Development of an Abridging Algorithm for Intrusion Detection System

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering
in
Computer Science and Engineering

Submitted By
Sheetal Garg
(801732047)

Under the supervision of:
Dr. Raman Singh
Assistant Professor
Dr. V.K. Bhalla
Assistant Professor

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY
PATIALA – 147004

July 2019

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, **"Development of an Abridging Algorithm for Intrusion Detection System"**, in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Computer Science and Engineering* submitted in Computer Science and Engineering Department of Thapar Institute of Engineering and Technology, Patiala, is an authentic record of my own work carried out under the supervision of **Dr. Raman Singh** and **Dr. V.K Bhalla** refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

Signature: Garg

Sheetal Garg
801732047

Place: Patiala
Date: 02/Aug/19

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Approved By:

Signature Raman Sh

Dr. Raman Singh
Assistant Professor
Computer Science and Engineering

Department
Thapar Institute of Engineering & Technology
Patiala (Punjab), India

Signature Vinod K Bhalla

Dr. V.K. Bhalla
Assistant Professor
Computer Science and Engineering
Department
Thapar Institute of Engineering & Technology
Patiala (Punjab), India

ACKNOWLEDGEMENT

No volume of words is enough to express my gratitude towards my guide, Dr. Raman Singh and co-guide, Dr.V.K. Bhalla. I thank them for their time, patience, discussions and valuable time comments. They have been concerned and have aided for all material essential for the preparation of this thesis report. Their enthusiasm and optimism made this experience both rewarding and enjoyable.

I am equally grateful to Dr. Maninder Singh, Head of Computer Science and Engineering Department and Mr. Ashutosh Mishra, P.G. Coordinator, for motivation and inspiration that triggered me for the thesis work.

I also want to express my gratitude to Dr. S.S. Bhatia, Dean of Academic Affairs, for making provisions of infrastructure such as library facilities, computer labs equipped with net facilities, immensely useful for the learners to equip themselves with the latest in the field.

I am also thankful to the entire faculty and staff members of Computer Science and Engineering Department for their help, cooperation and affection, which made my stay at Thapar Institute of Engineering & Technology memorable.

I also thank my friends for assisting me as per their abilities, in whatever manner possible throughout the process of research.

Last but not the least, I would like to express my very profound gratitude to my parents and Almighty for showing me the right direction. This accomplishment would not have been possible without them.

ABSTRACT

An intrusion detection system (IDS) is a system that monitors network traffic for suspicious activity and issues alerts when such activity is discovered. While anomaly detection and reporting is the primary function, some intrusion detection systems are capable of taking actions when malicious activity or anomalous traffic is detected, including blocking traffic sent from suspicious IP addresses. There are so many challenges occur in IDS like Network Traffic Dataset is Imbalanced i.e. there are few anomalous connections as compared to normal connections. Network Traffic Dataset is huge, High False Alarm rate and High Intrusion Detection Time.

This thesis focuses on various issues like huge network traffic dataset i.e. large number of instances, large feature set, low accuracy and high rate of false alarms. With the help of feature selection technique relevant features are extracting. Propose technique helps to reduce the number of instances of dataset. The aim of this thesis is to enhance training time as well as memory requirement for processing and storage.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

IDS	Intrusion Detection System
NIDS	Network Based Intrusion Detection System
HIDS	Host Based Intrusion Detection System
FST	Feature Selection Technique
Inf- FS	Infinite Feature Selection
SVM	Support Vector Machine
NSL-KDD	Network Security Laboratory-Knowledge Discovery and Data Mining
TOPSIS	Technique for Order of Preference by Similarity to Ideal Solution
DBSCAN	Density - based spatial clustering of applications and noise
KDD	Knowledge Discovery in Database
NSL-KDD	Network Security Laboratory-Knowledge Discovery and Data Mining

CHAPTER 1
INTRODUCTION

Internet is used to communicate with each other, sharing resources, online work, banking, online shopping, finding information, accessing sort of online work etc. Internet connecting billions of devices through global network like smart phones, desktops, laptops, tablets, smart watches, smart TVs etc. One side internet makes our life easy and comfortable but on the other side it also increases so many risk factors like illegal activities or abuse. Security is the major concern in the world of internet. Traditionally mainly encryption, firewall and other methods were used to secure the data, but now a day's intrusion detection system plays a major role in the field of security or to detect the attack type. Intrusions can affect host as well as network based systems. Intruders are carried out to destroy sensitive files, access personal files, steal or gain unauthorized information etc. Computer network and internet requires security to protect the data from intruders. Security systems like firewall, antivirus, intrusion detection systems are must to protect our system form hackers and attackers. IDS (Intrusion Detection System) used for many applications or environments. Ad-hoc network, wireless devices, wired devices, host based and network based environment securing their data with the help of IDS. Many techniques and algorithms are available which are used in building of IDSs. Machine learning is commonly used technique which provides effective results for IDSs. Prior knowledge is required to learn the system so; datasets are used for the learning process in machine learning.

IDS: Intrusion Detection System

IDS is a software in the form of application or a device which helps in detecting malicious activities policy violation by comparing the behavior of user to the user profile and if any malicious activity/ unauthorized event recognized then a control function is presented in a computer which automatically control the event and take action as a response to the event. The user profiles are dynamically constructed and updated and they are created in all computers when the user is first time log into computer.

The basic components of IDS are represented with the help of figure 1.1.

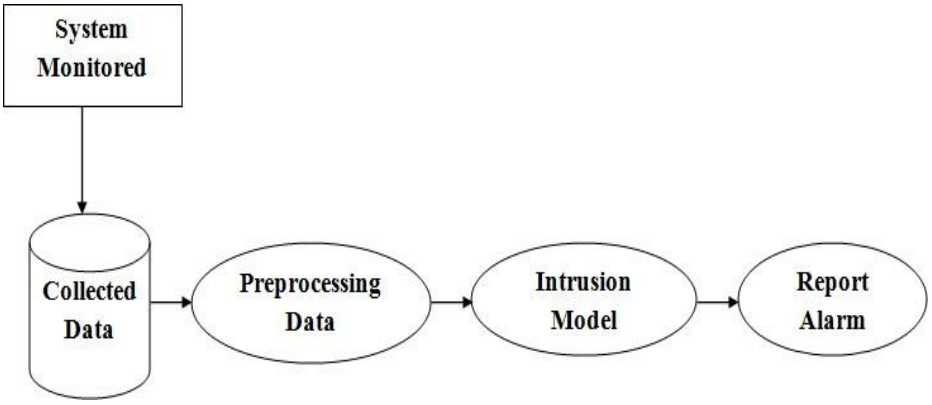

Figure 1.1: Basic component of IDS

These components discussed below:

- **System Monitored:-** The system which is going to be monitor, can be host based system or network based system, all that depending on the requirement of monitor. (NIDS) Network based intrusion detection system placed in place of networks, it process and analyze all the traffic passed through network, if any abnormal activity recognized then report to the administrator. Host based Intrusion detection system (HIDS) placed into particular devices or host, then monitor outbound and inbound traffic for that host.
- **Collected Data:-** Monitored system help to capture network traffic dataset. The data is captured from particular source it can be host or network based system. Dataset is used for training normal and abnormal patterns. Many tools are available for collecting or sniffing the traffic dataset. Examples: Wireshark, Tcpdump, NetworkMiner.
- **Preprocessing Data :-** Raw data is captured. It may contain redundant packets and missing values. Data preprocessing is a process to remove duplicates, redundant instances and clean missing values. Dataset contain huge number of features and instances. Extract the relevant features from the dataset is done by using (FST) feature selection techniques. In Data processing many languages are used like Matlab, Java, R language.
- **Intrusion Model:-** Modeling step is the focused step in the whole Intrusion detection system process. IDS learn normal and abnormal behavior using

captured dataset. For the Intrusion model the dataset divided into two parts i.e. training and testing. This step is completed in two steps: Train the model and test the model. Training part of the dataset is used to learn patterns from the dataset and testing part of dataset is used to find that model is working properly or not. Modern Intrusion detection systems use soft computing and machine learning to train them.

- **Report Alarm:-** If any attack is recognized then it report to the administrator and immediate action is needed to mitigate effect of attacks. Further processing of dataset is required after alarm raised to find out exact type of attack. Accuracy, False alarm rate of these alarm still a research big challenge.

Types of IDS

IDS classified as Signature / Misuse, Anomaly, Hybrid [10]. Theses IDSs described below:

Misuse/ Signature Based IDS:- This Intrusion detection system (IDS) is also called as misuse based IDS. It is used to detect abnormal activities or attacks in a host or network based systems. Signature based IDSs recognize intrusions or attacks by comparison of observed data with predefined signature or pattern or descriptions of intrusion behavior. Therefore, a signature database is specified priori corresponding to attacks. In Signature Based IDS learn only abnormal behavior space and rest is considered normal. Figure 1.2 shows Signature based IDS.

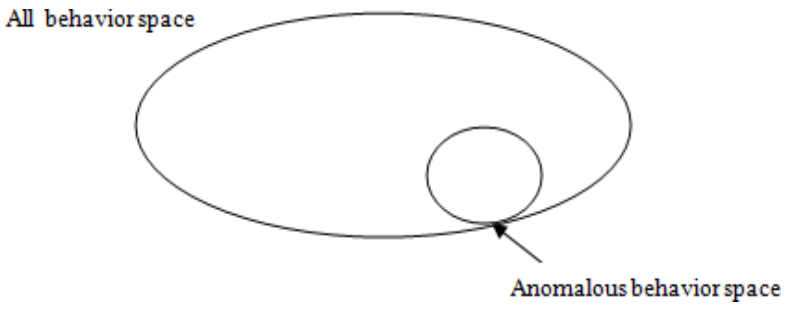

Figure 1.2: Signature based IDS

Misuse IDS does not recognize new pattern or attack due to no predefined signature in database. Many techniques are developed by researchers based on signatures. Some of the techniques are (LESG) Network-based length-based signature generator. It is fast, noise tolerant and has efficient signature matching. (SDC) Signature Detection

classification has highly demand to secure network for IDS. For SDC Genetic algorithm and Hidden Markov Chain are the best feature selection techniques. (NSG) Network – based Signature Generation (NSG) developed to quickly and automatically generate accurate pattern for worms, mainly polymorphic worms.

Anomaly Based IDS:-It was observed that abnormal behaviors/ connections are rare in dataset compared to the normal behavior and thus learn normal behavior rather than what is abnormal or anomalous. The issue have to face in anomaly IDS is that if new user occur in the system it shows that it is an abnormal activity. This type of IDS raises an alarm when it captures different activity from learned normal activities. Figure 1.3 shows Anomaly based IDS.

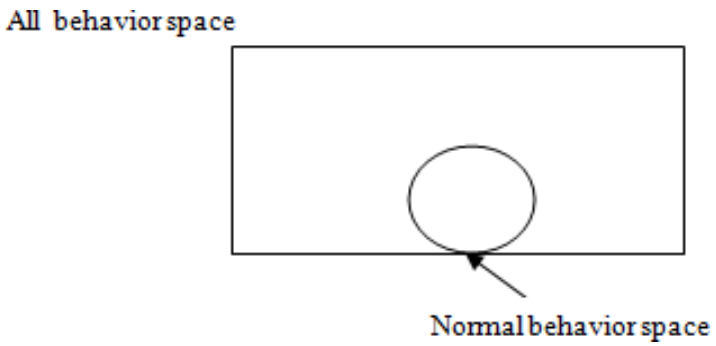

Figure 1.3: Anomaly based IDS

Hybrid IDS:- Misuse based IDS detects only system known attacks, if any new attack occurs then signature based IDS cannot capture that attack. Anomaly based IDS has higher (FPR) false positive rate. Hybrid IDS is a combination of both signature/ misuse based IDS and Anomaly based IDS.

Research Area

The field of IDS is mainly focused on (HIDS) Host based systems, (NIDS) network based systems, wired devices and wireless devices. It is important to secure the system from the malicious activities with less false positive rate (FPR) with in less amount of time and should have accepted accuracy. To achieve more accuracy with in less time, researchers try to apply different algorithms and methods on the network traffic datasets. Many datasets are available for the checking purpose of system

performance. Performance of system evaluated by use of parameters like accuracy, recall, precision, false alarm rate and time by using different datasets.

Major Contribution

- Large feature dataset and large number of instances dataset like network traffic dataset issues are analyzed.
- An intrusion detection based technique proposed for Dimensionality reduction of network traffic dataset.
- The proposed technique deals with IDS problems like large feature set, large instance of dataset, large training time and low accuracy.
- Infinite feature selection technique (Inf-FS) is extracting the relevant features from dataset and ranking all the features according to the relevant of features.
- Horizontal abridging algorithm is proposed to reduce the dataset instances even after preserve its characteristics so that it takes less amount of time while train the model or required less storage.
- Proposed technique is uses Kyoto University Benchmark 2009 and NSL-KDD 2009 dataset to train and test the classifier.
- Kyoto University benchmark 2009 dataset achieves accuracy of 87.55% and false positive rate (FPR) of 0.036 with detection time of 1 minute 21 seconds.
- Binary class NSL-KDD 2009 dataset achieve accuracy of 97.11% and (FPR) false positive rate 0.05 with lowest time of 3 min 45 seconds is achieved.

Thesis Structure

This thesis proposes intrusion detection algorithm which consider various problems like large number of featured dataset, large number of dataset instances, large storage requirement and training time.

- **Chapter 1** starts with the Intrusion detection system (IDS) introduction and its type with research area.
- **Chapter 2** explains the literature survey related to the work in past few years in the field of Intrusion detection system (IDS).
- **Chapter 3** explains the gap analyzed and issues in the studied survey and describes the problem of this thesis.

- **Chapter 4** explains proposed technique for intrusion detection. Discuss NSL-KDD 2009 and Kyoto University benchmark 2009 network traffic dataset. This chapter also discusses three experiments carried out in this thesis. Performance criteria will also explain.
- **Chapter 5** shows the result yield by purposed technique with Kyoto University benchmark 2009 dataset and NSL-KDD 2009.
- **Chapter 6** is final chapter which concludes the thesis work and future work.

CHAPTER 2
LITERATURE REVIEW

Susan M.Bridges et.al.[1] developed an architecture in which integrates intrusion detection methods with machine learning methods. Genetic Algorithm was improved the performance of system by extracting relevant features from dataset and tuning the membership function parameters.

Leonid Portnoy et.al.[2] presented a framework for intrusion detection with unlabelled data using clustering. Discussed method can be used as a part of larger system, which collect automatically data from the network and evaluate several intrusion detection methods for the data. KDD CUP 99 data used for training and testing purpose. The average (DR) detection rate was 40 -55 % with 1.3 – 2.3 % false positive rate.

Adhitya Chittur et.al.[3] presented an approach to detect intruders which used complex (AI) artificial intelligence method known as (GA) Genetic Algorithm. This algorithm was tested real-world simulation to gauge its effectiveness under unpredictable conditions. It was concluded that the use Genetic algorithm successfully get correct DR and low (FPR) false positive rate.

B. Balajinath et.al.[4] developed Intrusion detection called GBID i.e. Genetic Algorithm Based Intrusion Detector. This algorithm is based on "Learning the individual behavior". Detection of the intrusions from the system by using this approach was with accuracy 96.8% and FPR 3.2%.

Khaled Labib et.al.[5] developed a network based IDS using Self Organizing Map (SOM). The results showed that this technique can distinguish DOS attack and normal traffic.

Redundant and little contribution features of dataset need to be removed for effective and efficient process of IDS. Srilatha Chebrolu et.al.[6] used two feature selection algorithms. CART (Classification and Regression Trees), BN (Bayesian networks) and ensemble of CART and BN. The obtained results shows that Probe, DOS and Normal detected with 100% accuracy and U2R with accuracy 84% and R2L with 99.47%.

The most important problem in IDS is to increase the DR and decrease the FPR. Dongseong Kim et.al.[7] used (GA) Genetic algorithm and (SVM) Support Vector

Machine. Result shows that this approach gives optimal detection rate and false alarm rate.

Latifur Khan et.al.[8] proposed a method, called as Clustering Trees based on SVM (CTSVM). This method used to reduce the training set. This approach proved to outperform and work well compared to other techniques in terms of accuracy, (FNR) false negative rate and (FPR) false positive rate.

To secure the system new hybrid machine learning approaches are also considered. Chih-Fong Tsai et.al.[9] was proposed hybrid model for (ID) intrusion detection. This model was based on TANN (Triangle area based nearest neighbor). This approach gave higher DR, accuracy and low FPR in comparison to three single basic models SVM, combining of k - means and KNN, K-NN.

V. K. Pachghare et.al.[10] implemented an algorithm named as SOM (Self Organizing Map) based on NN (neural network) for intrusion detection system. The structure of SOM is single feed forward network, where each source node of the input layer is connected to all output neurons. The number of the input dimensions is usually higher than output dimension. Results shows that a when simple map trained on normal data, it detect the anomalous features. This approach is powerful because the self organizing map never needs to be told what intrusive behavior looks like. By learning to characterize normal behavior, it implicitly prepares itself to detect any unwelcome network activity.

R. Nakkeeran et.al.[11] proposed an approach for wireless network which used Bayesian classifier for training and testing purpose.

Hesham Altwaijry et.al.[12] used Bayesian probability to develop intrusion detection system. The system used naïve Bayesian classifier and its results were superior in detection rate.

Shi-Jinn Horng et.al.[13] used the hierarchical clustering algorithm to find the higher qualified, fewer and abstracted training instances for the SVM classifier so that they can classify accurately. Obtained results shows that the performance of Probe and DOS attacks detection rate and accuracy was better compared to other techniques.

Hichem Sedjelmaci et.al.[14] proposed hybrid ID clustered Wireless Sensor Network (WSN). Proposed algorithm consists of both anomaly and signature based detection with SVM.SVM separate data into normal and anomalous. Signature based techniques used to determine known anomalies patterns. The combination of anomaly and

signature based technique achieves high DR with (FPR) false positive and (FNR) false negative rate.

Many methods used to capture attack type for IDS but reducing FP and FN rate is a major issue. G. Gowrison et.al.[15] designed an intrusion detection system which enhanced the signature pattern by learn rules from the network behavior and classify the attacks within 0(n) complexity. They also examined that Adaboost is better than neural network. Future work of update, delete and adding the rules in the repository can be automatically and the performance can be measured through feedbacks and iterations.

Intruders affect the availability, confidentiality, integrity of cloud resources as well as services. Traditionally firewalls were used, which does not capable of detect insider attacks. Chirag N. Modi et.al.[16] proposed framework that integrate IDS in cloud front end and back end. It aims to reduce the impact of attacks, ensuring low false positive rate (FPR) and high detection rate (DR) within reasonable computation cost. L.Dhanabal et.al.[17] analyzed NSL-KDD dataset and used the WEKA tool for study the effectiveness to detecting the anomalies in the network traffic patterns. Correlation based Feature Selection method was used to reduce the dimension of dataset from 41 to 6 features. J48, SVM and Naïve Bayes classifier were used. It was observed that J48 classifier gives better performs compare to SVM and Naïve Byes when CFS feature selection technique was used.

High false alarm rate, low detection rate and size of data for processing are the challenge of IDS. Raman Singh et.al.[18] proposed IDS based on (OS-ELM) Online Sequential Extreme Learning Machine which uses ensemble of feature selection technique to reduce the features of dataset. Alpha profiling and beta profiling was used. Obtained results showed that this technique is effective way for network ID. There are many machine learning feature selection technique. Giorgio Roffo et.al.[19] proposed a feature selection technique which gives rank to each features according to the feature relevance. The proposed technique is known as Infinite feature selection technique (Inf-FS). This technique was compared against wrapper, filter and embedded method; It was observed that proposed technique was top perform than other.

Amira Sayed A. Aziz et.al.[20] applied different classifiers and their performance was compared to obtain best classifier. It also shows that a simple classifier such as Naïve

Bayes gives us better classification results in case of low represented attacks. Also Naive-Bayes Tree and Best-First Tree were better as compared to J48 (Weka implementation of C4.5) and Random Forest decision trees.

Akashdeep et.al.[21] proposed intelligent IDS. For the feature selection they combine the property of two algorithms that are Information gain and correlation. They select the relevant features by combining the obtained ranking of Information gain and Correlation. Pre-processing was done to remove duplicate and redundant data from the dataset. Reduced features are then fed to the forward neural network for training and testing purpose. Results were tested using five different subsets of KDD 99 datasets. Achieve results were outperform.

Ewa Roszkowska et.al.[22] described a multi criteria decision making (MCDM) method i.e. TOPSIS (Technique for Order Preference by Similarity to an Ideal Solution). Describe its steps and how to make matrix for TOPSIS.

Muhammad Fahad Umer et.al.[23] proposed a classification for Flow based IDS on the basis of different technique used for attacks in flow record. They also explained the pros and cons of this type of flow based intrusion detection. They describe the intrusion dataset are used for Flow based IDS. Statistical Techniques: Time series, Univariate and Multivariate; Machine learning Techniques: ANN (Artificial neural network), KNN (K nearest neighbor), SVM (Support vector machine), Clustering, DT (Decision tree); Other Techniques: Entropy, Threshold Value, Semantic Link Networks, Flow Signatures, Context. They describe the open issues and challenges of Flow based IDS.

Machine learning is also useful in the field of clinical and bioinformatics research area. Muhammad Akmal Remli et.al.[24] used the feature selection technique and clustering algorithm to distingue the cancer dataset into normal and tumor samples. They used Inf-FS (Infinite feature selection) technique to select informative gene in the dataset and k-mean clustering was used to cluster similar gene together and dissimilar genes in different clusters. This proposed technique achieved higher accuracy. 26 genes were identified for the further process. Colorectal cancer (CRC) and small round blue cell tumors (SRBCT) dataset were used to test the results.

Already existing algorithms cannot maintain the reliability (quality) in high speed network. Eduardo Viegas et.al.[25] proposed a BigFlow approach which has capability of processing large scale network traffic data with high speed packet rates.

To provide reliability BigFlow checks the output of classifier is valid or not, if the output is not valid or suspicious packet found then with the help of expert BigFlow changes the classification model. This approach requires 4% storage and 0.05 % - 4% training time compared to other approach.

A wireless device faces a security issues. R. Vijayanand et.al.[26] proposed a technique which used GA (genetic algorithm) for the selection of feature and multiple SVM (Support vector machine) is used as classifier. ADFA-LD and CICIDS2017 dataset were used for the testing of proposed technique. Proposed system results as less computation time and high accuracy with less communication overhead.

Redundancy in dataset and Irrelevant features in dataset cause a challenge for attack detection. Fadi Salo et.al.[27] proposed a hybrid technique which is a combination of IG (Information Gain) and PCA (Principle component analysis) for Irrelevant feature of dataset. They used multiple classifiers ensemble; those classifiers are IBK (Instance-based learning algorithms), MLP (multilayer perceptron) and SVM (Support Vector Machine). This technique gives better comparable analysis in terms of detection rate, time and accuracy.

Alex Shenfield et.al.[28] proposed an Artifical neural network approach which helps to detect the shellcode. It improve the performance of signature based IDS. Bytes data converted into integer values to feed into ANN. 'Magic numbers' were avoided due to easy for classifier to find. The ANN achieves perfect sensitivity as well as precision. This offline approach is used to detect shellcode pattern in the data.

For the security purpose Mehdi Moradi et.al.[29] proposed an offline intrusion detection approach. They used neural network for attack detecting purpose, they also detect the type of attack in the system. The final result concluded that with the help to two hidden layers in neural network can achieve 91% accuracy and with the help of one hidden layer accuracy was 87%.

False alarm rate and detection accuracy are the major problem in Intrusion detection system. Mehrnaz Mazini et.al.[30] proposed a hybrid approach for anomaly network based IDS (A-NIDS) which used the ABC (Artifical Bee Colony) for feature selection and AdaBoost for low (FPR) false positive rate and high (DR) detection rate.To handle massive data is a big challenge for Network Intrusion detection system. Selvakumar B et.al.[31] proposed a method for dimensionality reduction. They used ensemble of wrapper with Bayesian network, C4.5 and MI (Mutual Information) for

feature selection. KDDCUP 99 dataset have 41 features originally but this method reduced it to 10 features which reduce the computational cost of classifier.

Table 2.1: Related Work

Sr. No	Author Name	Year	Technique Used	Dataset Used	Remarks
1.	Susan M. Bridges et. al. [1]	2000	Genetic Algorithm		Genetic Algorithm was improved the performance of system by extracting relevant features from dataset and tuning the membership function parameters.
2.	Leonid Portnoy et. al. [2]	2000	Clustering	KDD CUP99	Presented a framework for intrusion detection with unlabelled data using clustering.
3.	Adhitya Chittur et. al. [3]	2001	Genetic Algorithm	Knowledge Discovery in Database (KDD) Cup	This algorithm was tested real-world simulation to gauge its effectiveness under unpredictable conditions. It was concluded that the use Genetic algorithm successfully get correct DR and low (FPR) false positive rate.
4.	B. Balajinath et. al. [4]	2001	GBID (Genetic Algorithm Based Intrusion Detector)		This algorithm is based on "Learning the individual behavior". Detection of the intrusions from the system by using this approach was with accuracy 96.8% and FPR 3.2%.
5.	Khaled Labib et. al. [5]	2002	Self Organizing Map		Discussed the structure of SOM. Result showed the classify DOS attack with Normal traffic graphically.
6.	Srilatha	2004	Bayesian	DARPA	Ensemble of BN and

	Chebrolu et. al. [6]		Network (BN), Classification and Regression Tree (CART)	Benchmark	CART for feature selection. Normal, Probe and DOS detected with 100 %, U2R with 84% and R2L with99.47% accuracy.
7.	Dongseong Kim et. al. [7]	2005	Genetic Algorithm and Support Vector Machine	KDD CUP 1999	Genetic Algorithm enhances the SVM performance. It gives optimal feature subset.
8.	Latifur Khan et. al. [8]	2005	SVM and DGSOT (Dynamically Growing Self-Organizing Tree)	1998 DARPA (Defense Advanced Research Projects Agency)	Combine the DGSOT and SVM to enhance the training time of SVM. Proposed technique outperform the Rocchio Bundling technique
9.	Chih-Fong Tsai et. al. [9]	2009	Triangle area based nearest neighbors (TANN)	KDD CUP	K-mean clustering was used to find the cluster corresponding to the attack. TANN provides higher accuracy, detection rate and less false alarm rate.
10.	V. K. Pachghare et. al. [10]	2009	SOM (Self Organizing Maps)	DARPA	Explained the SOM architecture. Its advantages and disadvantages.
11.	R. Nakkeeran et. al. [11]	2010	Data mining technique		Proposed system used current node, neighbor node and global network. Current node results were send to the neighbor node. Detection rate increased and false alarm rate reduced.
12.	Hesham Altwaijry et. al. [12]	2011	Bayesian Probability	KDD	Bayesian detect attack with superior detection rate.
13.	Shi-Jinn Horng et. al. [13]	2011	Clustering, SVM	KDD CUP 1999 training set	Combine Hierarchal clustering, Simple feature selection and SVM classifier. Obtained result showed higher accuracy with better

					detection of DOS and Probe attack.
14.	Hichem Sedjelmaci et. al. [14]	2011	Support vector machine, Signature based detection	KDD 99	Proposed distributed hybrid IDS for wireless sensor network. Combination between Signature detection and SVM achieved higher detection rate and low false alarm rate.
15.	G. Gowrison et. al. [15]	2012	Neural Network, Adaboost	KDD CUP99	Classification of Adaboost is better than Neural network.
16.	Chirag N. Modi et. al. [16]	2012	Signature apriori algorithm		Proposed algorithm detect attack in network cloud at the front as well as back end. Result achieved low false alarm rate and low computational cost.
17.	L.Dhanabal et. al. [17]	2015	Correlation–based Feature selection, J48, SVM, Naïve bayes	NSL-KDD	CFS used for dimension reduction with high accuracy and low false alarm rate. J48 classifies all attack with better accuracy.
18.	Raman Singh et. al. [18]	2015	Online Sequential Extreme Learning Machine (OS-ELM)	NSL-KDD, Kyoto University Benchmark	Alpha and Beta profiling used to reduce the size of training dataset. 98.66% and 97.67% accuracy of Binary NSL-KDD and Kyoto University achieved respectively. Achieved 96.37% accuracy for Kyoto University dataset.
19.	Giorgio Roffo et. al. [19]	2015	Infinite feature selection (Inf-FS)		Inf-FS used to extract relevant features from dataset on the basis of ranking of features.
20.	Amira Sayed A. Aziz et. al. [20]	2016	Naïve Bayes, Naïve Bayes Tree, Random Forest, Best	NSL-KDD	No single classifier can detect all type of attacks. Naïve Bayes is good for low

			First Tree		representation attacks. Naïve Bayes and Best First used for anomaly detection traffic.
21.	Akashdeep et. al. [21]	2017	Information gain, correlation, Artificial Neural Network (ANN)	KDD99	Combination of information gain and correlation ranking was used for feature selection and ANN used as a classifier.
22.	Ewa Roszkowska et. al. [22]		TOPSIS (Technique for Order Preference by Similarity to an Ideal Solution)		Describes the MCDM (Multi criteria decision making) technique and its matrix representation.
23.	Muhammad Fahad Umer et. al. [23]	2017	Statistical (Time series, Univariate and Multivariate), Machine Learning (ANN, KNN (K-nearest neighbor), SVM, Clustering, DT (Decision tree), Techniques	Sperotto, ISOT, TUIDS (Tezpur University Intrusion Dataset), CTU-13 and SSHCure intrusion datasets	Survey of Flow based IDS. Described the available flow based datasets. Presented the taxonomy on the basis of different techniques. Challenges and future research are in the field of flow based IDS.
24.	Muhammad Akmal Remli et. al. [24]	2017	K- Mean and Infinite Feature selection technique (Inf-FS)	colorectal cancer (CRC) and small round blue cell tumors (SRBCT) dataset	For feature selection used K-mean and Infinite feature selection technique. 26 genes are identified for CRC and SRBCT.
25.	Eduardo Viegas et. al. [25]	2018	BigFlow	MAWIFlow	Proposed approach, BigFlow aimed to achieve higher detection throughput.
26.	R. Vijayanand et. al. [26]	2018	Genetic Algorithm and Multi Support Vector	ADFA-LD and CICIDS2017 dataset	Genetic algorithm used for feature selection and SVM used for classification of wireless network.

			Machine		Achieved result shows higher accuracy, less communication overhead and less computational complexity
27.	Fadi Salo et. al. [27]	2018	IG, PCA, SVM, IBK (Instance-based learning algorithms), MLP (multilayer perceptron)	ISCX 2012, NSL-KDD, and Kyoto 2006+	Combining the Information agate and Principal component analysis with SVM, IBK and MLP. Result achieved high accuracy with low false alarm rate.
28.	Alex Shenfield et. al. [28]	2018	Artificial neural network		Proposed artificial neural network distinguished benign and malicious activity accurately
29.	Mehdi Moradi et. al. [29]		Multi Layer Perceptron (MLP) artificial neural network	DARPA (Defense Advanced Research Projects Agency)	Optimal neural network structure finds out using different hidden layer. Resulted accuracy with 2 hidden layers and 1 hidden layer was 91% and 87% respectively.
30.	Mehrnaz Mazini et. al. [30]	2018	Artifical Bee Colony (ABC) and AdaBoost	NSL-KDD and ISCXIDS2012	ABC for feature selection and AdaBoost for unbalanced data and to evaluate and classify the features.
31.	Selvakumar B et. al. [31]	2018	Firefly technique, C4.5 and Bayesian network classifier	KDD CUP 99	10 features were selected out of 41 and result shows improved accuracy by using selected features.

CHAPTER 3
PROBLEM STATEMENT

Research Gaps

There are some research gaps summarized below:

1. Imbalanced of Network Traffic Dataset i.e. Number of anomalous connections are lesser than normal connections.
2. Due to large number of instances in dataset (Hugeness) and large number of features, it make difficult to apply classic soft computing technique on whole network traffic dataset.
3. Due to high FPR (false positive rate), accuracy of IDSs will be low.
4. Higher False Alarm rate of IDS.
5. Hugeness of dataset takes large amount of time due to which Intrusion Detection time will be high.

Network Traffic Dataset issues

Some issues of network traffic dataset need to be addressed. Now a day's everyone is using internet and try to secure their data. For the security purpose, need to store all the required information which increase the size of dataset whose maintenance, storage, and time requirement for processing is a major issue. The normal events are more compared to attack events in host and network. Due to which most of the datasets have normal samples more compared to number of attack samples which is called imbalance captured dataset. To resolve these issues newer techniques like Machine Learning are used. Datasets have large number of features which all are not always necessarily, so there is so many feature selection techniques available who helps in reducing the number of features by extracting the relevant attributes from the dataset and still preserved quality or characteristics of dataset. It reduces the required time and memory storage [18]. These issues are discussed below sub-section:

Size of dataset (Hugeness)

Pre-processing step is required to detect attack in network traffic dataset. This step removes the missing values and noisy values. Due to the hugeness of dataset various

sampling technique used to limit the size. There are many sampling techniques available which are used and do not affect the characteristics of dataset.

Large Feature Dataset

Due to more number of attributes in a dataset the process time of dataset will be high and required large storage space in memory. There are so many techniques available to reduce the features from datasets, these techniques called as feature selection techniques. Feature selection technique is used to retrieve the relevant features from the dataset and the quality is still preserved. Some of the feature selection techniques (FST) are Genetic Algorithm (GA), Decision Tree (DT), Particle Swarm optimization (PSO), Infinite feature selection technique (INFFST) etc.

Imbalance of Dataset

Most of the dataset consists of normal connections numbers more compared to the attack connections, So at the time of training the model effectively learn the normal activity of dataset but difficult to learn all the attack behavior due to less attack information [30].

Research Objective

1. To suggest the appropriate feature selection technique to reduce features of network traffic dataset.
2. Design horizontal abridging algorithm to optimally reduce size of network traffic dataset.
3. To develop machine learning based intrusion detection system with proposed abridging algorithm.
4. To validate and evaluate the performance of proposed abridging algorithm and IDS using various parameters such accuracy, recall, precision, f1-score, t-value and detection time.

CHAPTER 4
METHODOLOGY

Rapid usage increment of computers, security is the major challenge. Intrusion detection is the necessary step to secure the system. In intrusion detection system, the model learns that the connection is normal or some attack type. This thesis proposes an abridging algorithm to reduce the dimensionality (Features as well as Instances) of dataset without affecting its quality or accuracy. If the dataset size will be reduced then it will take less amount of time in training of IDS i.e. detection time, detection rate and required less storage space, it helps in reducing the false positive rate (FPR). Infinite Feature Selection technique (Inf-FS) used to ranking the features according to the relevance of features in the dataset and proposed algorithm i.e. horizontal abridging algorithm is used to reduce the dataset instances. SVM (Support Vector Machine) algorithm is used for classification purpose. It is consider one of the most successful classification algorithm. Propose technique enhancing the training time of SVM meanly for huge dataset using hybrid of Infinite feature selection technique and Horizontal abridging technique. The result of thesis is analyzed using accuracy, recall, precision, f1-score, time and t-value. The flow of thesis shown in figure 4.1 below:

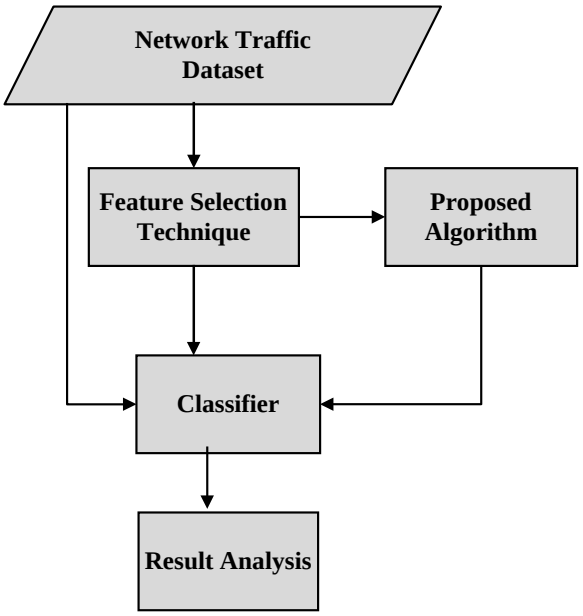

Figure 4.1: Flow of thesis

Dataset

Network traffic dataset is a collection of rules or collection of traffic flow information to generate collection of system logs, packets or network protocol logs etc. Dataset is a crucial part of Intrusion Detection System (IDS). IDS trained using properly labeled dataset. Datasets are either collected or generated for research purpose. These datasets are used worldwide by researchers. The Standard Network Analysis project (SNAP) creates network traffic datasets by using internet work, web graphs, various nodes of social networks, web graph etc. This dataset is collection of different dataset like citation network, Amazon network, communication network, social network, road network, Internet network etc. Some of these collections are unlabeled while others are labeled.

Dataset used in this thesis are mentioned in table 4.1.

Table 4.1: Required Dataset

Sr. No.	Dataset	Remarks
1.	NSL-KDD 2009	Labeled Dataset. No redundant instances
2.	Kyoto University Benchmark 2009 Dataset	Labeled Dataset.

From the research studied it has been found that mostly unlabeled datasets are found. Unlabeled dataset makes difficult to evaluate the performance of purposed IDS. For performance evaluation NSL-KDD dataset is best suitable.

NSL- KDD 2009 Dataset

NSL-KDD 2009 (Network Security Laboratory-Knowledge Discovery and Data Mining) are proposed to overcome KDD Cup dataset for research in IDS issues. Cyber Systems and Technology Group of MIT Lincoln Laboratory collected this dataset. A connection is a sequence of TCP packets that's starting and ending with same well defined time. 148517 unique data connections consist in this dataset. NSL-KDD 2009 dataset contain 1 class label with 41 unique features. Some features are categorical while some are continuous. Categorical feature values comes under

definite set while Continuous features values belong to indefinite set. All Features other than 2, 3, 4, 7, 12, 14, 15, 21, 22 and 42 are continuous. Duplicate and redundant records are removed for reducing the biasness of classifiers. Features of NSL-KDD 2009 dataset shown in table 4.2.

Table 4.2: NSL-KDD 2009 Dataset Features Details

Feature No.	**Feature name**	**Feature Description**
F1	Duration	length (number of seconds) of the connection
F2	Protocol type	type of the protocol, e.g. tcp, udp, icmp etc.
F3	Service	network service on the destination, e.g., http, telnet, etc.
F4	Flag	normal or error status of the connection
F5	Src_bytes	number of data bytes from source to destination
F6	Dst_bytes	number of data bytes from destination to source
F7	Land	1 if connection is from/to the same host/port; 0 otherwise
F8	Wrong_fragment	number of “wrong” fragments
F9	Urgent	number of urgent packets
F10	Hot	number of ``hot'' indicators
F11	num_failed_logins	number of failed login attempts
F12	Logged_in	1 if successfully logged in; 0 otherwise
F13	Num_compromised	number of ``compromised'' conditions
F14	Root_shell	1 if root shell is obtained; 0 otherwise
F15	Su_attempted	1 if “su root” command attempted; 0 otherwise
F16	Num_root	number of ``root'' accesses
F17	Num_file_creations	number of file creation operations
F18	Num_shells	number of shell prompts
F19	Num_access_files	number of operations on access control files
F20	Num_outbound_cmds	number of outbound commands in an ftp session
F21	Is_host_login	1 if the login belongs to the ”hot” list; 0 otherwise
F22	Is_guest_login	1 if the login is a “guest” login; 0 otherwise
F23	Count	number of connections to the same host as the current connection in the past two seconds
F24	Srv_count	number of connections to the same service as the current connection in the past two seconds
F25	Serror_rate	% of connections that have “SYN”' errors
F26	Srv_serror_rate	% of connections that have “SYN” errors
F27	Rerror_rate	% of connections that have “REJ” errors
F28	Srv_rerror_rate	% of connections that have “REJ” errors

Feature No.	Feature name	Feature Description
F29	Same_srv_rate	% of connections to the same service
F30	Diff_srv_rate	% of connections to different services
F31	Srv_diff_host_rate	% of connections to different hosts
F32	Dst_host_count	count of connections having the same destination host
F33	Dst_host_srv_count	count of connections having the same destination host and using the same service
F34	Dst_host_same_srv_rate	% of connections having the same destination host and using the same service
F35	Dst_host_diff_srv_rate	% of different services on the current host
F36	Dst_host_same_src_port _rate	% of connections to the current host having the same src port
F37	Dst_host_srv_diff_host_ rate	% of connections to the same service coming from different host
F38	Dst_host_serror_rate	% of connections that have “SYN" errors for destination host
F39	Dst_host_srv_serror_rate	% of connections to the current host that have “SYN” error
F40	Dst_host_rerror_rate	% of connections to the current host that have RST errors
F41	Dst_host_srv_rerror_rate	% of connections to the current host and specified service that have an RST error
F42	Class_ label	Label for each connection e.g. normal, attack

Kyoto University Benchmark 2009 Dataset

This dataset collected from November, 2006 through August 2009. This dataset consists of 43503 unique data connections, out of which 24933 are normal and 18570 are attacks connections. All Redundant and duplicate connections are processed and then removed. It consists of 1 class label with 16 features. All features number other than 2, 14, 15, 16, and 17 are continuous.

All the features name of Kyoto University benchmark 2009 dataset is shown in Table 4.3.

Table 4.3: Kyoto University Benchmark Dataset features

Feature No.	Feature Name	Feature No.	Feature Name
KF1	Duration	KF10	Dst host srv count
KF2	Service	KF11	Dst host same src port rate

Feature No.	Feature Name	Feature No.	Feature Name
KF3	Source bytes	KF12	Dst host serror rate
KF4	Destination bytes	KF13	Dst host srvserror rate
KF5	Count	KF14	Flag
KF6	Same srv rate	KF15	Source Port Number
KF7	Serror rate	KF16	Destination Port Number
KF8	Srvserror rate	KF17	Label
KF9	Dst host count	--------	----------

Experimental Details

Three experiments were performed with different procedure by using two different datasets for the proposed algorithm. System requirement and different experiments are explained in this section.

System Requirement

All experiments are implemented with Matlab (Version R2012b) and R language. Experiments are performed on computer system configured with Microsoft Window 7 (64 bit) professional operating system, Intel Core i5 3.20 Ghz processor, physical memory of 4 Giga Byte, and 320 Giga Byte hard disc.

Experiment 1: Used Full dataset

First experiment is performed with network traffic dataset. The whole dataset is divided into two parts i.e. training and testing dataset. Training Dataset contains 70% of network traffic dataset while testing dataset contains remaining 30% of network traffic dataset. Apply Support Vector Machine (SVM) classification to evaluate the performance by using follow parameters: accuracy, Recall, Precision, time, f1-Score and t-value using r language in R Studio. This experiment is applied on two different dataset i.e. NSL-KDD 2009 and Kyoto University benchmark 2009 dataset.
The flow of experiment1 was shown in figure 4.2.

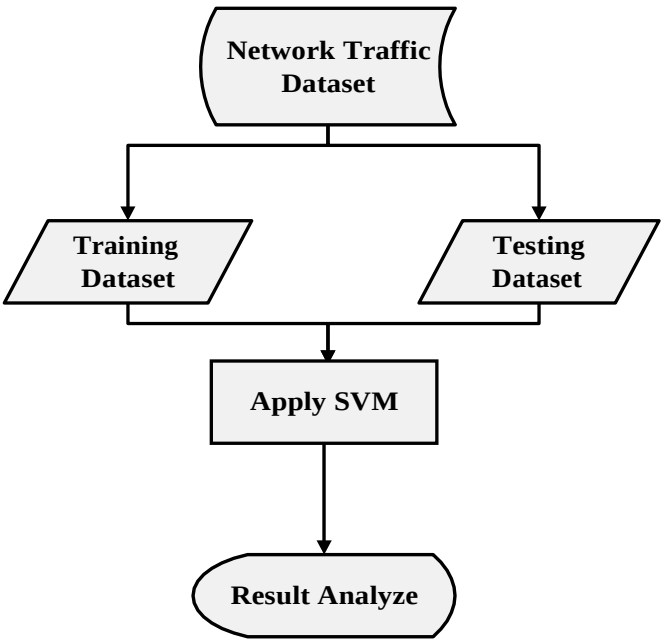

Figure 4.2: Experiment on Full Network traffic dataset

Experiment 2: Apply Feature Selection Technology

This experiment is performed to select relevant features from Network traffic dataset. It helps in reducing the features of datasets.

Firstly Infinite feature selection technique was applied to extract relevant features from dataset. INFFST gives subsets features from the dataset as a result. The reduced dataset was divided into two parts: Training reduced dataset and testing reduced dataset named as feature reduced training dataset and feature reduced testing dataset respectively. Apply Support vector machine (SVM) classification on feature training reduced dataset to evaluate the performance of reduced dataset. Performance evaluation parameters are accuracy, recall, precision, time, f1-score and t-value.

Compared the performance of experiment 1 and experiment 2, It was observed that the accuracy remains nearby but the experiment2 takes less amount of time than the experiment1. It concluded that all the features of the dataset do not necessary for the learning or training process of model. The infinite feature selection technique extracted features are relevant and the dataset quality or characteristics is still preserved.

The flow of experiment 2 is shown in figure 4.3.

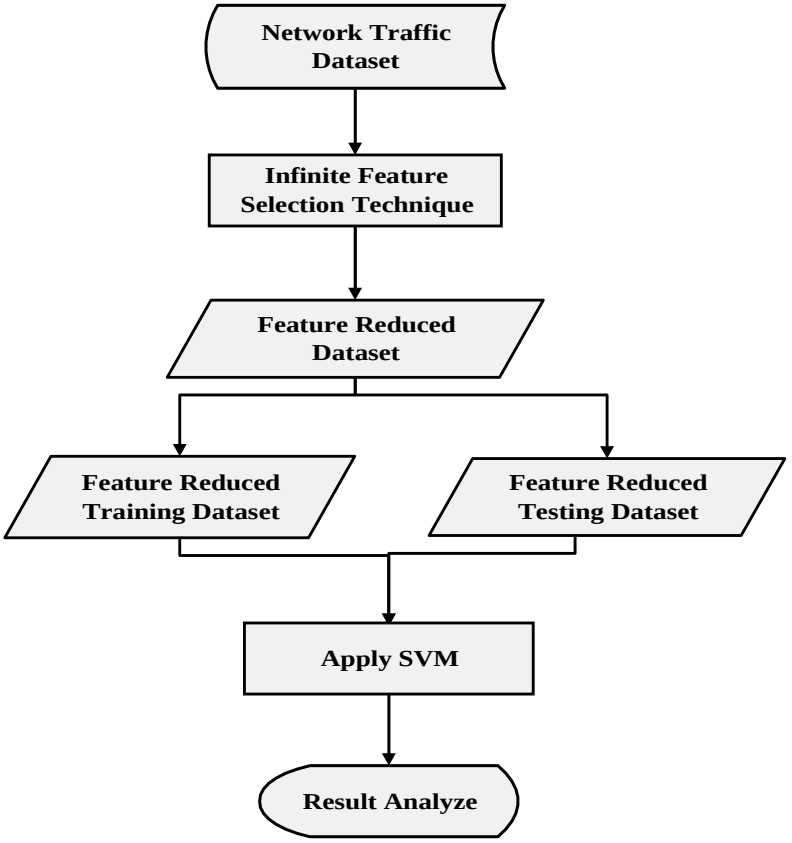

Figure 4.3: Apply Infinite Feature Selection Technique

4.3.3 Experiment 3: Proposed Horizontal abridging algorithm

Experiment 3 was performed to reduce the size of instances according to the similarity between the instances. This experiment will carried out the resulted dataset of experiment 2 i.e. feature reduced training dataset.

All the normal and attack label instances of feature reduced dataset saved into two different variables for execution of experiment. After the division of the dataset DBSCAN (Density - based spatial clustering of applications and noise) is applied on both the variables. DBSCAN is a clustering based algorithm. It forms clusters based on the similarities between the instances according to the threshold value.

If any cluster has more than 5 instances in it then it replaces that instances with their mean value, + standard deviation, - standard deviation, Maximum of minimum value index instance and Maximum of maximum value index instance. After that all the cluster instances were merged and randomized and it named as instance reduced

training dataset. The flow of experiment 3 was shown in the figure 4.4. Trained the system by SVM using Instance reduced training dataset and tested on Feature reduced testing dataset (from Experiment 2).

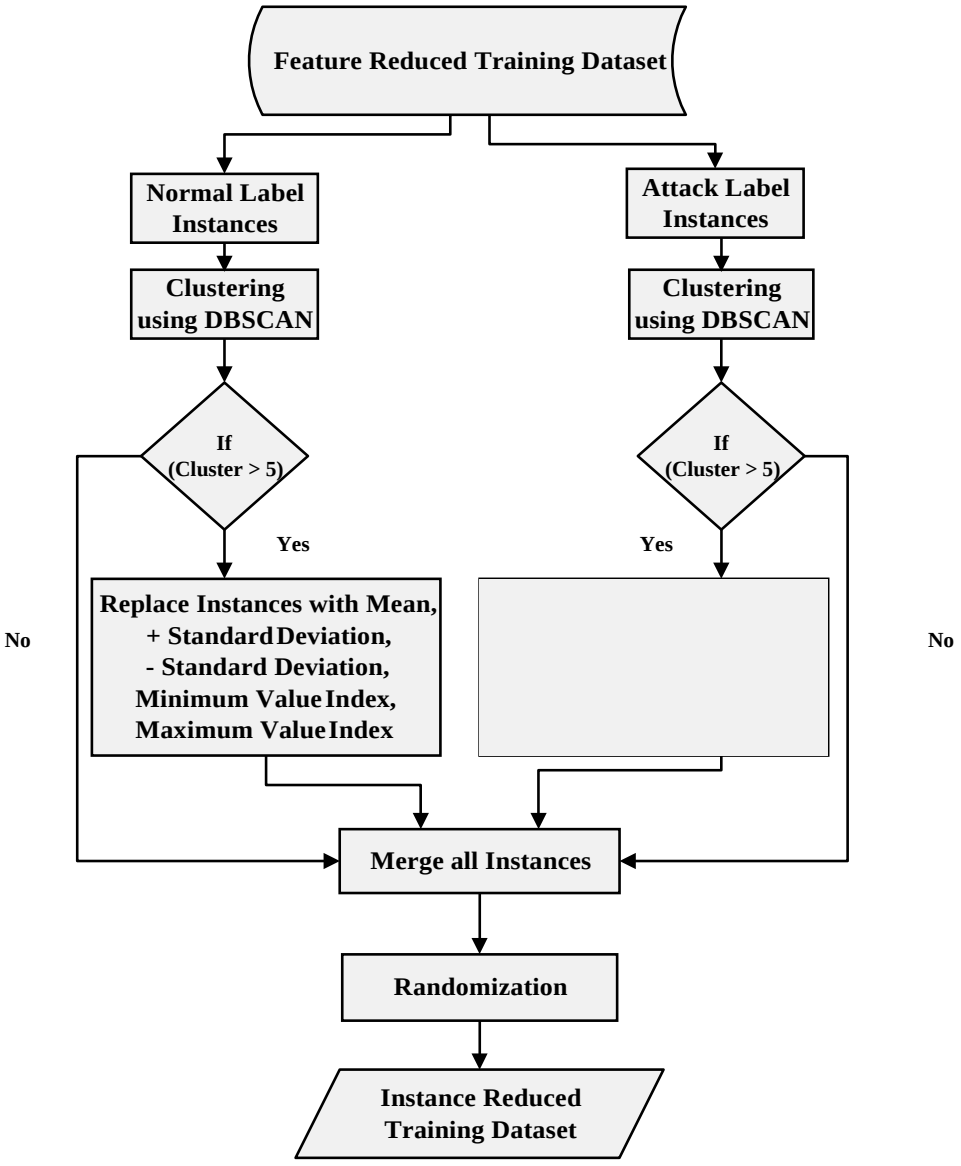

Figure 4.4: Horizontal abridging algorithm

Different threshold values are used and achieve different evaluation parameters values. TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) is used to select one of the best threshold values on the basis of overall best evaluation parameters values [21].

Compared the performance of all the three experiments and it was observed that experiment1 takes huge amount of space and time. Experiment 2 removes the irrelevant features from dataset and reduces the size of dataset by which it requires less storage space and less amount of training time with nearby accuracy. Experiment 3 merges the similar instances by which the dataset size reduces the instances without effecting dataset quality.

Evaluation Criteria for Performance

Standard performance evaluation parameters are used to check the results of all the experiments of this thesis. Performance parameter of proposed methodology calculated using Confusion matrix. In these experiments, normal label, value '1' represents positive event and attack label, value '2' represents negative events. The result of predicted versus actual class shows in confusion matrix. The (TP) True Positive, (FP) False Positive, (TN) True Negative and (FN) False Negative are defined as follows:

- **True Positive (TP):-** TP is number of events actually normally and it also classified as a normal connection. TP should be more in comparison to FP, FN.
- **False Positive (FP):-** This is type 'I' error. FP is the number of connections predicted normal but actually they are attack connections. FP rate should be minimizing to design IDS. In this situation, attack patterns are falsely identified as normal patterns by IDS which is not preferable.
- **False Negative (FN):-** This is also known as simply false alarms or type 'II' error. FN is the number of connections predicted attack but actually they are normal connections. All these connections are classified wrong. In this situation normal traffic patterns are falsely alarmed as attack patterns. However, for anomaly based IDS, False Positive is more important than false negative however ideal IDS should avoid false positives as well as false negative.
- **True Negative (TN):-** TN represents the numbers of attack event connections which have been classified as attack connection correctly. Higher TN is

always preferable along with high TP. Table 4.4 shows the binary classifier parameters in confusion matrix.

Table 4.4: Confusion matrix for binary classifier

	Actual class		
Predicted class		Normal	Attack
	Normal	TP	FP
	Attack	FN	TN

Performance evaluation parameters are discussed below which are for the proposed algorithm:

- **Accuracy:** It is the average of all the connections which are correctly classified with all the classified connections. It represents the how well algorithm works in terms of correctly classification. Accuracy is calculation using Eq. 4.1

$$\text{Accuracy} = (TP + TN) / (TP + FP + FN + TN) \quad (4.1)$$

- **(TPR) True Positive Rate:** Recall or Sensitivity is the other name of TPR. It is the ratio of how many connections are classified to normal to the actually total number of normal connections. For Anomaly based IDS activities high TPR means system is able to efficiently identify the normal connections. The TPR calculated formula is in Eq. 4.2

$$TPR = TP / (TP + FN) \quad (4.2)$$

- **(FPR) False Positive Rate:** It is the ratio or proportion of the connection classified as normal but actually they are attacks to the total of actually attack connections. Its value should be less in IDS.
  The calculated formula for FPR is shown in Eq. 4.3

$$FPR = FP / (FP + TN) \quad (4.3)$$

- **Precision:** Relevant selected proportion is shown by precision. It is the proportion that how many connections are classified as normal to the total

number of predicted normal connections. The precision calculated formula is in Eq. 4.4

$$Precision = TP / (TP + FP) \quad (4.4)$$

- **F1- Score:** It is calculated using Recall and Precision. It measure test accuracy which calculate harmonic mean of TPR/Recall and Precision. The formula used in the Eq. 4.5

$$F1 – Score = (2 * Recall * Precision) / (Recall + Precision) \quad (4.5)$$

Algorithm Used

Some machine learning algorithms were used in this thesis. These algorithms are explained below:

SVM (Support Vector Machine)

It is a supervised machine learning algorithm [7]. It is mostly used for classification challenges. In SVM kernel functions are used which helps to converts not separable problem to separable problem. Kernel function takes data as input and converts it into required form. SVM uses different type of kernels. For example Polynomial, Linear, Non linear, Radial function (RBF) and Sigmoid. In this thesis used Radial Function. $K(x, y) = < f(x), f(y) >$. K is the kernel function. f is a mapping. x, y are n dimensional input. $<f(x), f(y)>$ is denotes the dot product.

Infinite Feature Selection Technique

Infinite feature selection is a graph based technique which uses the property of power series of matrices. It evaluates the importance of each feature with respect to all other features taken together. Affinity graph is used to map each feature. In a graph, node is used to represent the features and edge is used to represent the relationship between the features. Each path of certain length l over a graph means l features selection. Varying this path length tends to an infinite number permits the investigation of the importance of each possible subset of features. This algorithm assigns a final score to

each features, Score represents the importance of each features with respect to other. Ranking is in descendant order as the outcome of the Inf-FS [19].

Proposed Algorithm: Horizontal Abridge Algorithm

Horizontal abridge is a proposed algorithm for rows reduction. In this algorithm all the instances are divided into normal and attack label connections. Apply DBSCAN algorithm for the clustering purpose on both the normal as well as attack connection variable. If any cluster having more than five instances than replace that cluster instances with the mean, + standard deviation, - standard deviation, Maximum number of minimum value instance, maximum number of maximum index value instance. Merge the entire updated instance and formed a new reduced dataset.

4.6.3.1 DBSCAN (Density Based Spatial Clustering of Applications with Noise)

It is a famous and common clustering algorithm. Prior knowledge of number of clusters does not required like in K-Mean algorithm. This algorithm reduces the number of dataset instances by clubbing redundant, similar and duplicate instances. Some time due to memory constraint it is required to reduce the size of dataset without affecting its quality. This algorithm requires two parameters. These parameters are min number of connection required (minPoints) and distance threshold (dstThreshold) [18]. This algorithm is robust to outliers and noise.

CHAPTER 5
RESULT ANALYSIS

In the proposed abridging algorithm Infinite Feature Selection technique and Horizontal abridge algorithm is used to improve the performance of training model. SVM is used as a classifier. The dataset firstly categorized into two parts i.e. training and testing. Training part consists of 70% of network traffic dataset and testing part consists of remaining 30% of network traffic dataset. Proposed algorithm enhances the time complexity and space complexity without affecting the quality of dataset.

First experiment is performed with full network traffic dataset. Second experiment is performed using Infinite Feature Selection to extract the relevant features from dataset. Third experiment is performed using Horizontal abridging algorithm to reduce the number of instance of dataset using DBSCAN technique. Two different network traffic datasets are used for the analyzed purpose, NSL- KDD 2009 and Kyoto University benchmark 2009. Accuracy, Recall, Precision, F1–score, number of instances reduced, number of attribute reduced and T- value are discussed. This chapter shows the obtained results. Obtained results are discussed below:

Result Analysis: Kyoto University Benchmark 2009 Dataset

The Kyoto university dataset consists of 43503 unique connections and 16 features with 1 class label. The dataset divided into training and testing part so the training part i.e. 70% of Kyoto university benchmark 2009 dataset consists of 30452 unique connections and 30% of remaining Kyoto university dataset i.e. 13051 connections are for testing purpose.

Experiment 1: The result of full Kyoto university dataset i.e. Accuracy, Recall, Precision, F- score and time described in below table 5.1.

Table 5.1: SVM on Kyoto University Benchmark 2009 Dataset

Sr. No.	Performance parameter	Result (in %)
1.	Accuracy	87.56
2.	Recall	80.34

3.	Precision	97.81
4.	F1- Score	88.21
5.	Time	118.8 sec

Experiment 2: Apply Infinite feature selection technique (INFFST) to select the relevant features from the dataset and after getting the relevant features applied SVM on selected features dataset and analyze its performance parameters. The selected features of Kyoto university dataset is shown in below table 5.2.

Table 5.2: Selected Kyoto University Benchmark 2009 features using INFFST

Dataset Name	Selected Features
Kyoto University Benchmark 2009 dataset	1,2,3,4,5,7,8,9,10,13,14,15,16

After selection of 13 features out of 16 features with 1 class label apply SVM to train the model. The performance parameters of SVM on selected features of Kyoto university dataset is shown in table 5.3.

Table 5.3: SVM on Kyoto University Benchmark 2009 selected features

Sr. No.	Performance parameter	Result (in %)
1.	Accuracy	87.86
2.	Recall	80.50
3.	Precision	97.79
4.	F- Score	88.30
5.	Time	104.4 sec

Experiment 3: In this experiment apply horizontal abridging algorithm which use DBSCAN with five different threshold values on training part of INFFST applied dataset.

30452 unique connections with 13 features and 1 class label are divided into normal and attack label instances. 17501 are the normal connections and 12951 are the attack connections. DBSCAN forms clusters using threshold value and using horizontal reduced technique merge the similar connections for instance reduction. The

performance parameters i.e. percentage of training row reduction, Accuracy, Recall, Precision, F1- score, Time and T – value were observed using SVM classifier. T–value is the t test value which is used to determine the independency between two samples. The obtained results are shown in table 5.4.

Table 5.4: Horizontal abridging algorithm obtained results for Kyoto University Benchmark 2009 dataset

Sr. No.	**Thresho -ld**	**Training Reduce Rows (In %)**	**Accuracy (In %)**	**Recall (In %)**	**Precisi -on (In %)**	**F1-Score (In %)**	**T-Value**	**Time (In sec)**
1.	0.05	11.31	87.51	80.6	96.8	87.9	0.163	76.50
2.	0.08	13.86	87.55	80.9	96.6	88.0	0.202	72.82
3.	0.10	15.20	87.54	80.8	96.6	87.9	0.223	68.75
4.	0.20	19.11	87.40	80.6	96.7	87.9	0.286	60.33
5.	0.30	22.76	87.35	80.5	96.7	87.8	0.347	55.26
6.	0.50	26.70	87.11	80.1	96.7	87.6	0.415	50.10

It was observed that the accuracy of full dataset is 87.56%. After the use of INFFST the accuracy is 87.86% by using less number of features. Horizontal Abridging Algorithm reduced the instances of dataset and obtained accuracy between 87.51% to 87.11% when the threshold value was between 0.05 to 0.50 respectively.

With the help of TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) try to find the best threshold value which suggests the overall best parameters values i.e. Row reduction, Accuracy, Recall, Precision, F1-score and T-value. This technique is used when there are many options to choice. It is the multi criteria decision technique. TOPSIS required weight for each parameter according to the priority of parameters. For the use of TOPSIS assigned weights are 2 for Training reduced rows, Accuracy and T- value; 1 for Recall, Precision, time and F1- score. It means that Training reduced rows, Accuracy and T-value is twice time more important than Recall, Precision, time and F1-score. Obtained result for all the threshold value by using TOPSIS is shown in the table 5.5 below.

Table 5.5: TOPSIS result for Kyoto University Benchmark 2009 Dataset

Sr. No.	Threshold	TOPSIS Value
1.	0.05	0.5181
2.	0.08	0.5222
3.	0.10	0.5102
4..	0.20	0.5218
5.	0.30	0.5049
6.	0.50	0.4762

The maximum obtained TOPSIS value i.e. 0.5222 means that the best suited Threshold value is 0.08 for all the performance parameters for Kyoto University benchmark 2009 dataset. So, even after 13.86% of rows reduction is achieve 87.55% accuracy.

Graphically represents the comparison between the original Kyoto university benchmark 2009 dataset and horizontal abridging algorithm best obtained results according to the TOPSIS in figure 5.1.

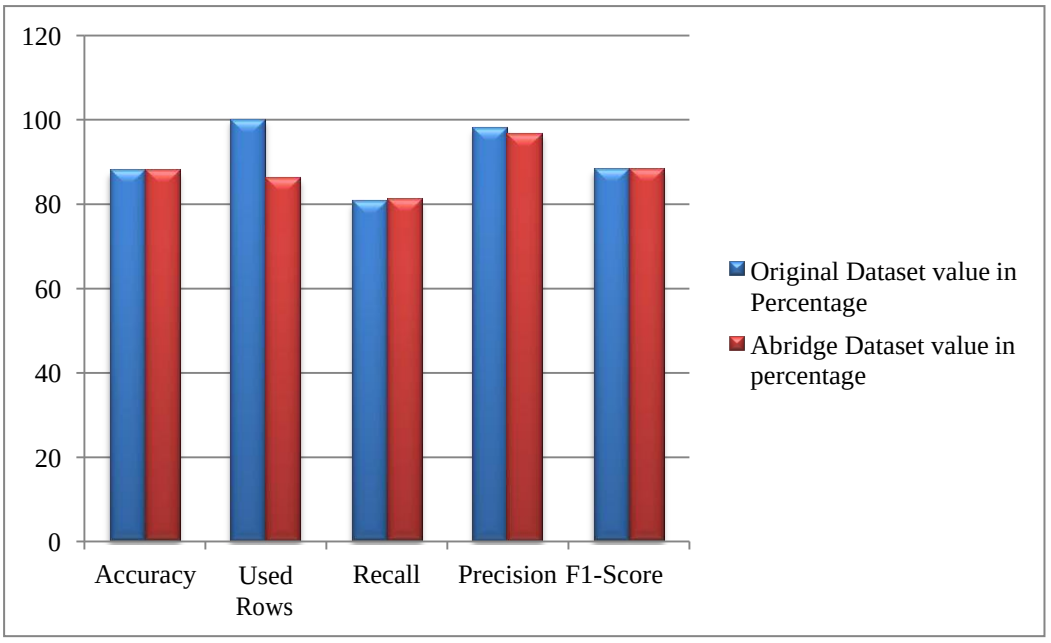

Figure 5.1: Original Kyoto University Benchmark 2009 Dataset versus Abridge Dataset value

Figure 5.1 shows that the Accuracy, recall, precision and F1-score are nearly equal in both the cases (Original and Abridge dataset) but the total rows reduction has major difference.

In figure 5.2 graphically visualize the increment of threshold value with performance parameters in percentage i.e. Accuracy, Total row reduction, Recall, Precision, F1-Score.

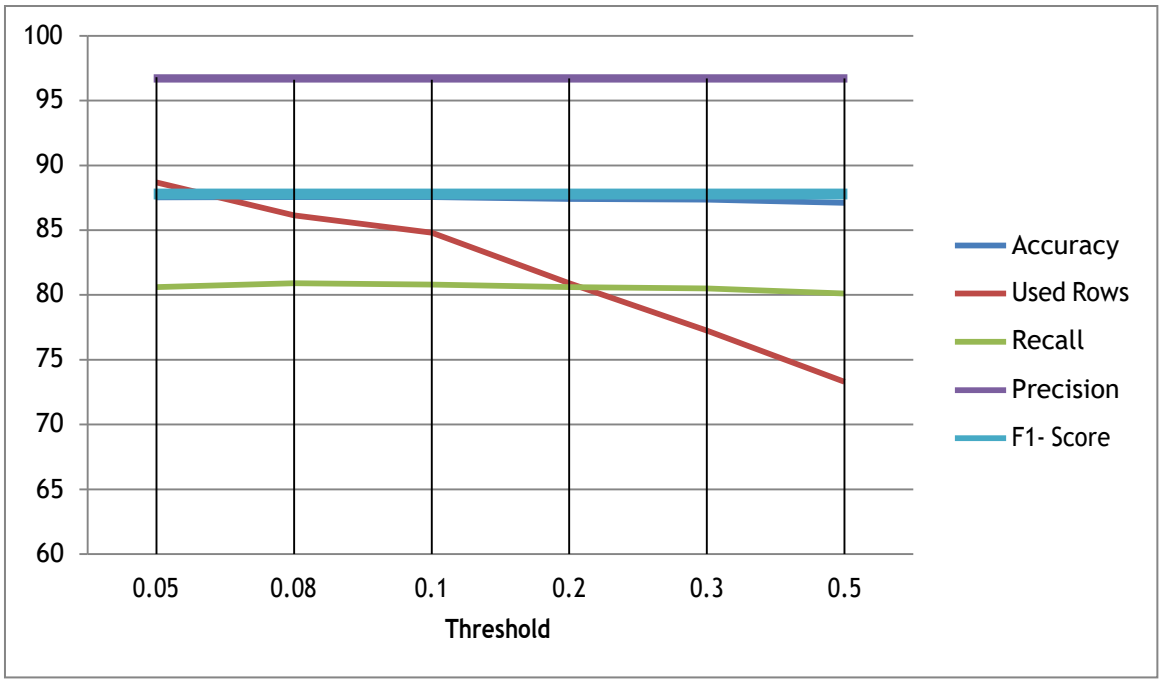

Figure 5.2: Represents Kyoto University Benchmark 2009 dataset performance parameters on the basis of threshold

It shows that as the threshold value increases the total rows of dataset is decreases with the no major change in Accuracy, Recall, Precision and F1- Score.

Result Analysis: NSL- KDD 2009 Dataset

The NSL-KDD dataset consists of 148517 unique connections and 41 features with 1 class label. The dataset divided into training and testing part so the training part i.e. 70% of NSL-KDD dataset consists of 103961 unique connections and 30% of remaining NSL-KDD dataset i.e. 44556 connections are for testing purpose.

Experiment 1: The result of full NSL-KDD 2009 dataset i.e. Accuracy, Recall, Precision, F1- score and time described in below table 5.6.

Table 5.6: SVM on NSL-KDD 2009 Dataset

Sr. No.	Performance parameter	Result (in %)
1.	Accuracy	97.33
2.	Recall	99.63
3.	Precision	95.40
4.	F1- Score	97.46
5.	Time	4644 sec

Experiment 2: Apply Infinite feature selection technique (INFFST) to select the relevant features from the dataset and after getting the relevant features applied SVM on selected features dataset and analyze its performance parameters. The selected features of NSL-KDD 2009 are shown in below table 5.7.

Table 5.7: Selected NSL-KDD 2009 features using INFFST

Dataset	Selected Feature
NSL-KDD 2009 Dataset	1,2,3,4,6,8,9,10,12,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,34,37,38,40,41

After selection of 33 features out of 41 features with 1 class label apply SVM to train the model. The performance parameters of SVM on selected features of NSL-KDD dataset is shown in table 5.8.

Table 5.8: SVM on NSL-KDD 2009 selected features

Sr. No.	Performance parameter	Result (in %)
1.	Accuracy	97.98
2.	Recall	98.80
3.	Precision	97.30
4.	F- Score	98.04
5.	Time	396.6 sec

Experiment 3: In this experiment apply horizontal reduction technique with using DBSCAN with five different threshold values on training part of INFFST applied dataset.

103961 unique connections with 33 features and 1 class label are divided into normal and attack label instances. 54080 are the normal connections and 49881 are the attack connections. DBSCAN forms clusters using threshold value and using horizontal reduced technique merge the similar connections for instance reduction. The performance parameters i.e. percentage of training row reduction, Accuracy, Recall, Precision, F1- score, Time and T – value were observed using SVM classifier. T– value is the t test value which is used to determine the independency between two samples. The obtained results are shown in table 5.9

Table 5.9: Horizontal abridging algorithm obtained results for NSL-KDD 2009 dataset

Sr. No.	Thres-hold	Training Reduced Rows (In %)	Accuracy (In %)	Recall (In %)	Precisi-on (In %)	F1-Score (In %)	T-Value	Time (In sec)
1.	1.0	1.48	97.93	98.6	97.3	97.9	0.026	383.72
2.	2.0	22.55	97.76	98.9	96.7	97.7	0.470	254.4
3.	3.0	31.76	97.11	99.4	95.1	97.2	0.680	207.53
4.	3.5	43.12	96.56	99.2	94.3	96.6	0.980	143.9
5.	4.0	44.66	96.36	99.2	94.0	96.5	1.023	136.47
6.	5.0	49.27	95.87	99.2	93.1	96.0	1.153	113.4

It was observed that the accuracy of full dataset is 97.33%. After the use of INFFST the accuracy is 97.98% by using less number of features. Horizontal Abridging Algorithm reduced the instances of dataset and obtained accuracy between 97.93% to % when the threshold value was between 1 to 5 respectively.

With the help of TOPSIS try to find the best threshold value which suggest the overall best parameters values i.e. Row reduction, Accuracy, Recall, Precision, F1-score and T- value. For the use of TOPSIS assigned weights are 2 for Training reduced rows, Accuracy and T- value; 1 for Recall, Precision, time and F1- score. It means that

Training reduced rows, Accuracy and T-value is twice time more important than Recall, Precision, time and F1-score. Obtained result for all the threshold value by using TOPSIS is shown in the table 5.10 below.

Table 5.10: TOPSIS result for NSL-KDD 2009 Dataset

Sr. No.	Threshold	TOPSIS Value
1.	1.0	0.4865
2.	2.0	0.5218
3.	3.0	0.5326
4.	3.5	0.5220
5.	4.0	0.5196
6.	5.0	0.5135

The maximum obtained TOPSIS value i.e. 0.5326 means that the best suited Threshold value is 3.0 for all the performance parameters for NSL-KDD 2009 dataset. So, even after 31.76% of rows reduction is achieve 97.11% accuracy.

Graphically represents the comparison between the original NSL-KDD 2009 dataset and horizontal abridging algorithm best obtained results according to TOPSIS.

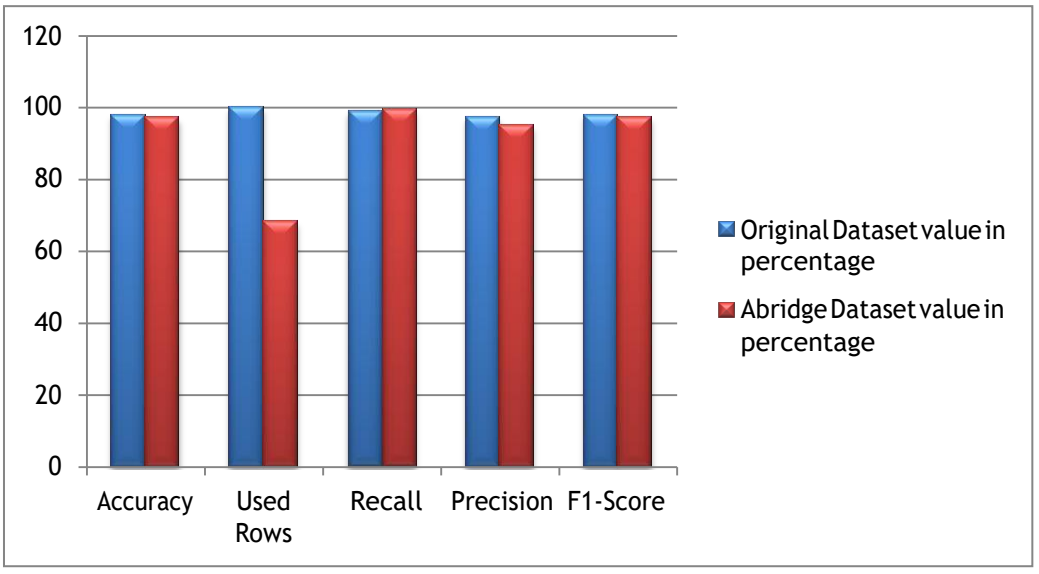

Figure 5.3: Original NSL-KDD 2009 Dataset versus Abridge Dataset value

Graphically shows in figure 5.3 that the Accuracy, recall, precision and F1-score are nearly equal in both the cases (Original NSL-KDD 2009 dataset and Abridge dataset) but the total rows reduction has major difference.

In figure 5.4 graphically visualize the increment of threshold value with performance parameters i.e. Accuracy, Total row reduction, Recall, Precision, F1-Score.

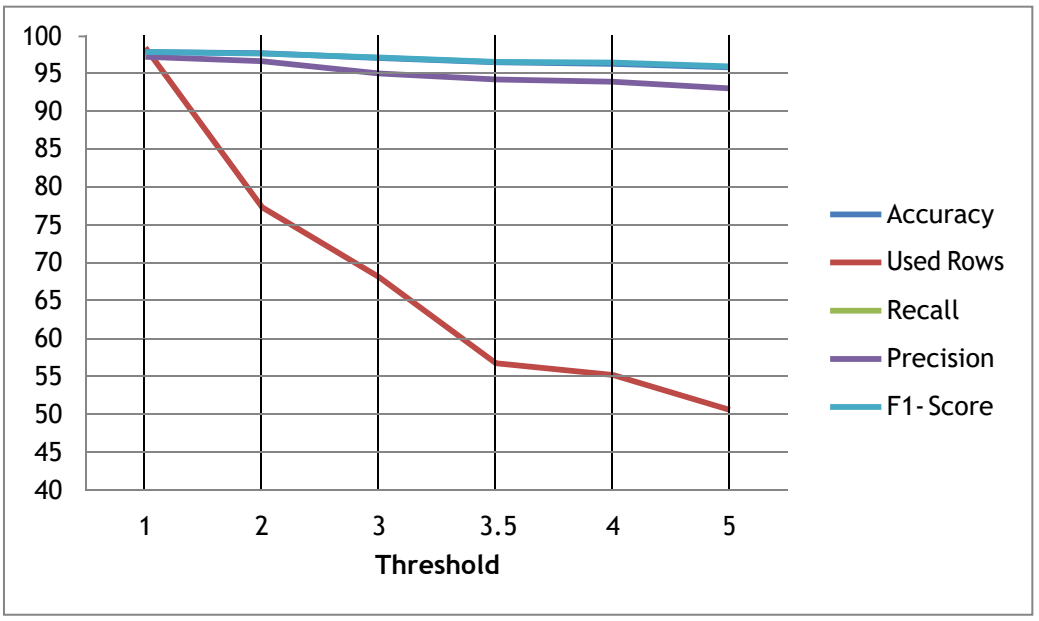

Figure 5.4: Represents NSL-KDD 2009 dataset performance parameters on the basis of threshold

It shows that as the threshold value increases the total rows of dataset is decreases with the no major change in Accuracy, Recall, Precision and F1- Score.

Obtained different threshold value results shows that as the threshold value increases the total number of rows reduction also increases without major change in accuracy, recall, precision, time, f1-score and t value. But, threshold value should not be increase till any extend for the reduction of rows because it also affects other parameters in a negative direction. Our main motive is to preserve all the performance parameter values with maximum percentage of rows reduction.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

Development and growth of internet increase day by day. Internet is doing many things which we did manually. Security is the major concern in the world of internet. Traditionally mainly encryption, firewall and other methods were used to secure the data, but now a day's intrusion detection system plays a major role in area of security or to detect the attack type. Two types of IDS are Signature based and Anomaly based IDS. In the proposed algorithm Infinite feature selection technique is used to reduce the size of attributes from the dataset by extracting relevant features. Proposed Horizontal abridge algorithm was used to combine the similar, redundant and duplicates instances to reduce the instances without affecting their characteristics. For the proposed algorithm Support vector machine classifier used as classification. TOPSIS is used to find one of the overall best calculated performance parameters i.e. total reduced rows, accuracy, recall, precision, time, f1- score and t value which depends on the threshold value used in DBSCAN algorithm. Two datasets were used to find the performance of proposed algorithm i.e. NSL- KDD 2009 and Kyoto University benchmark 2009 dataset.

- Horizontal reduction based abridging technique is proposed to reduce the number of instances without affecting the performance of IDS.
- Feature sets are reduced by 18.75% in Kyoto University dataset and by 19.51% in NSL-KDD dataset.
- The number of instances in training dataset is reduced by 13.86% and 31.76% in Kyoto University and NSL-KDD dataset respectively.
- The insignificant drop of 0.01% and 0.22% in terms of accuracy is noted in Kyoto University and NSL-KDD dataset respectively when using the proposed methodology.
- The positive significant drop in execution time of 38.70% in Kyoto University dataset and 95.53% in NSL-KDD dataset is achieved.

The conclusion of the proposed algorithm is that even after the reduction of features and instances of dataset till some extend, classifiers can get the same performance

parameter, so in place of using the whole dataset firstly try to reduce it and then trained your model for the better results within less amount of time.

The future scope of this research is that the proposed algorithm effectiveness can be tested by using different dataset other than NSL-KDD 2009 and Kyoto University Benchmark 2009 and in place of Infinite feature selection technique some other dimension reduction technique can be used to extract the relevant features from the dataset.

REFERENCES

[1] S. M. Bridges, “FUZZY DATA MINING AND GENETIC ALGORITHMS APPLIED TO INTRUSION DETECTION,” 2000.

[2] L. Portnoy, “Intrusion detection with unlabeled data using clustering”,

[3] A. Chittur, “Model Generation for an Intrusion Detection System Using Genetic Algorithms,” 2001.

[4] B. Balajinath and S. V Raghavan, “Intrusion detection through learning behavior model,” vol. 24, pp. 1202–1212, 2001.

[5] K. Labib and R. Vemuri, “NSOM: A Real-Time Network-Based Intrusion Detection System Using Self-Organizing Maps.”

[6] S. Chebrolu, A. Abraham, and J. P. Thomas, “Feature deduction and ensemble design of intrusion detection systems,” 2005.

[7] D. S. Kim, H. Nguyen, and J. S. Park, “Genetic Algorithm to Improve SVM Based Network Intrusion Detection System,” 2005.

[8] L. Khan, M. Awad, and B. Thuraisingham, “A new intrusion detection system using support vector machines,” pp. 507–521, 2007.

[9] C. Tsai and C. Lin, “A triangle area based nearest neighbors approach to intrusion detection,”

Pattern Recognit., vol. 43, no. 1, pp. 222–229, 2010.

[10] V. K. Pachghare, “Intrusion Detection System Using Self,” 2009.

[11] R. Nakkeeran, T. A. Albert, and R. Ezumalai, “Agent Based Efficient Anomaly Intrusion Detection System in Adhoc networks,” vol. 2, no. 1, pp. 52–56, 2010.

[12] H. Altwaijry and S. Algarny, “Bayesian based intrusion detection system,” pp. 1–6, 2012.

[13] S. Horng, M. Su, Y. Chen, T. Kao, R. Chen, and J. Lai, “Expert Systems with Applications A novel intrusion detection system based on hierarchical clustering and support vector machines,” Expert Syst. Appl., vol. 38, no. 1, pp. 306–313, 2011.

[14] H. Sedjelmaci and M. Feham, “N OVEL H YBRID I NTR USION D ETECTION S YSTEM,” vol. 3, no. 4, pp. 1–14, 2011.

[15] G. Gowrison, K. Ramar, K. Muneeswaran, and T. Revathi, “Minimal complexity attack classification intrusion detection system,” Appl. Soft Comput. J., vol. 13, no. 2, pp. 921–927, 2013.

[16] C. N. Modi, D. R. Patel, A. Patel, and M. Rajarajan, “Integrating Signature Apriori based Network Intrusion Detection System (NIDS) in Cloud Computing,” vol. 6, pp. 905–912, 2012.

[17] L. Dhanabal and S. P. Shantharajah, “A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms,” vol. 4, no. 6, pp. 446–452, 2015.

[18] R. Singh, H. Kumar, and R. K. Singla, “An intrusion detection system using network traffic profiling and online sequential extreme learning machine,” Expert Syst. Appl., vol. 42, no. 22, pp. 8609–8624, 2015.

[19] G. Roffo, S. Melzi, and M. Cristani, “Infinite Feature Selection.”

[20] A. Sayed, A. Aziz, S. E. Hanafi, and A. Ella, “Comparison of classification techniques applied for network intrusion detection and classification,” vol. 24, pp. 109–118, 2017.

[21] Akashdeep, I. Manzoor, and N. Kumar, “A feature reduced intrusion detection system using ANN classifier,” Expert Systems with Applications, vol. 88, pp. 249–257, 2017.

[22] E. Roszkowska, “MULTI-CRITERIA DECISION MAKING MODELS BY APPLYING THE TOPSIS METHOD TO CRISP,” no. Mcdm.

[23] M. Fahad, M. Sher, and Y. Bi, “Flow-based intrusion detection: Techniques and challenges,” vol. 70, pp. 238–254, 2017.

[24] M. A. Remli, K. M. Daud, and H. W. Nies, “K-Means Clustering with Infinite Feature Selection for Classification Tasks in Gene Expression Data K-Means Clustering with Infinite Feature Selection for Classification Tasks in Gene Expression Data,” no. June, 2017.

[25] E. Viegas, A. Santin, A. Bessani, and N. Neves, “BigFlow: Real-time and reliable anomaly-based intrusion detection for high-speed networks,” Futur. Gener. Comput. Syst., vol. 93, pp. 473–485, 2019.

[26] R. Vijayanand, D. Devaraj, and B. Kannapiran, “Intrusion detection system for wireless mesh network using multiple support vector machine classifiers with genetic-algorithm-based feature,” vol. 77, pp. 304–314, 2018.

[27] F. Salo, A. Bou, and A. Essex, “Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection,” Comput. Networks, vol. 148, pp. 164–175, 2019.

[28] A. Shenfield, D. Day, and A. Ayesh, “Intelligent intrusion detection systems using artificial neural networks,” vol. 4, pp. 95–99, 2018.

[29] M. Moradi and M. Zulkernine, “A Neural Network Based System for Intrusion Detection and Classification of Attacks.”

[30] M. Mazini, B. Shirazi, and I. Mahdavi, “Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms,” Journal of King Saud University - Computer and Information Sciences, 2018.

[31] B. Selvakumar, K. Muneeswaran, C. Science, M. Schlenk, and E. College, “Firefly algorithm based feature selection for network intrusion detection,” Comput. Secur., vol. 81, pp. 148–155, 2019.

PUBLICATIONS

1. Sheetal Garg, Raman Singh, V.K. Bhalla, “Statistical Abridging of Network Traffic Dataset for Intrusion Detection System”, Expert Systems with Applications, Elsevier. [Communicated]

PLAGIARISM REPORT

ME Thesis

ORIGINALITY REPORT

15%	8%	11%	10%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to De Montfort University Student Paper	2%
2	Submitted to King Saud University Student Paper	1%
3	www.ijcaonline.org Internet Source	1%
4	Submitted to Birla Institute of Technology Student Paper	1%
5	"Advances in Computing and Data Sciences", Springer Science and Business Media LLC, 2018 Publication	<1%
6	spectrum.library.concordia.ca Internet Source	<1%
7	"11th International Conference on Practical Applications of Computational Biology & Bioinformatics", Springer Nature, 2017 Publication	<1%
	Submitted to Institute of Technology	