the detection accuracy (98.28%). Application of decimal scaling algorithm helps to improvise the accuracy of detection in case of less frequent attacks such as U2R and R2L (98.99% and 99.03% respectively)

## V. CONCLUSION

Since huge amount of existing off-line data and newly appearing network records that needs analysis, data mining techniques play a vital role in development of IDS. Normalization is a Practice where the data are mapped so that as to fall within a particular range. This paper presents our experimental work on performance evaluation of Linear and nonlinear normalization Techniques. Here we did comprehensive analysis on individual normalization technique to study their impact on machine learning techniques, which intern influence the performance of IDS in terms of accuracy and time taken to detect intrusion. To study the impact of normalization on IDS, we utilized an Hybrid NIDS model as a Tool to measure the performance of IDS system, which intern judge the efficiency of Normalization technique.

A number of observations and conclusions are drawn from the results reported. The results prove that Z-score normalization Technique is superior for intrusion detection in terms of detection accuracy. Min-Max Normalization Technique is the winner among all other in terms of Execution time. It takes less time. In this paper, we also identified the efficient data normalization techniques that improvises the detection accuracy of a particular type of attack. For DOS and PROBE attacks, Z-score and sigmoid algorithms respectively give good accuracy. Whereas Decimal Scaling normalization algorithm performs better compare to other algorithms in case of detecting U2R and R2L attacks.

At the outset we can conclude that, more than one normalization Technique are suitable for IDS dataset to enhance its efficiency in detecting intrusion.

## REFERENCES

[1] J. Song, H. Takakura, Y. Okabe, and Y. Kwon, "A Robust Feature Normalization Scheme and Anomaly-Based IDS", In Proceedings of the 12th International Conference on Database Systems for Advanced Applications, 2007.

[2] Chen, Y. H., Abraham, A., & Yang, B, "Hybrid flexible neural-tree-based intrusion detection systems", International Journal of Intelligent Systems(IJIS), 22(4), pp. 337–352, 2007.

[3] Hansung Lee, Jiyoung Song, and Daihee Park, "Intrusion Detection System Based on Multi-class SVM", Dept. of computer & Information Science, Korea Univ., Korea, pp. 511–519, 2005.

[4] Han, J. and M. Kamber, 2001. Data Mining: Con Techniques, Morgan Kaufmann, USA

[5] http://www.mathworks.in/help/nnet/ug/choose-neural-network-Input-output-processing-functions.html

[6] Kevin L. Priddy, Paul E. Keller, "Artificial Neural Networks: An Introduction", First Edition, SPIE–, 2005.

[7] Vasudevan A R, S Selvakumar," Effect of Data normalization technique on Intrusion detraction dataset" NCIPS, 2012.

[8] Aickelin, U., Twycross, J., Hesketh-Roberts T, "Rule generalization in intrusion detection systems using SNORT", International Journal of Electronic Security and Digital Forensics, 1 (1), pp. 101–116, 2007.

[9] T. G. Dietterich, G. Bakiri."Solving multiclasslearning problems via error-correcting output codes",Journal of Artificial Intelligence Research (JAIR) vol 2, pp. 263-286, 1995.

[10 ] M. Tavallaee, E.Bagheri, W. Lu,A. Ghorbani. "A detailed analysis of the KDD CUP 99 data set", Proceedings IEEE international conference on Computational intelligence for security and defense applications, pp. 53-58, Ottawa, Ontario, Canada, 2009.

[11] A. M. Chandrashekhar, K Raghuveer, "Fortification of hybrid intrusion detection system using varients of neural networks and support vector machines" International journal of of network security and its applications (IJNSA), Vol 5, No 1, Jan 2013.

[12] R. Kohavi and F. Provost, "Glossary of terms," Editorial for Special Issue on Applications of Machine Learn-ing and the Knowledge Discovery Process. Machine Learning, pp. 271–274, 1998.

IJREAT
PRDG