Comparative Analysis of Intrusion Detection Attack Based on Machine Learning Classifiers

Surafel Mehari and **Prof. Dr. Anuja Kumar Acharya**

School of Computer Engineering, KIIT University, Bhubaneswar, India

Abstract
In current day information transmitted from one place to another by using network communication technology. Due to such transmission of information, networking system required a high security environment. The main strategy to secure this environment is to correctly identify the packet and detect if the packet contain a malicious and any illegal activity happened in network environments. To accomplish this we use intrusion detection system (IDS). Intrusion detection is a security technology that design detects and automatically alert or notify to a responsible person. However, creating an efficient Intrusion Detection System face a number of challenges. These challenges are false detection and the data contain high number of features. Currently many researchers use machine learning techniques to overcome the limitation of intrusion detection and increase the efficiency of intrusion detection for correctly identify the packet either the packet is normal or malicious. Many machine-learning techniques use in intrusion detection. However, the question is which machine learning classifiers has been potentially to address intrusion detection issue in network security environment. Choosing the appropriate machine learning techniques required to improve the accuracy of intrusion detection system. In this work, three machine learning classifier are analyzed. Support vector Machine, Naïve Bayes Classifier and K-Nearest Neighbor classifiers. These algorithms tested using NSL KDD dataset by using the combination of Chi square and Extra Tree feature selection method and Python used to implement, analyze and evaluate the classifiers. Experimental result show that K-Nearest Neighbor classifiers outperform the method in categorizing the packet either is normal or malicious.

Keywords:
Classifiers, False Detection, Python, NSL KDD, Intrusion Detection, Machine-learning

1. Introduction

In modern world internet growth and user is rapidly increase with a high benefit for the development of e-government and e- commerce. However, there is a challenge about confidentiality, integrity and availability of internet and resource. To safe the network or system, the organization uses a security technology for preventing and protecting sensitive information from intruders. Protect network environment is maintained by Intrusion detection system (IDS). Intrusion detection system is hardware or software that observes the data traffic carefully identify the malicious activity and decide that it is normal of attack. At this, time many organization practice intrusion detection system to protect their system from intrusion. However, this technology is suffer a common problem, which is generating huge number of false alarm and contain high number of data features. Machine learning is important techniques that used to improve effectiveness of intrusion detection system for detecting as well as monitoring the network environment. This thesis proposed studying a comparative analysis of three machine learning classifier i.e. Support Vector Machine (SVM), Naïve Bayes (NB) and K-Nearest Neighbor (KNN) classifiers. To select the best classifier by studying a comparison analysis of this machine learning use NSL KDD standard dataset and using the combination of Chi square and Extra Tree feature selection method. For simulation of the NSL KDD dataset, we use python.

2. Problem Statement

The underlying research problem that needed this research is the existence of false detection rate and containing high number of data features in intrusion detection system. In Present intrusion detection system scenario false detection and high number of data features are a big problem, The reasons for these large amounts of alerts is not caused by a single fault but a combination of several factors for example technology problem, lack of knowledge for configuring intrusion detection correctly and no baseline designing security policy. With a huge volume of wrong detection, by this reason the actual threats undetected as the system administrator will start ignoring reported incidents, as the overflow of alerts gets too many. This compromises the whole intrusion detection system reducing its security value to almost none, system performance degraded, consuming high power and memory usage, high wastage of time and cost.

This research is applicable and promising to make comparison analysis of machine learning classifier and select best feature to enhance intrusion detection efficiency.

Manuscript received October 5, 2024
Manuscript revised October 20, 2024
https://doi.org/10.22937/IJCSNS.2024.24.10.14

3. Intrusion Detection System

Intrusion detection system is network security technologies originally built for detect any malicious activities that compromise the principle of security that is confidentiality, integrity and availability. These security technologies have the ability to block threat and detect and alarm or notify when any malicious activity happened to the security officer through email or phone.

The main object of intrusion detection system is to have a high defense that does not allow such kind of offensives. It detects unauthorized activity in a network environment, firstly introduced in 1980 by Anderson [1].

Intrusion detection system grouped into in different categorized based on location, response time, detection and architecture [2] [3]. These are network and host, active and passive, signature and anomaly, and central and distributed intrusion detection system respectively.

Network Intrusion Detection System

Network intrusion detection is method for detection of interference and that monitors the whole network activity by sniffing the incoming, outgoing packet and identify it is normal or attack. If the packet is different from normal behavior the intrusion, detection is give notification for security officer or system administrators.

Host Intrusion Detection System

Host intrusion detection configured for detection of log suspicious event and sent alert to a responsible person when any malicious activity happened. HIDS install on individual host and analyze outgoing packet from a particular device and it is better to detect a particular device compering to host base intrusion detection system.

Active Intrusion Detection System

Active intrusion detection system configured for the purpose of automatically blocks the intrusion when malicious activity happened in a system. This intrusion detection also known as intrusion prevention system.

Passive Intrusion Detection System

Passive intrusion detection system is systems that configured for monitoring and analyzes the networks, and notify alarm to a system administrator when any malicious activity happened.

Signature Based Intrusion Detection System

Signature base intrusion detection also known as misuse detection or knowledge-based intrusion detection system. It uses to analyze system activity observing for events similarity pre-defined signatures that describe a recognized attack.

Anomaly Based Intrusion Detection System

Anomaly intrusion detection system or behavior based detection method that underlining on identifying uncommon activities in a networks environment, it operates using statistical measure. The intrusion detection system looking the entering packet and analyze it, if the packet is differ from the normal behavior the system generate alarm.

4. Literature Survey

Manjula C. Belavagi et al. (2016) [4], proposed a classification machine-learning model for intrusion detection system. They compared Logical Regression, Gaussian Naïve Bayes, Support Vector Machine and Random Forest finally; conclude that by using experimental result and Random Forest outperforms a high accuracy, low false positive rate, and high true positive rate other classifies weather the data is Normal or Attack. To test the classifier the author use NSL KDD dataset.

Ahmad, Iftikhar, et al. (2018) [5], They proposed and compared a three machine learning classifiers to increase or enhance the performance or accuracy and reduce false alarm of intrusion detection system. Those classifiers are Support Vector Machine, Random Forest and Extreme Learning. To evaluate the classifier they use NSL KDD dataset. The authors determine that from the experiment result Extreme learning is a high accuracy to detect the attack especially its design to analyze huge amount of data comparing to Support Vector machine and Random Forest.

Mehmood et al. (2016) [6], They used four machine learning classifier Support Vector Machine Naïve Bayes, J.48 Decision Tree and Decision Table for intrusion detection system and also compared the accuracy. By experimental result, J.48 Decision Tree is efficient compared to other classifiers. The Author uses KDD CUP99 standard Dataset for experiment.

Sumouli Choudhury et al. (2015)[7], proposed several machine-learning classifiers model for network intrusion detection system and compare its performance. To compare the classifiers, the author use WEKA Tools and for evaluating the performance or accuracy to detect attack, NSL KDD dataset are used. They conclude that from the experiment Random Forest and Bayes Net are best classifiers comparing to other by using WEKA Tools.

5. Proposed Model

Proposed model it include dataset, pre- processing, feature extraction and machine learning classification. This model is working based on machine learning classifiers for identifying normal or attack. To implement we use python as a programming language to write a code and python as tool to analysis the classifiers, NSL KDD standard dataset and for classification we use three machine learning technique that is support vector machine, K nearest neighbors and Naïve Bayes. Finally, we compare outcomes and identify better technique that applied in intrusion detection system for identifying attack.

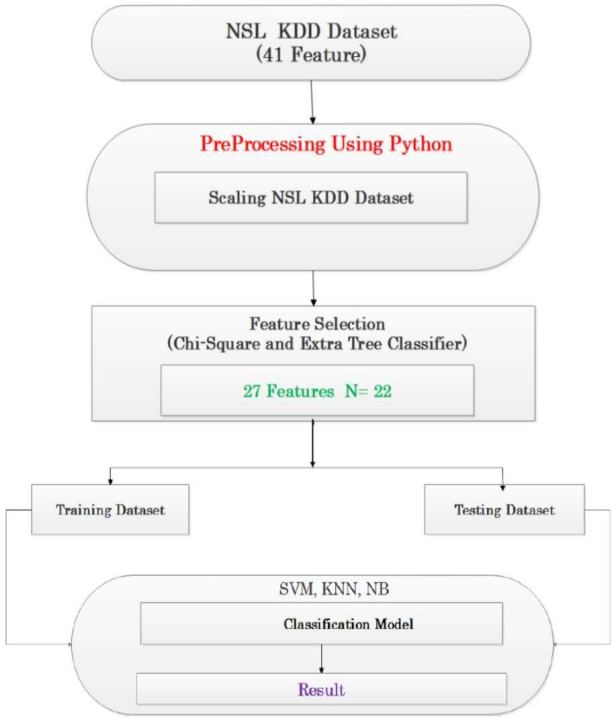

Figure 1.1 Proposed models for Machine Learning Classifiers using NSL KDD Standard Dataset

Machine Learning for Intrusion Detection System

Machine learning is used for improve detection rate, reducing false detection. We proposed for selecting the best machine learning classifier comparing by their effectiveness of identifying attack. We compare Support Vector Machine, K Nearest Neighbors and Naïve Bayes using NSL KDD standard dataset.

Support Vector Machine (SVM)

Support Vector Machine initially proposed by Vapnik in 1995 for solving problem of classification [8]. It used for both binary and multiclass classifiers. For our work, we use a binary classification method of support vector machine. The working norm of support vector machine is a linear separated the data, it use a class labeled data in training and mostly used for pattern recognition.

Naïve Bayes (NB)

Naïve Bayes is a supervised machine-learning algorithm that used to for classifying attack in intrusion detection system based on probability [9]. This classifier state that the probability of each attributes, which belongs to each class, considered for a prediction. The algorithm is assume that the probability of each attribute belonging to a given class value is not depend on all other attribute.

Prediction calculated by using Bayes' Theorem.

$$P(S/A) = P(A/S) * P(S)/P(A)$$

K Nearest Neighbors Classifier (KNN)

It is a supervised machine learning that used to classifying the given data based on similarity measure and this machine learning classifier are very important for numbering data. It is one of the simplest classification methods. It calculates the gap between distinct statistics points at the enter vectors and assigns the unlabeled facts factor to its nearest neighbor elegance using Euclidian distance measure.

NSL KDD Standard Dataset Description

NSL-KDD [10], is a data set suggested to solve some of the inherent problems of the KDD'99 data set. The dataset exist in different data file format. In this work, we use a labeled .CSV data file format. NSL-KDD standard dataset is a level dataset. A labeled dataset can easily learn the system. The original NSL-KDD dataset is divided into training and testing sets with 125973 and 25544 records, respectively. In addition, a training set contains 20% of the original training set called KDDTrain+ 20%. The types of attacks included in NSL-KDD are the same as the original KDD99 dataset. This Dataset contains 41 features. We use this 10% and 20% standard dataset for measuring the performance of our machine-learning model. It classifies the network packet into two categories that is normal and attack. The training and testing dataset are in CSV file format.

Table 1.1 NSL KDD Standard Dataset Features

Feature #	Feature Name	Feature #	Feature Name	Feature #	Feature Name
1	duration	15	su_attempted	29	srv_serror_rate
2	protocol_type	16	num_root	30	srv_serror_rate
3	service	17	num_file_creations	31	srv_diff_host_rate
4	src_bytes	18	num_shells	32	dst_host_count
5	dst_bytes	19	num_access_files	33	dst_host_srv_count
6	flag	20	num_outbound_cmds	34	dst_host_same_srv_rate
7	land	21	Is_hot_login	35	dst_host_diff_srv_rate
8	wrong_fragment	22	Is_guest_login	36	dst_host_same_src_port_rate
9	urgent	23	count	37	dst_host_srv_diff_host_rate
10	hot	24	serror_rate	38	dst_host_serror_rate
11	num_failed_logins	25	rerror_rate	39	dst_host_srv_serror_rate
12	logged_in	26	same_srv_rate	40	dst_host_rerror_rate
13	num_compromised	27	diff_srv_rate	41	dst_host_srv_rerror_rate
14	root_shell	28	srv_count		

Dataset Pre-Processing

In this preprocessing phase, the main task done is standardizing of the data with a range of [-1 and 1], Scaling is important to removing the noise of the data. To scaling dataset, we import **StandardScaler** python library from **sklearn.** Preprocessing is very important for data cleaning the data set, remove redundancies data and reduce large.

Feature Selection

In this work, we use two-feature selection method to select the best features that is Extra Tree classifier and Chi-Square feature selection and select 27 feature from 41 features by taking a combination of the two method when the K values 22 feature input. Feature selection is reducing data amount by selecting only useful feature we can avoid meaningless calculation on the useless feature.

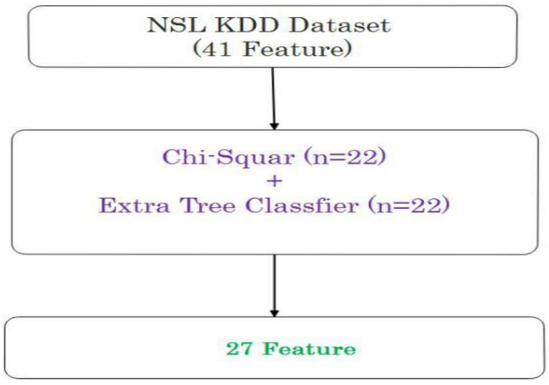

Figure 1.2 Proposed Model for Feature selection using Chi-Square and Extra Tree Classifier

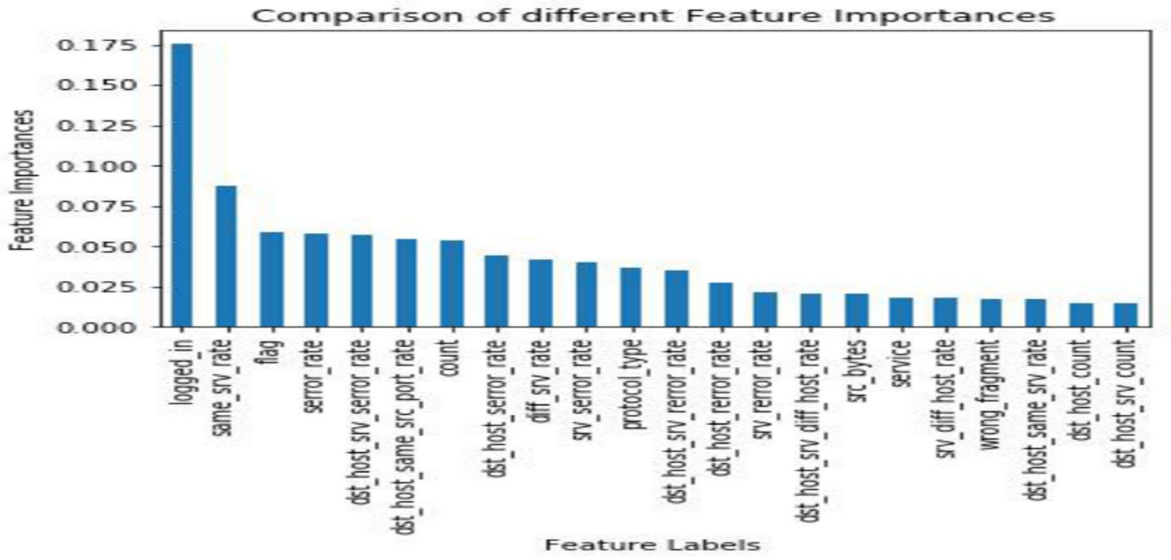

Figure 1.3 Chi-Square Feature Selection (n = 22) and Extra Tree Classifier method (n = 22) respectively

We use a hybrid of two-feature selection method to select the best features that is Extra Tree classifier and Chi-Square feature selection and select 27 features from 41 features by taking a combination of the two method when the n values 22 feature input.

Table 1.2 Selected 27 Feature (27 Features)

Feature #	Feature Name	Feature #	Feature Name
1	duration	28	srv_count
2	protocol_type	29	srv_serror_rate
3	service	30	srv_serror_rate
4	src_bytes	31	srv_diff_host_rate
5	dst_bytes	32	dst_host_count
6	flag	33	dst_host_srv_count
8	wrong_fragment	34	dst_host_same_srv_rate
12	logged_in	36	dst_host_same_src_port_rate
16	num_root	37	dst_host_srv_diff_host_rate
23	count	38	dst_host_serror_rate
24	serror_rate	39	dst_host_srv_serror_rate
25	rerror_rate	40	dst_host_rerror_rate
26	same_srv_rate	41	dst_host_srv_rerror_rate
27	diff_srv_rate		

6. Experiment Results and Analysis

We are considering three machine learning classifier and two standards data set these are support vector machine, Naïve Bayes and K Nearest Neighbors, NSL KDD standard dataset. These algorithms tested on Intel(R) Core™ i5 – 6200U CPU @ 2.4 GHZ, 8GB RAM and coding & analysis are done by Python3.7.

Standard Metrics to Evaluate Machine learning Model

True Positive, True Negative, False Positive and False Negative are a standard metric to evaluate machine learning classier their accuracy, recall, precision and f score.

Table 1.3 Standard matrixes for evaluation Machine Model

	Prediction		
Actual		**Normal**	**Attack**
	Normal	TN	FP
	Attack	FN	TP

The representation of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) defined as follows:

True Positive (TP) - The number of malicious records that correctly identified.

True Negative (TN) - The number of legitimate record that correctly classified.

False Positive (FP) - The number of records that are incorrectly identified as attacks however in fact they are legitimate activities.

False Negative (FN) - The number of records that are incorrectly classified as legitimate activities however in fact they are malicious.

Performance Measure

Accuracy - The number of correct predictions when expressed in percentage terms indicates the accuracy. It can be calculated from the confusion matrix by the the following formula.

Accuracy = (TP+TN)/ (TP+TN+FP+FN)

False Positive rate - It indicates the possibility of an algorithm to predict instance an attack which are actually normal.

False positive Rate (FPR) = FP/ (TN+FP)

Precision - It estimate the probability of positive prediction being correct.

Precision = TP/(TP+FP)

Recall - Recall (also called True Positive Rate) (TPR)) or Sensitivity, Recall measures the number of correctly classified examples relative to the total number of positive examples. In other words, the number of class members classified correctly over the total number of class member.

Recall = TP/(TP+FN)

F1 Score - It defined as the harmonic mean of sensitivity (recall) and precision.

F1 Score = 2*TP/ (2TP+FP+FN)

True Negative Rate - True negative rate also called Specificity. It measures the actual negatives, which are identified correctly.

True Negative Rate = TN/(FP+TN)

Experimental result I

NSL KDD dataset used for testing the performance of the machine-learning model that is KNN, NB and SVM. For this experiment, we use 10% and 20% data from the original NSL KDD dataset from this 80% for training and 20% for testing set. The following diagram shows the accuracy of 80 by 20 Training and Testing in KNN Model. For all experiment, we use 80 by 20 Training-Testing Dataset.

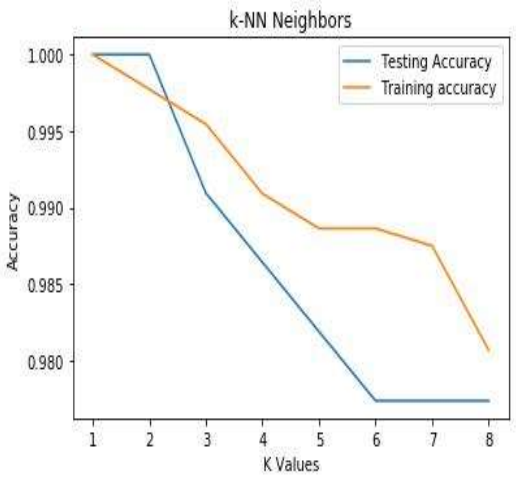

Figure 1.4 Accuracy of 80% by 20% Training Testing dataset for 27 Feature

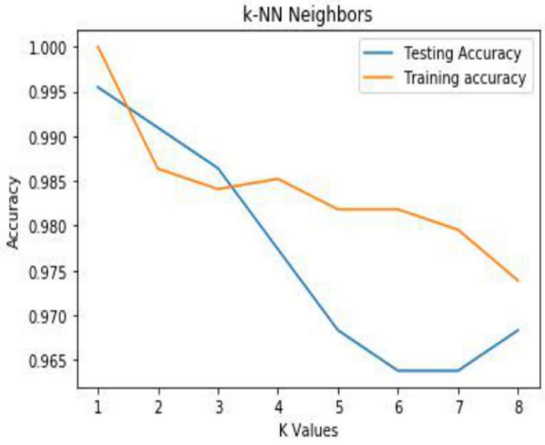

Figure 1.5 Accuracy of 80% by 20% Training Testing dataset for 41 Feature

Experimental result II

The second experiment was performed with a KNN Classifier using 20% and 10% NSL KDD dataset with 41 features and 27 features.

Table 1.4 Experiment result of KNN Classifier

Method	NSL KDD Dataset	Feature	Precision	Recall	F1-Score	Accuracy	False Positive Rate
KNN	10 % NSL KDD	41 Feature	94	95	98	97	0.8
		27 Feature	100	98	97	99	0.0
	20% NSL KDD	41 Feature	93	94	96	95	0.5
		27 Feature	96	98	98	98	0.9

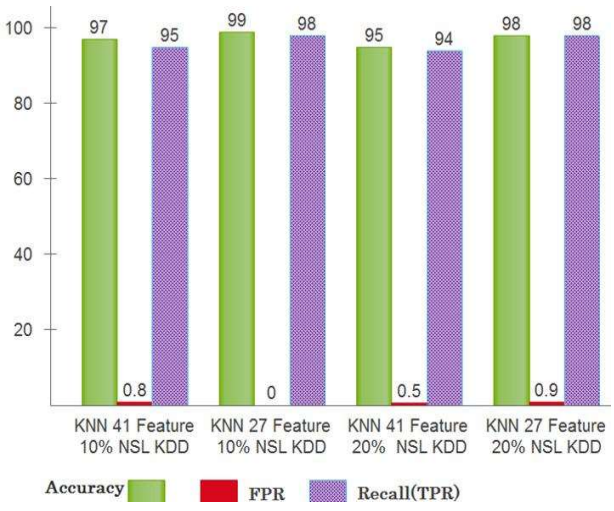

Figure 1.6 KNN Experiment Result

Figure 1.6 and table 1.4 shows that 20% and 10% NSL KDD dataset with 27 and 41 feature accuracy, false positive rate and true positive rate of KNN model for binary classifier either normal or attack. From the figure conclude that 10 percent of NSL KDD data with 27 feature KNN model has high score that is 99 percent accuracy, 98 percent recall(True positive Rate).

Experimental result III

The Third experiment was performed with a NB Classifier using 20% and 10% NSL KDD dataset with 41 features and 27 features.

Table 1.5 Experiment result of NB Classifier

Method	NSL KDD Dataset	Feature	Precision	Recall	F1-Score	Accuracy	False Positive Rate
NB	10 % NSL KDD	41 Feature	90	70	80	82	5
		27 Feature	94	95	95	93	0
	20% NSL KDD	41 Feature	82	86	84	85	10
		27 Feature	94	92	93	92	0

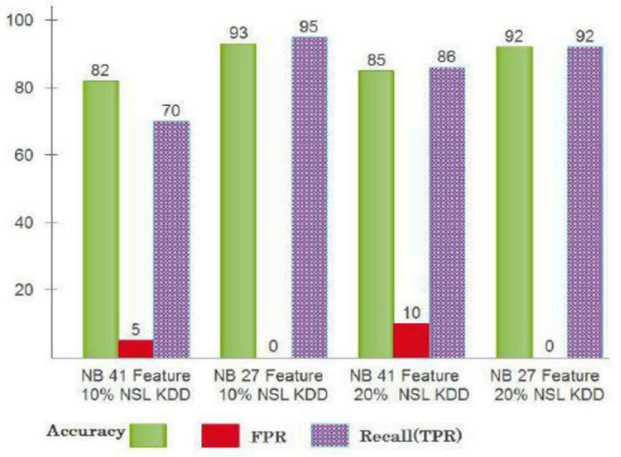

Figure 1.7 NB Experiment Result

Table 1.5 and figure 1.7 shows that 20 % and 10% NSL KDD dataset with 27 and 41 feature accuracy, false positive rate and true positive rate of NB model for binary classifier either normal or attack. From the above table and

figure we conclude that 10% of NSL KDD data with 27 feature NB model has high score that is 93 percent accuracy, 95 percent recall(True positive Rate).

Experimental result IV

The forth experiment was performed with a SVM Classifier using 20% and 10% NSL KDD dataset with 41 features and 27 feature.

Table 1.6 Experiment result of SVM Classifier

Method	NSL KDD Dataset	Feature	Precision	Recall	F1-Score	Accuracy	False Positive Rate
SVM	10 % NSL KDD	41 Feature	90	93	92	91	3
		27 Feature	93	93	95	94	2
	20% NSL KDD	41 Feature	92	93	93	92	4
		27 Feature	93	94	95	95	1

Table 1.6 and figure 1.8 shows that 20% and 10% NSL KDD dataset with 27 and 41 feature accuracy, false positive rate and true positive rate of SVM model for binary classifier either normal or attack. From this table and figure we conclude that 10% of NSL KDD data with 27 feature SVM model has high score that is 95 percent accuracy, 94 percent recall(True positive Rate).

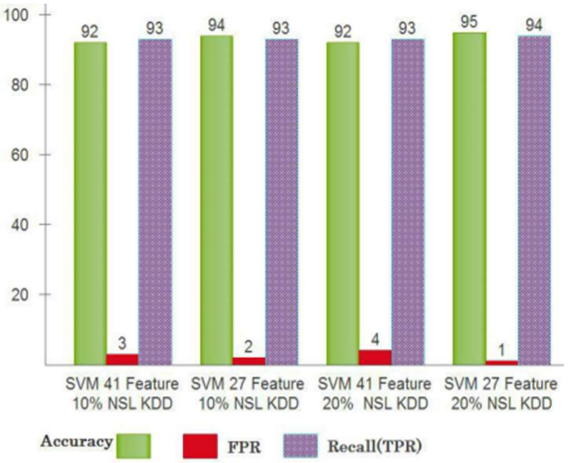

Figure 1.8 SVM Experiment Result

Experimental result V

The fifth experiment was performed with a comparison of three Classifier that is KNN, NB and SVM using 20% and 10% NSL KDD dataset with 41 features and 27 features.

Table 1.7 Comparison of Three Classifies

Method	NSL Dataset	Feature	Precision	Recall	F1-Score	Accuracy	False Positive Rate
KNN	10 % NSL KDD	41 Feature	94	95	98	97	0.8
		27 Feature	100	98	97	99	0.0
	20% NSL KDD	41 Feature	93	94	96	95	0.5
		27 Feature	96	98	98	98	0.9
NB	10 % NSL KDD	41 Feature	90	70	80	82	5
		27 Feature	94	95	95	93	0
	20% NSL KDD	41 Feature	82	86	84	85	10
		27 Feature	94	92	93	92	0

SVM	**10 % NSL KDD**	**41 Feature**	90	93	92	91	3
		27 Feature	93	93	95	94	2
	20% NSL KDD	**41 Feature**	92	93	93	92	4
		27 Feature	93	94	95	95	1

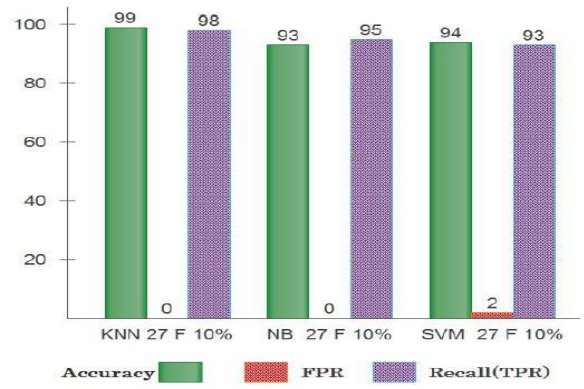

Figure 1.9 Comparison of Classifier 10% and 20% NSL KDD Dataset with 27 and 41 Features

From Figure 1.9 and table 1.7 shows that 20% and 10% NSL KDD dataset with 27 and 41 feature accuracy, false positive rate and true positive rate of three machine learning model for binary classifier either normal or attack. The experiment result show that KNN model perform best accuracy compered to SVM and NB model. The result show 99 percent accuracy, 98 percent recall (True positive Rate) with low false positive rate in 10 percent of NSL KDD dataset with 27 feature.

7. Conclusion

In this research, we study the comparative study of supervised machine learning algorithms classifiers KNN, NB and SVM for identifying whether the data is normal or attack for binary classification. These algorithms tested using NSL KDD standard dataset. Effective classifiers identified by comparing the performance on the Accuracy, False Positive rate and True Positive Rate (Recall). We conclude that from the experiment KNN Classifier outperforms other classifiers using 27 feature of NSL KDD dataset for both 10% and 20% NSL KDD standard Dataset. It has the accuracy of 99 percent.

8. Future work

Currently, Intrusion Detection security technology is important for any organization to identify the packet whether it is normal or attacks. Our future work, we use deep learning for increasing the performance of intrusion detection system.

References

[1] Anderson, J. P. Computer security threat monitoring and surveillance, 1980.

[2] J. J. Davis and A. J. Clark, "Data preprocessing for anomaly based network intrusion detection: A review," Computers & Security, vol. 30, pp. 353-375, 2011.

[3] “Host- vs. Network-Based Intrusion Detection Systems," Global Information Assurance Certification Paper. 2005

[4] Belavagi, Manjula C., and Balachandra Muniyal. "Performance evaluation of supervised machine learning algorithms for intrusion detection." Procedia Computer Science 89 (2016): 117-123.

[5] Ahmad, Iftikhar, et al. "Performance comparison of **support vector machine, random forest, and extreme** learning machine for intrusion detection." IEEE Access 6 (2018): 33789-33795.

[6] Mehmood, Tahir, and Helmi B. Md Rais. "Machine learning algorithms in context of intrusion detection." 2016 3rd International Conference on Computer and Information Sciences (ICCOINS). IEEE, 2016.

[7] Choudhury, Sumouli, and Anirban Bhowal. "Comparative analysis of machine learning algorithms along with classifiers for network intrusion detection." 2015 International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM). IEEE, 2015.

[8] Jha, Jayshree, and Leena Ragha. "Intrusion detection system using support vector machine." International Journal of Applied Information Systems (IJAIS) 3 (2013): 25-30.

[9] Chitrakar, Roshan, and Chuanhe Huang. "Anomaly based intrusion detection using hybrid learning approach of combining k-medoids clustering and naive Bayes classification." 2012 8th International Conference on Wireless Communications, Networking and Mobile Computing. IEEE, 2012.

[10] http://nsl.cs.unb.ca/NSL-KDD/, November 2014.