International Journal of Computer Networks and Communications Security

VOL. 8, NO. 10, October 2020, 84–92
Available online at: www.ijcncs.org
E-ISSN 2308-9830 (Online) / ISSN 2410-0595 (Print)

Analysis of Intrusion Detection Using Machine Learning Techniques

PRIYA NERLIKAR[1], SAMIKSHA PANDEY[2], SWETA SHARMA[3], SUDHIR BAGADE[4]

[1, 2, 3, 4] Usha Mittal Institute of Technology, Department Of Computer Science and Technology, S.N.D.T Women's University Mumbai, 2020

[1]priya.nerlikar@gmail.com, [2]samiksha291098@gmail.com, [3]sweta15965@gmail.com,

[4]bsudhiran@ieee.org

ABSTRACT

Whenever an intrusion occurs into the computer system, the protection and computer assets are being compromised. Network-based attacks make it difficult for legitimate users to access various network services by purposely occupying or sabotaging network resources and services. The mechanisms such as, sending of large amounts of network traffic, exploiting well-known faults in networking services, and by overloading network hosts, the services are made unavailable. Hence, there is a desire for an Intrusion Detection System (IDS). Detection and analysis of such IDS is challenging. In this paper, we aim at detecting and analysing computer based attacks by examining various data records, for example KDDCup99 dataset. In this paper, the KDDCup99 dataset is employed for analyzing intrusion using machine learning techniques, like Support Vector Machine (SVM). For the efficiency, the dataset is reduced using data pre-processing methods i.e irrelevant or redundant features are removed. As a result, the accuracy and performance is evaluated using the Support Vector Machine Classification Algorithm. The accuracy obtained by our mechanism is 96.5%. The performance is evaluated by the error rate, so in our proposed system, we got an error rate as 3.38% which is observed to be efficient in terms of performance.

Keywords: *Accuracy, Intrusion Detection, Machine Learning, Network traffic, SVM*.

1 INTRODUCTION

An intrusion is defined as any set of actions that compromise the integrity, confidentiality or availability of resources [1]. These issues amplify the requirement of an efficient system that accurately detects the intrusion. An intrusion detection system (IDS) could be a course of action of monitoring network traffic and alerts network administrators against suspicious activity [2]. However, in the present scenario a variety of security tools are available to secure the system like firewalls. However, these firewalls don't seem to be always effective against the emerging intrusion attempts. Other than this issue the accessible IDS generates high false alarms and ends up in a higher error rate. Therefore, accurate intrusion detection systems are an essential component of the network to be secured [2].

A significant number of techniques have been developed which are based on machine learning approaches. So, for identifying the intrusion we have designed the support vector machine (SVM) learning algorithm. By using the SVM algorithm, we find out if there's any intrusion or not. The Dataset used in this paper is a Network Intrusion Dataset extracted from Kaggle Website which is KDDCup99 dataset [3].

In general attacks are classified into four major categories [4], which is described below.

a) Denial of Service Attack (DoS): This type of attack makes the network system and resource unavailable to its legitimate users. Attacks may completely crash the targeted system by software bug or send large amounts of useless traffic towards the target system to simply exhaust network, or computer resources.

b) Probe: This type of attacks typically scan the activities of the target host to gain systematic information or to find known vulnerabilities. Additionally, these attacks are more dangerous since they offer a map of machines or services that have known vulnerability in a network.

c) Remote To Local (R2L): By using this type of attack the attacker tries to gain access to the target machine without having any account on the machine.

d) User-to-Root (U2R): U2R describes attacks where local users of the system obtain unauthorized access to confidential information through root privileges. This attack typically exploits the weakness in the operating system or system programs running with root privileges, including those programs which are susceptible to buffer overflow attack.

In this paper, attack classification and mapping of the attack features is provided equivalent to each attack. The types of attacks and the features are classified as follows [4]:

1. Features for DoS: [logged in, count, 'serror rate','srvserror rate', 'same srv rate', 'dst host count','dst host srv count', 'dst host same srv rate', 'dst host serror rate', dst host srvserrorrate','service http', 'flag S0','flag SF'] [4].
2. Features for Probe:['logged in', 'rerror rate','srvrerror rate', 'dst host srv count','dst host diff srv rate', 'dst host same src port rate','dst host srv diff host rate', 'dst host rerror rate','dst host srvrerror rate', 'Protocol type icmp', 'service eco i', 'service private', 'flag SF'] [4].
3. Features for R2L:['src bytes', 'dstbytes','hot', 'num failed logins', 'is guest login','dst host srv count', 'dst host same src port rate','dst host srv diff host rate', 'service ftp','service ftp data', 'service http', 'service imap4','flag RSTO'] [4].
4. Features for U2R:['urgent', 'hot','root shell', 'num file creations', 'num shells','srv diff host rate', 'dst host count','dst host srv count', 'dst host same src port rate','dst host srv diff host rate', 'service ftp data','service http', 'service telnet] [4].

Intrusion detection is analysed using machine learning technique and accuracy is predicted. The rest of the paper is organized as follows.

Section 2 presents the literature survey. In section 3, we proposed our system. Implementation details are presented in section 4. The results and analysis is stated in section 5 and lastly the conclusion and future scope is in section 6.

Now, in the next section, we present the literature survey.

From this section, input the body of your manuscript according to the constitution that you had. For detailed information for authors, please refer to [1].

2 LITERATURE SURVEY

In this section, the critical review of various machine learning algorithms for intrusion detection is presented. We have studied various papers, where authors have discussed the methods, analysis and results of various machine learning algorithms for intrusion detection. These literatures are presented below.

Giriraj Vyas et. al [4] discussed categorization of IDS based on misuse detection vs anomaly detection. Misuse detection contained a signature database to check the attack signature as checked by any antivirus system. Unlike a misuse detection system, the anomaly detection system may detect novel attacks. However, false positives are a weakness of anomaly detection but the low false negatives are it's strength to extend the robustness, and accuracy of IDS system proposed ensemble classifier based approach. Proposed approach gave the most effective accuracy for every category of attack patterns. Ensemble method aims at improving the predictive performance of the given statistical learning and model fitting. A decision tree algorithm detects the anomaly attack. The experimental result shows that SVM gives 100% detection accuracy for normal and denial of service classes and comparable to warning rate, training, and testing times. The experimental results show that the proposed algorithm gives better and robust representation of information, because it was able to select features leading to 80.4 to statistical data reduction, select significant attributes from the chosen features and achieve detection accuracy about 96.7% with a warning rate of 3%.

Markus Ring et. al [5] provided a literature survey of existing network based intrusion detection data sets. At first, the underlying data are investigated in additional detail Network-based data appear in packet-based or flow based format. While flow-based data contain only *meta* information about network connections. The packet based data also contain payload. Further, the paper analyzes and groups different dataset properties which are often employed in literature to gauge the quality of network-based data sets. If none, delete this survey is an exhaustive literature overview of network-based data sets and analysis on which data set fulfils which data set properties. The paper focuses on attack scenarios within data sets and highlights relations between the information sets. Furthermore, we briefly subsume traffic generators and data repositories as further sources for network traffic besides typical data sets and supply some observations and recommendations. As a primary benefit, this survey establishes a group of information set properties as a basis for comparing available data sets and for identifying suitable data

sets, given specific evaluation scenarios. Further, they created a web site which references any or all mentioned data sets and data repositories.

Jayshree Jha et. al [6] proposed an intrusion detection system using a support vector machine. The dataset used is NSL KDD which efficiently characterizes normal traffic and distinguishes it from abnormal traffic using SVM. This research uses hybrid approach for feature selection i.e features has been ranked using independent measures(Information Gain Ratio).The K means classifiers predictive accuracy is employed to reach an optimal set S of the simplest features. After selecting relevant features from the above approach the dataset is reduced and can increase the performance and detection accuracy of SVM detection models. Moreover training testing will also be reduced. The Paper also presents a study that incorporates Information Gain Ratio and K-Means Algorithm. The NSL KDD dataset is ranked using IGR and also the feature subset selection done using K means classifiers predictive accuracy is employed to achieve an optimal set of the features which maximize detection accuracy of SVM classifier. The feature selection algorithm starts with an empty set S of the best features and so proceeds to feature features F into S sequentially .After each iteration the goodness of the resulting set of features S is measured by the accuracy of the k-means classifier. The choice process stops when the gained classifiers accuracy is below a specific selected threshold value or in some cases when the accuracy of the current subset is below the accuracy of the previous subset. The dataset used is NSL KDD and also the detection detection-model on SVM classifier.

Ripon Patgiri et. al. [7] proposed various machine learning techniques algorithms which applied to detect the intrusion like SVM, Naive Bayes, K-Nearest Neighbors (KNN) used for handling regression and classification tasks. The dataset used is NSL KDD which contains data like attack types which are the features. Feature selection is an important part of this process. The algorithm working would be supported selected features only for e.g: Features selected for DoS: [’duration’, ’wrong fragment’, ’hot’, ’num compromised’, ’num root’, ’numfile creations’, ’is guest login’, ’srv count’, ’dst host rerror rate’, ’Protocol type icmp’, ’Protocol type tcp’, ’Protocol type udp’, ’flag S0’] likewise we are going to select the features forest of the attacks i.e Probe, R2L, U2R. According to this the speed and accuracy of the technique are going to be predicted. The paper also uses a recursive feature elimination method which is beneficial for choosing sets of features used for SVM and Random Forest Algorithm. As a result they need stated that SVM performed better than Random Forest for eg. DoS attack accuracy for SVM is 86% whereas Random Forest performed 80% likewise all the attacks are compared and predicted that SVM performed better.

Mohammad Almseidin et. al [8] proposed machine learning algorithms like j48, Random Forest, Random tree, Decision table, MLP, Naive Bayes and Bayes network technique have been performed on KDD cup 99 dataset. it’s successfully performed classification to gauge the chosen classifier. In this paper, authors discussed that the 148753 instances of records have been extracted for training the model. During this training phase will have 60000 random instances of records, here several metrics computed accuracy rate, precision, false positive and false negative, true positive and true negative. This experiment demonstrated that there’s no machine learning algorithm which may define all styles of attack. The MLP algorithm recorded the highest MLP accuracy rate of 98.36%. They implemented a detection system supported extended classifier system and neural network to scale back false positive alarm the maximum amount possible. After implementation confusion matrices were generated for every machine learning classifier. The random forest achieved the highest accuracy of 93.77% with smallest RMSE and false positive rate. The Random tree classifier reached an all-time low average accuracy rate 90.73 with smallest ROC value. J48 classifier tested confidently factor of 0.25, num folds 3, seed =1, unprund false , collapse tree = tree.

Janu Gupta et. al [9] discussed machine learning algorithms like k-means , MLP , decision tree , Naive Bayes, SVM, Random forest are performed on KDD Cup 99 dataset. In this paper, authors focused on establishing relationships between the attack and protocol employed by attackers by using cluster data. Signature based IDS detect intrusion by looking for specific patterns, it can detect only known attacks and they’re unable to detect new attacks. PCA is employed for dimensionality reduction, they also removed duplicate samplesfrom the dataset. They have used Oracle 10g data metrics as a tool for the analysis of dataset and build 1000 clusters to segment the 494021 records here, training dataset contains 494021 samples while testing contains 311029 samples. As a result accuracy for all the classification techniques used is more than 90%. Among the classification techniques, the choice tree classifier gives the best accuracy of just about 95%. Further, the training and testing time for decision tree classifiers is quite good.

Riyazahmed A. Jamdar et. al [10] proposed a model for intrusion detection system using decision tree algorithm. This model detects anomaly based intrusion. Here, authors use the dataset change

control identifiers (CCIDS) 2017. They have used recursive feature elimination (RFE) for the selection of features. This paper can detect signature based attacks and profile based attacks. The accuracy of the classifier on test data is 99%. The true positive rate (TPR) is 99.9% and false positive rate (FPR) is 0.1 %. It doesn't include any data processing model. The drawback of this paper is that it cannot handle a great deal of knowledge about the attacks. The database CICIDS 2017 provides 84 features with 4 categorical columns. The 11 important criteria for developing a dataset are given. Some of them are complete network configuration, complete traffic, labelled dataset and complete interaction attack diversity.

Yasir Hamid et. al[11] in this paper showed a relative investigation of execution of various machine learning algorithms using *weka*. Paper uses KDDCUP99 using 10 fold cross validation. The paper mainly concentrates on the issue of IDS by applying different prediction procedures under supervised learning techniques. It focuses on different prediction techniques that includes Random Forest Zero R, One R, Naive Bayes, Multilayer perceptron, k star, and AdaBoost. Training data consists of 22 differing types of attack and 39 attacks are present within the test data. Features are classified into four groups: features consisting of all the attributes from TCP/IP, the features are time based, connect-based. The mechanism performed best on the dataset with 99.97% with mean absolute error of 0. In the paper, two sets of experiments are performed, one with only 11 element attributes the classification algorithm doesn't depend on all 41 features. The technique can regain results with reduced number of attributes.

Yuyang Zhou et. Al [12], for increasing the detection ability of intrusion, authors proposed an efficient machine learning based IDS using CFS-BA algorithm. In this paper, a novel machine learning based IDS is proposed to extend the accuracy and efficiency of detection classification. They have used correlation based-feature-selection-Bat-algorithm (CFSA-BA). CFSA used for determining a subset of the initial feature to eliminate irrelevant features. The KDDCup'99, NSL-KDD and CIC-IDS2017 datasets are used for the analysis. For increasing the classification performance they need combined decisions from multiple classifiers (C4.5, RF and forest PA). The classifier combined the algorithm C4.5, RF and Forest PA supported the AOP combination rule. The training and testing time are going to be reduced From 113.53 and 2.93 to 44.78 and 2.06 seconds on the CIC-IDS2017. The classification achieves the highest F-measure of 0.998 and lowest warning Rate (FAR) of 0.17% on KDDCup'99. The disadvantages is that this algorithm can't handle larger network traffic for detection of rare attacks.

Based on above surveyed techniques, the summary of literature survey is shown in Table 1. We list the advantages, disadvantages, detection accuracy and observations of the surveyed methods in Table 1. It is observed that detection accuracy in the literature [6], [11] is above 99%.

Table 1: Summary of literature survey

Title	Algorithm/Technique Used	Accuracy	Disadvantages	Advantages
"A Survey of Network-based Intrusion Detection Data Sets"[6]	K-Means Algorithm To Support Vector Machine	99.37 %	The proposed model uses other existing hybrid models for comparison also ignores feature dependencies.	Easy to implement ,training done in faster manner.
"Detecting Anomaly Based Network Intrusion Using Feature Extraction and Classification Techniques"[9]	MLP Decision tree	90%	In the proposed model performance of the classifier depends on type of dataset if the type of dataset is numeric it generates complex decision tree.	Minimizes the ambiguity of complicated decisions and assigns exact values .
"An Efficient Intrusion Detection System Based on Feature Selection and Ensemble Classifier"[10]	Recursive Feature Elimination Method	99% TPR=99.9% FPR=0.19%	It is observed that method used can not handle great deal of knowledge.	Compatible with plethora models also the method does not require knowledge of what feature represents.
"Machine Learning Techniques for Intrusion Detection: A Comparative Analysis" [11]	-Random Forest zero R one R -K-star adaboost -Naive Bayes -multilayer perceptron	-56.837% -78.537% -99.745%	It is observed that The mentioned methods regain result with reduced Number of attributes.	-Random forest can handle thousands of input variables without variable deletion. -Naïve bayes dosen't require much training data. -MLP yield required decision function directly via training.
"Network Intrusion Detection System Using Machine Learning"[12]	Correlation based-feature-selection-Bat-algorithm (CFS-BA) Algorithm.	F-measure-0.998% FAR-0.17%	It is observed that mentioned algorithm can't handle Larger network traffic for detection of Rare attacks.	Computationally cheap, fast running time, ability of good generalization.

In the next section, we discuss proposed system

3 PROPOSED SYSTEM

In the proposed system, the dataset KDDCUP'99 is extracted from Kaggle website [3]. The data usually does not contain clear or formatted data so we will be performing pre-processing steps on that data for increasing the efficiency and accuracy. The snippet of the dataset is shown in fig 3, of Section 5. The features in the dataset characterize normal traffic and distinguish it from abnormal traffic using SVM classifier. For training the model, various steps are shown in Figure 1, which includes:

A. Data Collection

B. Data pre-processing techniques

1. Encoding Categorical Data.
2. Removing Missing Data

3. Train,Test, Split

4. Feature Scaling

C.Classificatio

n(SVM)

D. Prediction

E. Performance analysis

Fig. 1. Flowchart of proposed system.

A. Data Collection

The dataset used is KDDCup99 dataset which is extracted from the Kaggle website [3].The dataset contains the information about the duration, flag, service, src bytes, dest bytes and class labels. It is possible that the dataset will have missing value, redundant feature irrelevant feature so we will be performing data pre-processing techniques. The dataset contains total 41 features.

B. Data pre-processing

It is a process of preparing the raw data suitable for machine learning models. The dataset extracted is usually not clear or formatted data so we will be performing pre-processing on the steps of that data for increasing the efficiency, accuracy of machine learning techniques.

1. Encoding Categorical Data

It is a pre-processing technique which uses a label encoder and one hot encoder. Label Encoder: As we can't have text in our data so we need to make this data into numerical for easy processing. We use label encoder class. In this proposed system we will import the label encoder class from *sklearn* library and transform the columns which are needed to be label encoded than replace the existing data with new encoded data. The main disadvantage of using only label encoders is if there are different numbers in the same columns, the possibility to misunderstand the data is quite high. So to overcome this problem we use one hot encoder. One hot encoder: It takes the column which has Categorical data, which has label encoded split the column into multiple column where the numbers in the column are replaced by 1's and 0's depending on the column having value after one hot encoder then we will fit and transform the data in the process some new column will be added with 1st 0's depending on the rows/ column.

2. Missing Data Removal

In this data pre-processing steps the null values such as missing values are removed using an imputer library, i.e. the dataset having redundant, irrelevant, missing features are removed. This is done by deleting the row or by calculating the mean by the strategy which is useful for features. In this proposed work we are using the strategy " most frequent" which will replace the missing value by using the most frequent value along each column , this can be used with strings or numeric data .

3. Split Train Test

In this data pre-processing technique, we divide the dataset into training set and testing set. One portion of the data is used to develop a predictive model, and the other to evaluate the model's performance. This is the most important step of pre-processing, as it will enhance the performance of our machine learning technique. In proposed work, we have initialized test size as 0.20 which is 20% of the dataset. As the test dataset is 20% so our training dataset becomes 80%. We used 4 variables for output which is stated below.

x – train (feature for training data), x – test (feature for testing data), y- train (dependent variable for training data) and y- test (Independent variable for testing data).

4. Feature Scaling

In the proposed system we are using standard scaler for feature scaling.

Standard scaler

It transforms data in such a manner that it has mean as 0, and standard deviation as 1. So basically it standardizes data. It is useful for data which has negative value. It arranges the data in normal distribution after standardization; it is more

efficient in classification tasks than regression. It is performed during data pre-processing to handle highly varying values or units. If feature scaling is not done then machine learning algorithms weigh greater value higher and consider smaller value as lower value regardless of what are the unit's values.
Formula for
standardization:

Z= (x- u) / s
Where,
u= mean, x= variable (raw score), s= standard deviation.

C. Classification:

a. Support vector machine

SVM is a flexible class of supervised algorithms for both classification and regression tasks but in our proposed work, we will be using SVM for classification tasks. It is defined by a separating hyperplane. It outputs an optimal hyperplane which categorizes new examples. In two dimensional spaces, the hyperplane is a line dividing a plane in two parts where in each class lay in either side, which has the most important distance from the nearest point, in high dimensions, which clearly separates training set into categories. The vectors which are nearest to the hyper-plane are Support vectors as shown in Fig 2. We have a dataset that has two tags (green and red), whereas *x1* and *x2* are the features. We want a classifier that can classify the pair*(x1, x2)* of coordinates in either green or red, as shown in Figure 2. The SVM algorithm helps to find the best line or decision boundary; this best boundary or region is called a hyperplane. The SVM algorithm finds the closest point of the lines from both the classes. These points are called support vectors. The distance between the vectors and the hyper plane is called the margin. The goal of SVM is to maximize this margin. The hyper plane with maximum margin is called the optimal hyperplane. The aim is to select a hyperplane having as much margin as possible between hyperplane and any vector within the training set, giving a greater chance of new data being classified correctly.

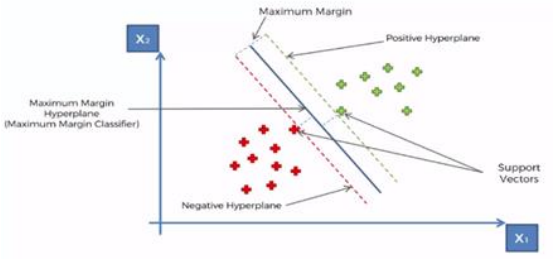

Fig 2. Support vector machine for classification

In the proposed work we have used SVM classifier which is a built-in-class of Scikit-learn's SVM library mainly used for classification tasks [13].The class takes one parameter which is the kernel-type. In our work we are using simple SVM so we set "linear" as a parameter since Simple SVM classifies only linearly separable data. The fit method used is of SVC class for training data which is passed as a parameter to fit method. To predict the class we used the predict method of the SVC class.

Advantages of Support Vector Machine

- SVM works well when data is not known.
- It also works well with semi-structured or uni-structured data like images, trees, and text.
- Kernel trick is the real strength of SVM.
- In our proposed system we have used linear kernel type/trick. SVM can scale relatively high dimensional data.
- Since the SVM models have generalization the risk of overfitting is less in SVM.

D. Prediction

It's a process of predicting the attacks in the network from the dataset. This project will effectively predict the data from a dataset by enhancing the performance of the overall prediction results. The prediction is represented by a confusion matrix, as shown in Table 3. A confusion matrix is a technique for summarizing the performance of the classification algorithm. The accuracy, precision, and recall are calculated using the information given in a confusion matrix. Fig 5, is the result of performance in terms of confusion matrix.

Table 3: Confusion Matrix Table

		Predicted	
		NO	YES
Actual	NO	$\sum TN$	$\sum FP$
	YES	$\sum FN$	$\sum TP$

Advantages of Proposed System

- It gives the accurate results as compared to existing systems.
- The method used for feature scaling is easy to implement and understand.
- Reduces the information Loss and the bias of the inference due to the multiple estimates.

In the next section, we give the implementation details.

4 IMPLEMENTATION DETAILS

The software used for implementation in our proposed work is SPYDER [14]. It is a package of Anaconda Navigator. It gives better visualization results as compared to other packages.

KDDcup'99 Dataset

The KDDCup'99 Dataset [3] is the most widespread dataset used for Intrusion detection and analysis. Extracted dataset is of Network Intrusion Detection system. It has around 25000 instances. In our proposed work we initially are using 10%-20% of the data. The dataset has features of 4 attack types. Every attack has some selected important features which will be useful for Intrusion detection. The 4 attack types are as follows:

1.Features for DoS: [logged in, count, 'serror rate','srv serror rate', 'same srv rate', 'dst host count','dst host srv count', 'dst host same srv rate', 'dst host serror rate', dst host srv serror rate','service http', 'flag S0','flag SF'].
2.Features for Probe:['logged in', 'rerror rate','srv rerror rate', 'dst host srv count','dst host diff srv rate', 'dst host same src port rate','dst host srv diff host rate', 'dst host rerror rate','dst host srv rerror rate', 'Protocol type icmp', 'service eco i', 'service private', 'flag SF']
3. Features for R2L:[src bytes', 'dst bytes','hot', 'num failed logins', 'is guest login','dst host srv count', 'dst host same src port rate','dst host srv diff host rate', 'service ftp','service ftp data', 'service http', 'service imap4','flag RSTO']
4.Features for U2L:['urgent', 'hot','root shell', 'num file creations', 'num shells','srv diff host rate', 'dst host count','dst host srv count', 'dst host same src port rate','dst host srv diff host rate', 'service ftp data',' service http', 'service telnet]
The dataset is being cleaned, encoded with categorical data, splited to train and test, feature scaled. After pre-processing, we create an instance of a linear SVM classifier from *scikit learns* [13]. Once, we trained the SVM classifier, we tested it for prediction. Next, we analysed the result and performance using metrics like True Positive Rate, True Negative Rate, False Positive Rate, False Negative rate, accuracy, precision, recall, and F1-Score [15]. Here, we briefly define these metrics.

True Positive Rate(TP) – The probability that an actual positive will test positive.
True Negative Rate(TN) – The probability that an actual negative will test negative.
False Positive Rate(FP) – The probability that a false alarm will be raised i.e a positive result will be given when true value is negative.
False Negative rate(FN) – The probability that a true positive will be missed by test.
Accuracy – It is the ratio of correctly predicted observation to the total observations.
TP+TN / TP+TN+FP+FN.
Precision – It is the ratio of correctly predicted positive observations to the total predicted positive observations.
TP / TP+FP.
Recall (Sensitivity) – It is the ratio of Correctly predicted positive observations to the all observations in actual class-yes.
TP / TP+FP.
***F1-Score* –** It is the weighted average of Precision and Recall. (2*Recall * Precision) / (Recall + Precision).
In the next section, we discuss the results and analysis.

5 RESULT AND ANALYSIS

In this section, the results and analysis of the proposed method is presented. The result will be generated based on the overall classification and prediction of attacks. The performance of this proposed approach is evaluated using metrics listed in the above section 4. Based on the literature survey we have compared the accuracies of existing and proposed systems as shown in Table 2 It is observed that the method we used for feature scaling i.e standard scalar method in proposed system is easy to implement and works effectively for the classification task using a support vector machine. The accuracy we got is 96.5% which is observed to be best for a simple support vector machine algorithm where the data is linearly separable

Table 2: Comparison of accuracies for existing and proposed system

Existing System Accuracy	**Proposed System Accuracy**
Ripon Patgiri et al in **[5]** titled "An Investigation on Intrusion Detection System Using Machine Learning" **DoS attack accuracy for SVM =86%.**	Classification using SVM **Method used for feature scailing = Standard Scaler** **Accuracy=96.5%** TP Rate=98.04% FP Rate=1.9522% FN Rate=4.823% TN rate=95.17% **Accuracy=96.61%** Error Rate=3.38% Precision=98.04% Recall=95.311% F1-Score=96.65%
Mohammad Almseidin et al in **[8]** titled "Evaluation of Machine Learning Algorithms for Intrusion Detection System" **Random forest accuracy achieved=93.77%(Smallest RMSE)** **Lowest Average Accuracy rate=90.73%(smallest ROC value)**	
Janu Gupta et al in **[9]** titled "Detecting Anomaly Based Network Intrusion Using Feature Extraction and Classification Techniques" **Decision Tree Accuracy achieved =90%**	
Riyazahmed A. Jamdar et. al in **[12]** titled "Network Intrusion Detection System Using Machine Learning" **Correlation based-feature-selection-Bat-algorithm (CFS-BA) Algorithm accuracy achieved =99.8%**	

The obtained results are shown in Fig 4, Fig 5, Fig 6 after training and testing the SVM algorithm. The obtained results are discussed below.

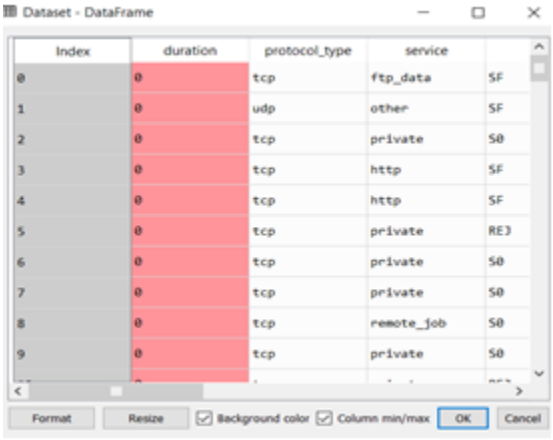

Fig. 3. KDD CUP'99 Dataset.

The Figure 3 demonstrates the dataset which is extracted from Kaggle website [3]. The dataset contains 41 features for DoS, Probe, U2R, R2L attacks etc.

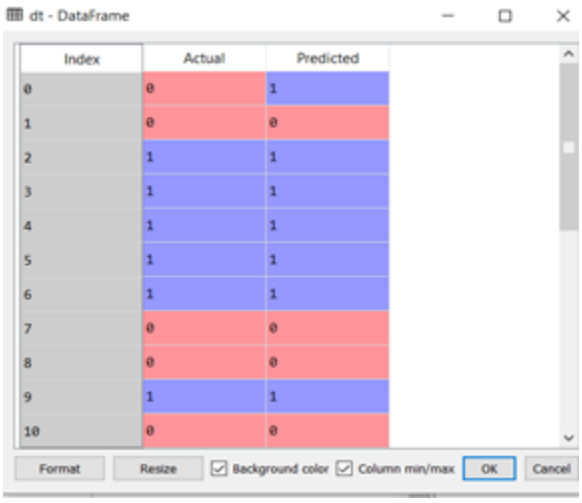

Fig. 4. Actual and predicted data frame (Comparison of the y_pred and y_test to check the difference between actual and predicted values)

After fitting the training set to the SVM classifier, we imported SVC class from Sklearn.svm library. We have used kernel='linear', as we are creating SVM for linearly separable data. The model performance can be altered by changing the value of the kernel. The result of y_pred and y_test is compared to check the difference between the actual value and predicted values as shown in Figure 4.

Fig. 5a. Results of confusion matrices

In Figure 5a, there are 452+513= 965 correct predictions and 26+9= 35 incorrect predictions. Therefore, we can say that our SVM models performance is better as compared to the existing work in the literature [5], [8].

```
In [1]: runfile('C:/Users/priya nerlikar/Documents/Source code/Intruders/major.py',
wdir='C:/Users/priya nerlikar/Documents/Source code/Intruders')
             duration  num_file_creations  ...         hot  wrong_fragment
count      119.000000               119.0  ...       119.0      119.000000
mean       357.294118                 0.0  ...         0.0        0.075630
std       2559.653096                 0.0  ...         0.0        0.472277
min          0.000000                 0.0  ...         0.0        0.000000
25%          0.000000                 0.0  ...         0.0        0.000000
50%          0.000000                 0.0  ...         0.0        0.000000
75%          0.000000                 0.0  ...         0.0        0.000000
max      25950.000000                 0.0  ...         0.0        3.000000

[8 rows x 7 columns]
```

Fig. 5b. Overall accuracy result

The overall accuracy for DoS, R2L, Probe, U2R that is predicted using SVM Algorithm, and using 41 features is shown in Figure 5b. The calculation of mean and min, max values is also mentioned for 41 features.

```
Result Generation
------------------

True positive =  98.0477223427332 %

False positive =  1.9522776572668112 %

False negative =  4.823747680890538 %

True negative =  95.17625231910947 %

Accuracy =  96.61198733092132 %

Error Rate =  3.3880126690786745 %

Precision =  98.0477223427332 %

Recall =  95.31089846409039 %

F1-Score =  96.65994171710379 %
```

Fig. 6. Calculated TP, FP, FN, TN, Accuracy, Error Rate, Precision, Recall, and F1-Score

In Figure 6, the accuracy of the proposed system obtained is 96.6% which is higher for the model to be considered as accurate. TPR, TNR play a major role in predicting accuracy while FPR, FNR play a

major role for calculating F1-score. The less the error rate, more precise is the classifier. Error rate is 3.38% which is observed to be good. However, the existing work in [11] shows that the error rate is 0 in some attack cases.
Below, we present the conclusion of our paper.

6 CONCLUSION AND FUTURE WORK

In this paper, we presented the system for intrusion detection which uses machine learning techniques namely SVM. SVM is good when data is not known. It also works well with semi-structured or uni-structured data like images, trees, text. Kernel trick is the real strength of SVM. In our proposed system, we have used linear kernel type/trick. SVM scales relatively high dimensional data where the number of features are generally greater than the number of observations i.e model complexity is of O(n-features * n^2 samples). Thus, it is perfect for SVM to work with high dimensional data. Since, the SVM models have generalization the risk of over fitting, which is less in SVM. We worked on this algorithm using all 41 features (Listed in Section 4). It is observed that some features are redundant and irrelevant so feature selection plays an important role in our proposed work. After performing the classification techniques on the dataset we got accuracy as 96% as a result. Due to the optimal margin gap between separating hyperplanes, SVM can do better with test data and the robustness or stability results in high accuracy which makes SVM computationally more efficient in the selected scenario.
In future, we are intending to apply Random Forest Algorithm [16] which is also used mainly for classification tasks. After applying the algorithm, the accuracies of both the algorithms i.e SVM and Random Forest will be compared and analysed in terms of performance and accuracy.

7 REFERENCES

[1] “Intrusion Classification” Available On : https://www.cerias.purdue.edu/about/history/coast_resources/idcontent/classification.html [Online]

[2] “Introduction to Intrusion Detection System” Available On:https://www.keyinfo.com/introduction-to-intrusion-detection-systems-ids/[Online]

[3] “KDDCUP’99 Dataset” Available on: URL(https://www.kaggle.com/sampadab17/network-intrusiondetectionTraindata.csv) [Online]

[4] G. Vyas, S.Meena and P. Kumar, ” Intrusion Detection Systems: A Modern Investigation” , International Journal of Engineering, Management Sciences ,ISSN: 2348 –3733, Volume-1, Issue-11, November 2014.

[5] R.Patgiri, U.Varshney, T.Akutota, and R. Kunde, ”An Investigation on Intrusion Detection System Using Machine Learning” IEEE international Conference on Symposium Series on Computational Intelligence SSCI 2018 Assam-788010.

[6] M. Ring, S. Wunderlich, D.Scheuring, D.Landes and A. Hotho, ”A Survey of Network-based Intrusion Detection Data Sets” arXiv:1903.02460v2 csCR 6 July 2019.

[7] J. Jha, L. Ragha , ”Intrusion Detection System using Support Vector Machine” International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868,FCS New York ,USA ,International Conference workshop on Advanced Computing 2013,Navi Mumbai.

[8] M. Almseidin, M. Alzubi, S. Kovacs and M. Alkasassbeh, ”Evaluation of Machine Learning Algorithms for Intrusion Detection System” , Department of Information Technology, University of Miskolc, H-3515 Miskolc, Hungary,Dec 2018 .

[9] J. Gupta , J. Singh , ”Detecting Anomaly Based Network Intrusion Using Feature Extraction and Classification Techniques” , International Journal of Advanced Research in Computer Science,Volume 8, No. 5, May – June 2017.

[10] Y. Zhou, G. Cheng, Senior Member, IEEE, S. Jiang, and M. Dai, ”An Efficient Intrusion Detection System Based on Feature Selection and Ensemble Classifier”, JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015.

[11] Y. Hamid, M. Sugumaran, L. Journaux, Senior Member, IEEE, S. Jiang, and M. Dai,”Machine Learning Techniques for Intrusion Detection: A Comparative Analysis” , INTERNATIONAL CONFERENCE ON INFORMATICS AND ANALYTICS (ICIA ’ 16), Aug 2016, Pondichery,´ India. pp.53, ff10.1145/2980258.2980378ff. ffhal-01392098f.

[12] R. Jamadar, ”Network Intrusion Detection System Using Machine Learning” , Indian Journal of Science and Technology, Vol 11(48), DOI: 10.17485/ijst/2018/v11i48/139802, December 2018.

[13] “Scikit Built-In-Libraries for python “Available On: https://www.datacamp.com/community/tutorials/svm-classification-scikit-learn-python[Online]

[14] “SPYDER software “ Available On https://www.spyder-ide.org/ [Online]

[15] "Accuracy,Precision,Recall,F1-score Interpretation Of Performance measures" Available On : https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/#:~:text=Accuracy%20-%20Accuracy%20is%20the%20most,observation%20to%20the%20total%20observations [Online]

[16] N.Fanaz,"Random Forest Modelling for Network Intrusion Detection System",Procedia Computer Science VOL 89,Pages 213-217,August 2016.