B.Sc. in Computer Science and Engineering Thesis

A Correlation-based Feature Extraction method and intrusion detection framework for ML-based Intrusion Detection System (IDS)

Submitted by

Md. Mumtahin Habib Ullah Mazumder

160041066

Supervised by

Prof. Muhammad Mahbub Alam PhD

Department of Computer Science and Engineering

Islamic University of Technology

K B Bazar Road, 1704, Dhaka, Bangladesh

March 2021

CANDIDATES' DECLARATION

This is to certify that the work presented in this thesis, titled, "A Correlation-based Feature Extraction method and intrusion detection framework for ML-based Intrusion Detection System (IDS)", is the outcome of the investigation and research carried out by myself under the supervision of Prof. Muhammad Mahbub Alam PhD.

It is also declared that neither this thesis nor any part thereof has been submitted anywhere else for the award of any degree, diploma or other qualifications.

Md. Mumtahin Habib Ullah Mazumder
160041066

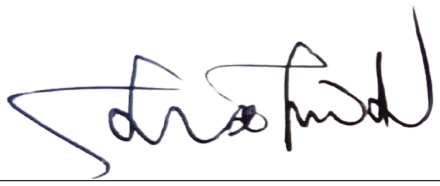
CERTIFICATION

This thesis titled, **"A Correlation-based Feature Extraction method and intrusion detection framework for ML-based Intrusion Detection System (IDS)"**, is the outcome of the investigation and research carried out by myself under the supervision of Dr. Muhammad Mahbub Alam as the requirements for the degree B.Sc. in Computer Science and Engineering in March 2021.

Author:

Md. Mumtahin Habib Ullah Mazumder

Supervisor:



Prof. Muhammad Mahbub Alam PhD
Professor
Department of Computer Science and Engineering
Islamic University of Technology

ACKNOWLEDGEMENT

I am indebted to my supervisor Prof. Muhammad Mahbub Alam, PhD, Professor, Department of Computer Science and Engineering, Islamic University of Technology (IUT), for being my adviser and mentor. His supervision, suggestions, knowledge and insights for this research have been invaluable. His valuable opinion in domain selection, subject finalization, algorithm proposition and implementation helped myself to make the endeavor into a complete research work.

Finally, I wish to express gratitude to my beloved parents for their love, support and encouragement. I am indebted to my parents for providing me the opportunities and experiences.

Dhaka
March 2021

Md. Mumtahin Habib Ullah Mazumder

Contents

List of Figures

List of Tables

ABSTRACT

The number of cyber-attacks has increased in recent years in both the number and varieties which demands a dynamic way of detection. Network Intrusion Detection System (IDS) leverages the key feature of Machine Learning algorithms to analyze network traffic and to build a sophisticated and dynamic system. However, the performance of Machine Learning algorithms depends on the representation of dataset.Recent research on Network Intrusion Detection has focused on feature selection and feature extraction techniques to obtain the best output and to adapt to continuously varying attacks. In this paper, we present a correlation-based technique for feature extraction from the traffic information. Our feature extraction framework builds a normal traffic profile and consider the deviation of network traffic information from normal traffic profile as the new feature set. The new derived set of features optimizes the anomaly detection technique using classification algorithm. Our evaluation conducted on KDD-CUP99,UNSW-NB15,NSL-KDD,AWID and CIC-IDS2017 dataset and outperformed detection rate for intrusions compared to other recent state-of-the-art anomaly detection methods.

Chapter 1

Introduction

1.1 Overview

With the development and increase of connectivity dependent devices in recent years, it has become a challenge for IT security professionals.Symantec Internet Security Threat Report claimed that more than three billion zero-day security attacks were reported on a single day in USA and Australia [1]. The dark web is also being a market place of such network intrusion mechanism and the interest in network intrusion activity is also in a rise there. A report found the rise of network intrusion posts on dark web by 69% in the first quarter of 2020 compared to the fourth quarter of 2019 [2]. In spite of development in cyber-security domain, it still a challenge to cope up with such new varieties of intrusion technique and attacks. To-address the alarming issue, extensive research is going on in Network Intrusion Detection System (IDS) nowadays.

As intrusion comprises any set of action that attempts to compromise availability, confidentiality, integrity or bypass security mechanism of a system, the approach of Intrusion Detection System is usually analyzing incoming network traffic in search of such attempts. For real time detection of such event, there are approaches like individual packet analysis, flow of traffic analysis and behavior of the source or attackers analysis [3]. The aim of Intrusion Detection System is to detect malicious activity, attack on computer security and confidentiality, spreading of computer virus, eavesdropping or stealing info from network activities and taking necessary action needed. An IDS works like the primary guard wall for network traffic.Typical IDS works based on a server which is set up on the links of a backbone network so that it can monitor all the traffic. It can even installed on smaller system of network traffic gateways like switch or router. Dedicated IDS can also be used to ensure security for individual server or connectivity dependent devices as well.In such a case it works by analyzing incoming traffic to the

system [4].

1.2 Intrusion Detection System (IDS)

Intrusion Detection System is commonly a reactive agent which examines the traffic and take action rather than being a pro-active agent.IDS works by monitoring network traffic, real-time examining the traffic and making a guard-wall against the attacks. An Application Protocol-based intrusion detection system is placed on a group of servers that are dedicated to coordinate the operation of an application and thus monitors and examines the traffic on application specific protocol.On the other hand, A Host-based Intrusion Detection System is installed in Individual devices participating in network.It analyzes all the events of that host devices like system calls, application and network logs, file system activities and system state [4].

There are three common methods of intrusion detection for IDS - Signature-based Detection, Anomaly-based Detection and Stateful Protocol Analysis. Hybrid method by combining these methods are also being used for intrusion detection [5].

1.2.1 Signature-based Detection (SD)

Signature in traffic commonly aims at searching for a pattern in the flow of traffic. SD analyze the captured network events and attempts to recognize possible intrusions. It is also called as knowledge based intrusion detection or misuse detection. Because it utilizes knowledge about the attacks or possible vulnerabilities of the system in search of footprint in network traffic flow. It works on comparative manner. SD is the simplest and effective method of intrusion detection which works based on contextual analysis [6].
SD also faces several issues. It seems ineffective with unknown attacks. It possess little understanding about internal working methodology of protocols or machine states. SD requires keeping signatures of attacks up to date. Such frequent knowledge management seems costly and time consuming.

1.2.2 Anomaly-Based Detection (AD)

Anomaly means the deviation from the 'normal traffic behavior'. It works in two stage - Generating normal traffic profile and Comparing with observed events. It characterize attacks as the dissimilar traffic to the normal traffic profile. It works effectively with new and unseen attacks. It is less dependent on context rather detects by privilege abuse [7].

Anomaly-based detection also faces some issues. It needs a lot of data to create traffic profile. It depends on quantitative manner more instead of qualitative approach.

1.2.3 Stateful Protocol Analysis

It works based on the knowledge of protocol standard for protocol operating in a network. This method considers unusual state of protocol as an intrusion and flags the traffic that causes such unusual state as potential intrusion. Stateful protocol analysis depends on vendor-developed generic profile and follows a qualitative approach to detect intrusion. This method traces protocol states and finds unusual traffic or commands. This method fails to detect attack which imitate like normal traffic [6].

1.3 Machine Learning based IDS

Machine Learning Algorithms comes with inherent property of deep statistical analysis of dataset. Intrusion Detection System leverages this key feature of Machine Learning to analyze network traffic info. Machine Learning based IDS works in anomaly based intrusion detection approach [8]. Anomaly based detection works like binary classification approach. AD builds a normal network traffic profile and identifies attack or intrusion traffic as discrepancy to normal traffic behavior. Machine Learning classification algorithms play the role of classifier in this case. Machine Learning provides a significant support for traffic analysis and profile generation as the inherent property of machine learning algorithm supports deep dive into the statistical analysis of data distribution. ML supported IDS is a quantitative process and self-learn the characteristics of traffic by discovering large amount of network traffic data.

However, the performance of Machine Learning model depends upon the amount of data is was introduced to, variation in dataset and the descriptiveness of data distribution in the samples of data [7]. The work of ML supported IDS becomes easy whenever the data distribution of 'normal' traffic is easy to characterise using simple mathematical model. But most of the real world system doesn't possess such distribution rather holds complex behavior. To support such data distribution, machine learning algorithms uses several deep analysis and optimization techniques. Machine Learning algorithms results in the characteristics of the traffic of the system from the observed data. Machine Learning based IDS performs in two steps - Feature Selection and Outlier Detection [8].

1.3.1 Feature Selection

Feature selection is the selection process of subset of relevant features to be used for model construction. Feature selection makes the model simplified and easier to interpret. In case of IDS, feature selection makes the best choice of data from the raw traffic info to take into consideration for the best performance of mathematical descriptor. Feature selection techniques helps to generalize the model and reduces overfitting. Feature Selection is a must in case of Big-data challenges as it can reduce computational demand. It also shorts time to train the model. There are three types of feature selection methodology is used commonly - Wrapper-based, Filter-based and Hybrid feature selection method [9].

Filter-based method utilizes independent algorithm or method to find out meaningful feature subset. It makes the selection by two common approaches: filter-based feature ranking and filter based subset evaluation. Filter-Based feature ranking method weighs the importance of each feature independently. It ranks the features based on the worthiness of the feature to make distinct decision about the sample. Ranking method is faster than the other method method as it considers individual features at a time. But the ranking method works poor on removing correlated features.It ranks the individual features based on self-distribution only.
Subset evaluation techniques works by making group of features. This method evaluates the descriptive property of the sub-group of features as well as the interrelation between the features participating in the sub-group by taking the mutual correlation into consideration. This method uses multi-variate measurement and compares between subsets of features to select the worthiest subset among those [10].

Wrapper-based method uses classifiers to make the choice of worthy subset of features. The subset evaluators of wrapper-based method also run search algorithm on the subset of features to find out the subset containing most descriptive info aiming at classification. But the approach is quite different. The selection and the evaluation is performed from the point of view of a learner. A learning algorithm like decision-tree is performed on different subset of available features and makes the best choice of subset. This method seems more realistic as the point of view of learner is taken into consideration. However, evaluation of the subsets through classifier makes the method extremely computationally expensive [11].

Hybrid-method combines both the ranking and wrapper method of feature selection. It takes the individual worthiness of features into consideration as well as importance in the sense of learner. The method attempts to combine the accuracy benefits of filter-based method with the computational efficiency of filter-based method [9].

1.3.2 Outlier Detection

Outlier is an observation point at distant from the ideal state or boundary. The intrusion detection is like a binary approach of normal versus attack. So the attack or malicious traffic is considered outlier and is identified by evaluating the nature of traffic. In this step, machine learning classifier comes into effect by utilizing the selected features. The classifier evaluates the characteristics of traffic flow and make the decision about the possibility of traffic belonging to an attack category. [8]

1.4 Feature Extraction method for IDS

Feature is commonly defined as the variables in representative from and derived from raw data. The distribution of characters or group of characters is also known as features. But while using the features for training a model, meaningful features only should be taken into account. Feature Selection method works in this motivation. However, to ease up the learning process more, analyzing and finding out inherent method among the given features, finding out the hidden statistical data distribution of the features and utilizing these extracted relation of the features can be beneficial. Such feature extraction or feature engineering approaches can help the classifier to learn the data distribution more clearly and thus helps to make accurate decision.Feature extraction method helps to limit storage requirement and faster learning by reducing amount of data to calculate. Moreover, the process reduces set of features to work with. Considering the relationship between features also helps in data understanding. Overall classification accuracy is also increased in this process as the features becomes distinct and more descriptible.
Feature extraction method comprises of two steps: Feature Construction and Feature Selection [12]. Feature construction step combines set of feature to produce a more meaningful, distinct and useful feature. Whereas feature selection works here as a second level of evaluator. Feature Selection has been described in brief in the previous section. So in this section will describe the feature construction method only.We present three popular feature construction methods: Association Rule Learning, Frequency Episode Extraction and n-grams Extraction [13].

1.4.1 Association Rule Learning

Association Rule is a famous method of data extraction in Data Mining. It finds out hidden relations between variables or features in a large dataset and the extracted relation can be utilized to train the model more efficiently [14]. It analyzes relation between events and tries to find out more likely event that will occur based on the evidence. In case of IDS, it considers the common characteristics of traffic or user behavior as evidence to determine a possible extracted feature which can describe the hidden property of those features.

1.4.2 Frequency Episode Extraction

Frequency episode represents the features as a sequential audit of data occurrences in the feature set. In fact, frequent episodes are the collection of events occurring together [15]. It helps in reducing dimension of feature set by considering combination of frequently occurring events. In case of IDS, the frequency episode extractor looks for sequential pattern in the features of given traffic. Such relation can describe temporal properties as well as statistical nature of data. So the relation can be further used as feature to make the classification more distinct.

1.4.3 N-grams Extraction

N-gram approach works like searching for a string matching of length n-characters [16].Many intrusion attempt exploiting vulnerabilities of a system can be identified by analyzing the header, monitoring the connection behavior and observing the session variables. But in case of N-gram extraction, the payload info is also taken into consideration. It attempts to find the payload pattern in packet inspection [13]. Such pattern is also being used as meaningful features.

1.5 Problem Statement

The objective of our work is to develop a feature extraction method and proposing a framework for Machine Learning based Intrusion Detection System classifier so that the classifier can be more efficient in case of detecting the traffic containing attack. As ML based IDS follows Anomaly Detection (AD) approach [8], the meaningful feature extraction can facilitate its classification accuracy between normal traffic and attacks. The existing dataset of IDS contains traffic information as features. Considering such large amount of traffic information as workable features can cause curse of dimensionality. On the other hand, extracting meaningful features from them can help to distinguish difference in data distribution between normal traffic and attacks.
We aim at thoroughly studying the existing approach of feature extraction for anomaly-based detection and the existing approaches of Machine Learning based Intrusion Detection System (IDS). We will make improvement with regard to benchmark metrics by introducing our proposed method. We will evaluate our proposed methodology on different available dataset of intrusion detection system. The problem statement can be described more specifically as:

”Inspecting and analyzing the incoming traffic info, how much efficiency of a Machine Learning based Intrusion Detection System (IDS) can achieve in detecting the potential attacks or security threat in the scale of Detection Rate (DR).”

1.6 Research Challenges

Dealing with network traffic data seems more challenging with respect to dealing with typical machine learning data. Most of the dataset is consist of only the header file info related to incoming traffic. To detect anomaly from the incoming traffic, it is more challenging to find out meaningful features to train the classifier.

Moreover, the intrusion detection system (IDS) is an anomaly-based detection classifier. So some previous attempt was to characterize an ideal profile of normal traffic which will help to identify the attacks as anomaly traffic deviated from the normal characteristics. However, the attempt fails when the attack can successfully imitate like normal-traffic behavior.

Most of the existing machine learning based IDS method doesn't take the qualitative analysis into consideration. So in the name of feature selection , it drops some important features of traffic which may be important traffic entity by convention of protocol. Such attempt of intrusion detection may result in good accuracy but loses explainability in real life scenario.

There are several intrusion detection dataset: KDD-CUP99, UNSW-NB15, NSL-KDD, CIC-IDS, CIC-DoS etc. These datset contains different data distribution for normal traffic as well as attacks. Most of the research methods aiming at only achieving accuracy which may performs good at particular dataset but fails in others. So there are a few generalized approach for all the dataset. Lack of generalization may seems inappropriate for intrusion detection in real life scenario. Generalized method for all the dataset is a research challenge.
To make a more meaningful and generalized intrusion detection system (IDS) is challenging.

1.7 Overview of Our Solution Approach

Our solution approach follows two typical steps of Anomaly-based detection - traffic profile generation and anomaly detection. However, in case of traffic profile generation, we have used correlation based grouping of features. The features that are mutually correlated will form a group. So the normal traffic profile will form groups of features characterizing the normal traffic behavior. Now the extracted features for training will be deviation value of all the traffic in training set from the groups of 'normal traffic profile'.Such extracted data can identify possible deviation of normal traffic from 'ideal characteristics' as well as the range of deviation for attack from ideal normal traffic profile. The extracted data will work as training set for the machine learning classifier.

In our solution approach, we have followed traditional Anomaly-based Detection method for Intrusion Detection. However, our attempt was to develop a more generalized framework. Moreover, we attempted to keep approach of classification more meaningful. As most of the methods in literature drops meaningful features in feature selection and thus loss qualitative strength, we have followed feature extraction method instead of feature selection. So that our method doesn't drop away features, rather consider interrelation between them for training classifier. Such an approach can ease up the training process by taking hidden properties into account as well as it reduces no of features to work with which releases from 'curse of dimensionality'.

1.8 Research Goals

- Propose a novel feature extraction method for Network Traffic dataset.
- Propose a framework of Intrusion Detection System.
- Exploring the effectiveness and justifying the approach.
- Study the effect of different parameters on the Detection Rate (DR) of Intrusion Detection System (IDS) approach.
- Try to formulate a better and efficient approach in respect of Detection Rate (DR) compared to the methods in Literature.

1.9 Thesis Outline

In Chapter 1 we have discussed our study about the domain of working in descriptive manner. Chapter 2 contains the necessary background study and literature review for our work and the contribution of these paper. In Chapter 3 we have stated our proposed method in detail with proper explanation and figures to provide detailed insight.In Chapter 4 we have shown the result analysis of our proposed model and comparative analysis with the existing work in literature. Chapter 5 draws the conclusion of our thesis work with stating the future work. The final segment of this book contains all the references.

Chapter 2

Literature Review

2.1 Statistics-based Techniques

The statistics-based techniques utilizes statistics techniques to support anomaly-detection model. The common features of network intrusion system is the header information containing by a network traffic packet or flow information of data traffic. Statistics-based method inspects network traffic by individual packet and detects potential intrusion. Being an anomaly-detection model, statistics-based approach builds a normal traffic profile based on statistics-based data distribution model. The statistics-based IDS calculates different statistical properties like mean, standard deviation, min, max, median etc from the features.The normal traffic profile is created based on the extracted pattern of statistical values and distribution. Any packet information deviates from the normal traffic profile, is marked as anomaly. Statistics-based techniques provides a real-time inspection by evaluating each incoming data packet based on normal traffic behavior. The three common types of statistics-based intrusion detection techniques are: Univariate Technique, Multivariate Technique and Time Series Model.

2.1.1 Univariate Technique

The univariate technique takes the statistical property of each features into account individually. It calculates some statistical properties on each of the features or the complete set of features. The characteristics of the whole dataset or individual features is taken into account but the mutual correlation between the features is ignored in this approach.

Univariate technique approach considers the extracted statistical properties as behavior metrics. The characteristics of incoming unknown traffic is evaluated based on the behavior metrics of normal traffic. Univariate technique evaluate individual traffic packets for each of the behavior

metric and based on the result, the Univariate Intrusion Detection System detects abnormalities [17].

2.1.2 Multivariate Technique

The multivariate-technique for intrusion detection system is also an anomaly detection techniques which leverages relation between the features or variables to build normal traffic profile. Multivariate approach takes into account the combination of features. In general approach, the correlation value of feature pairs is used as the measurement of mutual relationship of features. However, several statistical measurement techniques are also used for measuring the relationship. The relationship among the variables provides a second order representation of the normal traffic behavior. So utilizing the relationship information in normal traffic profile generation can add another layer on the normal traffic behavior.

Ye et al. presented multivariate correlation technique to create long term normal behavior profile and detecting anomaly based on the profile. This approach utilized the motivation of quality control method where long term static-behavior of expected traffic is made by statistical analysis of the mutual combination of ideal traffic and any deviation from the expected behavior is marked as abnormalities [17]. However, different multivariate techniques have been taken into account to identify the mutual relationship more precisely. Yeung et al. introduced Covariance-matrix based profile modeling for normal traffic and examined their method on flooding attacks. The covariance-matrix based modeling method showed how multivariate anomaly detection can be used for the detection of flow-based DoS attacks as well [18]. Tan et al. proposed multiplication based multivariate correlation technique and intrusion detection framework for Denial-of-Service attack. The mathematical product of all feature pairs is considered as the feature correlation in this method. Moreover, a distance-based classifier using mahalanobis distance metric with normal traffic is proposed [19]. Li et al. extended the work and examined addition based correlation on the framework. Addition based correlation takes mathematical addition instead of product among the feature pairs [20]. However, multivariate correlation technique faces the challenge of curse of dimensionality as the number of features increases extensively. The increase in number of features causes the estimation of statistical distribution difficult [21]. Dimensionality reduction techniques such as feature selection is used to meet the challenge. Gottwalt et al. proposed a τ(tau)-correlation based feature selection technique on extracted multivariate features [22]. Our proposed method also contains a novel multivariate feature modeling techniques. So, we have presented the technique of some related papers in this section in detail.

Covariance-Matrix Modeling and Detecting Various Flooding Attacks [18]:

This paper introduced the novel method of using co-variance of the features of normal traffic and utilizing correlation of features to detect anomaly detection. The authors discussed about the importance of correlation in building normal traffic profile. This paper showed that the deviation in the covariance of features from normal traffic profile can quantify the anomaly in unknown traffic packets.

The proposed method can detect flooding/DoS attacks.The traffic flow information of **X** temporal sample stream is divided into **l** chunks containing n number of samples each.(fig-2.1)

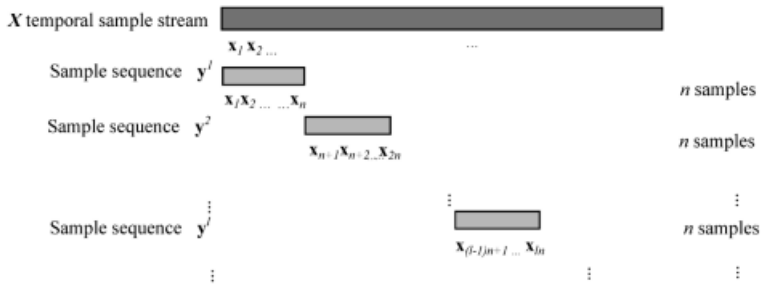


Figure 2.1: Segmentation of observed temporal sample stream

For any chunk-**l**, a matrix representation of the samples y^l is considered. The chunk contains n number of samples, where each of the samples having p features, as fig-2.2

$$\mathbf{y}^l = \begin{pmatrix} f_1^{l,1} & f_2^{l,1} & \cdots & f_p^{l,1} \\ f_1^{l,2} & f_2^{l,2} & \cdots & f_p^{l,2} \\ \vdots & \vdots & \ddots & \vdots \\ f_1^{l,n} & f_2^{l,n} & \cdots & f_p^{l,n} \end{pmatrix}$$

Figure 2.2: Matrix representation of chunk - l

For the chunk-**l**, a co-variance matrix is formed on the p-features which represents the correlation between the feature pairs participating in the respective chunk(fig-2.3).

The data dimension is shifted from a p-feature space to a p(p+1)/2 feature space because of the formulation of covariance matrix representation. The detection methodology is modeled on the dissimilarity between matrices. For an unknown traffic flow, the respective covariance matrix for the chunk containing the traffic flow is generated. Comparing the covariance matrix of unknown traffic flow with trained covariance matrix representation of normal traffic flow

$$\mathbf{M}^l = \begin{pmatrix} \sigma_{f_1^l,f_1^l} & \sigma_{f_1^l,f_2^l} & \cdots & \sigma_{f_1^l,f_p^l} \\ \sigma_{f_2^l,f_1^l} & \sigma_{f_2^l,f_2^l} & \cdots & \sigma_{f_2^l,f_p^l} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{f_p^l,f_1^l} & \sigma_{f_p^l,f_2^l} & \cdots & \sigma_{f_p^l,f_p^l} \end{pmatrix}$$

Figure 2.3: Covariance matrix of the features for chunk-l

using a distance based function, the anomalous traffic flow is identified. The methodology is shown in fig-2.4.

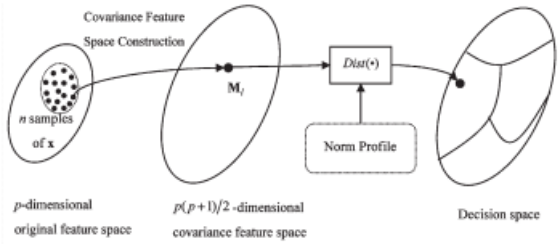


Figure 2.4: Illustration of the covariance-matrix-based detection model

The distance function is a binary decision function which compares the values of covariance of normal traffic profile and covariance of observed incoming traffic flow. M^obs is the covariance-matrix containing the record of a chunk of sequence of observed data where **N** is the matrix of normal traffic profile and **T** is the threshold of dissimilarity. For each of the element in covariance matrix of observed traffic and normal traffic,the difference is calculated using the dissimilarity function (fig-2.5) and based on the dissimilarity threshold. A binary matrix is formed as a result of dissimilarity function. The observed traffic is marked as normal for the formation a zero matrix and anomaly for a non-zero matrix.

$$\mathrm{Dist}(\mathbf{M}^{\mathrm{obs}}, \mathbf{N}; \mathbf{T}) = (d_{u\nu})_{p\times p}$$
$$\forall m_{u\nu}^{\mathrm{obs}} \in \mathbf{M}^{\mathrm{obs}} \quad \forall n_{u\nu} \in \mathbf{N} \quad \forall \delta_{u\nu} \in \mathrm{T}$$
$$d_{u\nu} = \begin{cases} 1, & \text{if} \left| m_{u\nu}^{\mathrm{obs}} - n_{u\nu} \right| \geq \delta_{u\nu} \\ 0, & \text{if} \left| m_{u\nu}^{\mathrm{obs}} - n_{u\nu} \right| < \delta_{u\nu} \end{cases}$$

Figure 2.5: Dissimilarity Function

A System for Denial-of-Service Attack Detection Based on Multivariate Correlation Analysis [19]

The authors proposed a multiplication based multivariate correlation technique and a 'mahalanobis distance' based detection framework.In the multiplication based correlation method, the mathematical product of all possible feature pair is considered as multivariate correlation

based extracted feature. The method is dynamic in nature as it is considering the product of two features instead of individual features. Any changes in individual feature is interpreted in mutual representation. So,updating the normal traffic profile is dynamic and the detection of anomaly is real-time.

The normal traffic behavior is measured by calculating the mean and covariance for ideal normal traffic. The possible deviation of real world normal traffic from the ideal profile is calculated as standard deviation. Now, any traffic obtains the distance within the ideal normal traffic behavior maintaining standard deviation is considered as 'Normal' traffic. Whereas, the traffic out of expected boundary is marked as 'Anomaly'.

An Intrusion Detection System Based on Polynomial Feature Correlation Analysis [20]

This paper proposed another multivariate correlation technique named Addition-based Correlation (ABC) and extends the framework proposed by Tan et al [19]. In case of multivariate correlation of feature, the mathematical addition of feature pairs is considered in this method. The normal traffic profile is created on the extracted multivariate features and the anomaly is detected based on the deviation from normal traffic profile. The method is consist of three step: feature extraction process using Addition Based Correlation (ABC) method, creating Normal Traffic Profile and detecting the anomaly using Distance-based Classifier.

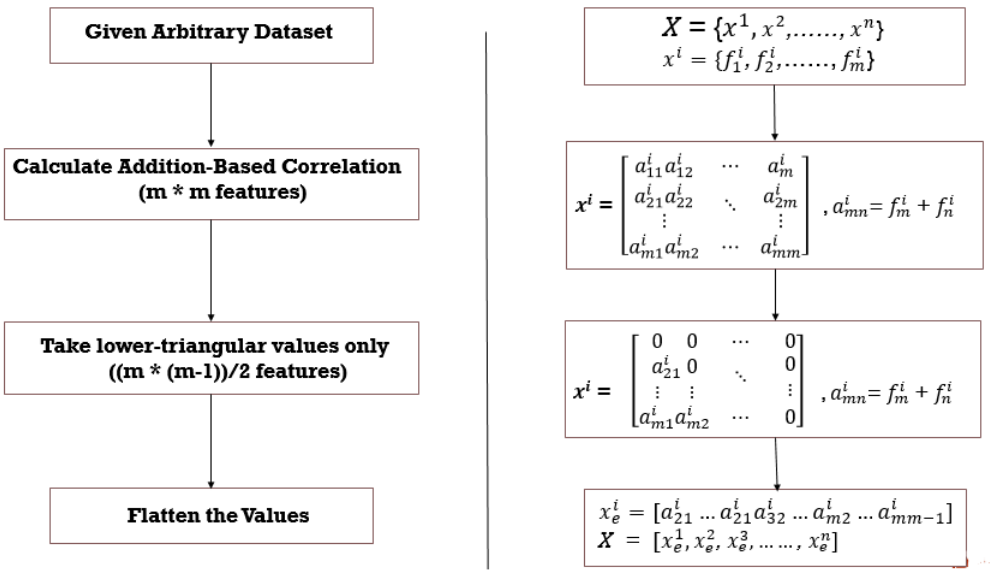


Figure 2.6: Feature Extraction process of ABC-method

For feature multivariate feature extraction step (fig-2.6),a novel feature extraction method is introduced known Addition Based Correlation(ABC).In the ABC method, mathematical addition

between all the feature pairs is calculated. The new extracted multivariate feature is the sum of all possible feature pairs.

Given a traditional dataset of m features, for the combination of all possible feature pair, the Addition-based Correlation is calculated. The features of extracted feature set is the sum of feature pairs of original dataset. The original dataset of m feature results in an extracted feature set of $\frac{m(m-1)}{2}$ multivariate correlated features.

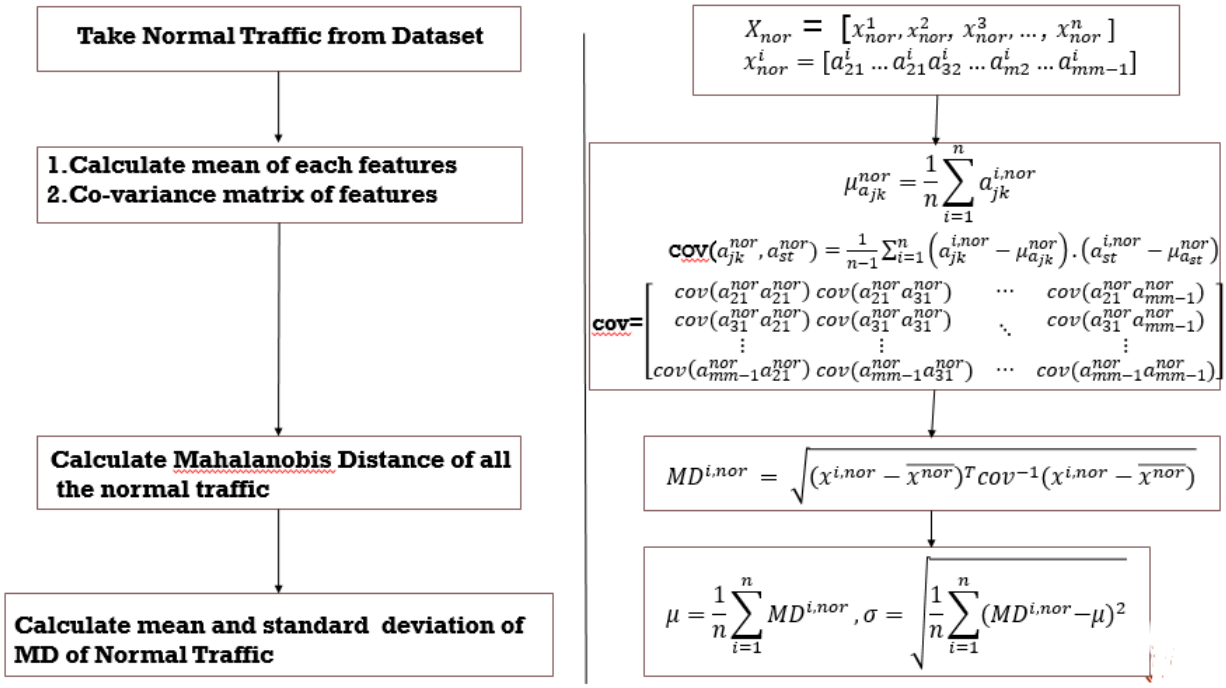


Figure 2.7: Creating Normal Traffic Profile

For Normal Traffic Profile generation step (fig-2.7), the mean and covariance matrix for the multivariate correlated features. To estimate the normal traffic behavior, 'Mahalanobis Distance' for all normal traffic in train set from the expected value (mean) of normal traffic is obtained. Finally, the expected mahalanobis distance and possible deviation from the mean is measured. The calculated mean and standard deviation of mahalanobis distance of normal traffic is considered as the normal traffic behavior profile.

$$MD^{New} = \sqrt{(x^{New} - \overline{x^{nor}})^T cov^{-1}(x^{New} - \overline{x^{nor}})}$$

Figure 2.8: Mahalanobis Distance based Classifier

In the final step, a mahalanobis distance-based classifier(fig-2.8) is used. The expected range of Mahalanobis distance from the mean of features of normal is defined by the mean and standard

deviation of normal traffic($\mu - \sigma$ to $\mu + \sigma$). For an incoming traffic, the distance from normal traffic mean is calculated. In case of the calculated Mahalanobis Distance value of unknown traffic is in the range of expected Mahalanobis Distance value of normal traffic maintaining the boundary of standard deviation, the traffic packet is marked as 'Normal'. Otherwise, the traffic is considered as 'Anomaly'.

CorrCorr: A Feature Selection Method for Multivariate Correlation Network Anomaly Detection Techniques [22]

The multivariate correlation techniques suffers from curse-of-dimensionality as the number of features increases extensively. Working with the increased number of features can cause additional complexity in building normal traffic profile [21]. Gottwalt et al. proposed a feature selection method for dimensionality reduction for the increased number of feature space.

The authors used a symmetrical τ(tau) correlation based feature selection method [23] on the extracted multivariate features to select the reduced feature set. The symmetrical τ(tau) correlation method is a filter-based feature selection method. The ranking of the multivariate features is generated not only by the descriptive information it contains but also taking the mutual correlation and inter-dependency into consideration. From the ranking of feature, set of most relevant features is obtained. Moreover, as the symmetrical τ(tau) correlation method takes the mutual correlation of the features into consideration, a second layer of correlation information between the features is obtained whereas the first layer was the technique of multivariate correlation. More descriptive normal traffic profile can be obtained using the second-order correlation framework along with existing statistical analysis.

2.1.3 Time Series Model

Time series is a sequence of observations made over a certain time interval. Time series model analyzes traffic flow or a number of time-varying sequential packets and performs statistical analysis on the flow to detect abnormalities. Time series model works on aggregated packets over a period of time. A certain flow of normal packets may possess 'strong similarities' in behavior whereas an intrusion or anomaly can brings dissimilarity to the flow.

Time series approach of intrusion detection system (IDS) performs based on probability of occurring of an incoming traffic in a particular time. Viinikka et al. proposed a method where network flow are aggregated to generate a flow characteristics. The extracted data representation of normal traffic possess 'strong similarity' in behavior whereas attacks causes inconsistency. Moreover, aggregated traffic analysis helps to identify interesting phenomena about relevancy

of normal traffic and attacks [24]. Qingtao et al. presented a model where time series model mimics the anomaly-detection approach. Abnormalities are detected by observing abrupt variation in time series data of traffic flow [25].

2.2 Knowledge-based Techniques

Knowledge based approach is a qualitative anomaly detection technique. Unlike the other anomaly-detection techniques, the knowledge-based IDS is created on a set of rules defines normal network activity. The knowledge-based techniques is based on standard rules and a new traffic is evaluated on the rules. The knowledge-based IDS faces low false-positive ratio. However, this approach lacks of extensive data analysis and significant statistical extraction of information like traditional quantitative approaches.The common types of statistics-based intrusion detection techniques are: Finite State Machine, Description Language, Expert System and Signature Analysis.

2.2.1 Finite State Machine

A finite state machine-based model is a representation of normal traffic behavior in possible sequence of execution flow. The control operations, behavior traffic activities and resultant set of machine variables for normal traffic is mapped to a finite state machine with states and transitions. Any variation from expected transitions is marked as anomalous behavior. Walikinshaw et al. proposed a FSM model where possible states and transitions represents legitimate system behavior and detected unexpected transitions in FSM is marked as anomaly [26].

2.2.2 Description Language

Description Language-based approach defines the syntax possible set of rules where the rules specifies the behavior and signature of known attacks. Studnia et al proposed an intrusion detection approach leveraging language theory to represent attack-signature from the behavioral pattern of malicious traffic [27].

2.2.3 Expert System

An expert system is represented by a number of manually defined rules which contains the known attack behavior of traffic. Kim et al. proposed a qualitative approach of hierarchical misuse detection model designed based on defined set of rules by the domain experts [28].

2.2.4 Signature Analysis System

The signature analysis approach is the earliest one in the field of Intrusion Detection System (IDS). It works like traditional string matching or pattern matching approach. A signature database of known attacks stores the signature of all the known attacks. The signature-based IDS inspect all the incoming packets in real time for possible string, word or pattern matching with the known attacks. Kenkre et al. proposed a framework using IPS open source tool to inspect incoming network traffic and logging suspicious packet information [29].

2.3 Machine Learning Techniques

Machine Learning algorithms has brought an immense change in quantitative data analysis. The inherent property of machine learning algorithms is used to identify the data distribution from a given set of data. ML methods performs a complex pattern-matching calculation to extract pattern and complex relationship in dataset. Machine Learning techniques have been used extensively for intrusion detection system to discover knowledge from intrusion dataset [30].

2.3.1 Supervised Learning Methods

A supervised learning method learns the characteristics of traffic behavior from a train set with labelled data and evaluated on test dataset. The trained model is used to detect anomalous traffic.

Decision Tree: Decision tree converts a intrusion dataset into decision node and branch representation. As IDS system faces curse-of-dimensionality issue because of considering raw traffic information as features, Decision Tree based approach performs extensively well through selection of meaningful features [31]. Thaseen et al. proposed intrusion detection system based on random-tree model which reduces false alarm rate of existing classifiers [32].Khraisat et al. examined c5 decision tree based classifier to reduce both the false positive and false negative rate [33].

Naive Bayes: This methods works based on bayes principle and dependency assumption among the attributes. Koc et al. experimented naive bayes based method for intrusion detection system and found naive-bayes to be extremely helpful for high-dimensionality issue. Moreover, Hidden Naive Bayes model achieves a high speed network by utilizing the property of highly interdependent attributes. [34].

Genetic Algorithms: Genetic algorithm in Intrusion Detection System (IDS) is used by defining properties of network intrusion dataset as genome and population. Hoque et al. defined feature characteristics as genome and number of random rules as population [35].

Fuzzy Logic: Fuzzy logic based classifier considers degree of uncertainty rather than traditional numerical value analysis approach only. As the features of intrusion dataset is the header and flow information of network traffic, the features aren't equally meaningful and suffers from vagueness. Elhag et al. used fuzzy logic to handle such ambiguous data [36].

Support Vector Machine (SVM): SVM usually draws a splitting hyperplane between normal traffic and attacks in intrusion detection system. Intrusion dataset often suffers from high-dimensionality whereas SVM can utilize the high-dimensionality nature of IDS dataset. Li et al. leveraged SVM with a feature selection technique for intrusion detection system [37].

Hidden Markov Model: HMM is used in IDS by modeling HMM model with known data characteristics. Using markov model characteristics, the unseen data can be identified [38].

K-Nearest Neighbour: KNN is a classical approach for anomaly detection. Lin et al. used Nearest Neighbour approach with combining the center of clusters for profile definition of network traffic [39].

2.3.2 Unsupervised and Semi-supervised Learning Methods

Unsupervised Learning methods extract interesting properties from the input dataset without class labels.

K-means clustering: K-means clustering technique is used to extract out group similar properties of data which can be leveraged in group similar intrusion detection system modeling. Annachhatre et.al. used k-means clustering to identify host behavior [38].

Semi-supervised learning can be used in case of occasional labelled data [40]. Different semi-supervised techniques such as co-training [41], self-training [42], Expectation maximization [43] and Graph-based method [44] are used.

2.3.3 Ensemble Methods

Multiple machine learning algorithms is used together to achieve a better performance. Aburomman et al. utilized ensemble method for modeling meta-classifier in two step learning [45]. Jabbar et al. proposed ensemble method of Random Forest and AODE-based naive bayes classifier model to leverage enhanced precision and attribute dependency respectively [46]. Gaikwad et. al. proposed a bagging ensemble method using REPTree and achieved low false positive rate [47]. Moustafa et. al. proposed a an AdaBoost ensemble method combining Decision Tree, Naive Bayes and Artificial Neural network to achieve a high detection rate [48]. Paulauskas et. al. combined four weaker learners: Naive Bayes,Partial Decision List (PART), J48, C5.0 and evaluated their increase of performance in ensemble method [49]. Zhou et al. examined with the combination of C4.5, Random Forest and Forest by Penalizing Attributes (Forest PA) which exhibits better performance than state-of-the-art approaches under several metrics [50].

2.3.4 Feature Selection Methods

As network traffic faces problem of extensive features, feature selection methods is used to solve curse-of-dimensionality.Three common feature selection methodologies are- Wrapper-based, Filter-based and Hybrid feature selection method [9].

Filter-based method uses multi-variate measurement and compares between subsets of features to select the worthiest subset among those features [10].Abdullah et al. proposed a method where a dataset is divided into subsets according to each attack and feature selection is performed based on Information Gain (IG) of each feature in all of the subset [51].Hota et. al. examined four feature selection method: Info Gain, Correlation, ReliefF, Symmetrical Uncertainty and showed a comparative analysis [52].Zhou et al. used a hybrid selection method combining CFS and Bat-Algorithm [50].
Wrapper-based method uses classifiers to make the choice of worthy subset of features through an extremely computationally expensive process [11].Khammassi et. al. proposed a wrapper method using Genetic Algorithm and Logistic Regression [53].Pajouh et. al proposed another wrapper method principle component analysis and linear discriminate analysis [54].
Hybrid method combines the accuracy benefits of filter-based method with the computational efficiency of filter-based method [9].Several hybrid method such as central point of attributes with ARM [55] is also used.

A hybrid feature selection for network intrusion detection systems: Central points [55]

The authors proposed an adaptive feature selection method for intrusion detection system.The proposed method(figure-2.9) contains two algorithms which works as the two principle step of the method.

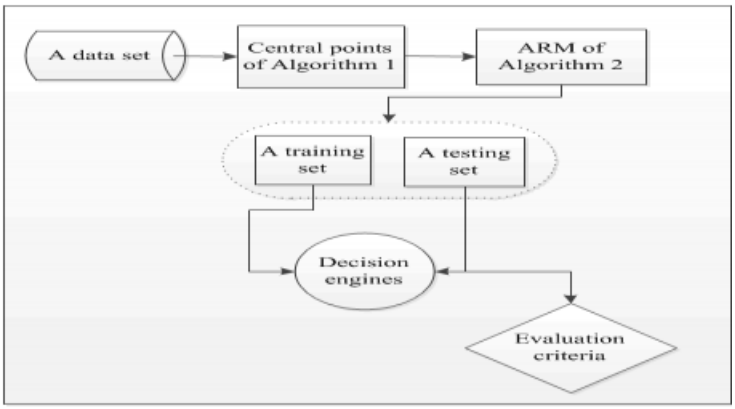


Figure 2.9: The Proposed Architecture for an Adaptive NIDS

In the first algorithm, 'Central Points of Attribute values' is calculated. By the term central point, it refers to the most frequent occurring value of an attribute. For the calculation of central points, the dataset is splitted into partitions and the central points for each of the partitions is calculated. The number of the partitions is denoted by p where,

$$p = \#ofpartitions = \frac{\#ofrecords}{\#ofattributes}$$

The second step of the method is Association Rule Mining [14] based feature selection.The algorithm discards away the similar features containing same central point and works with the independent features only. The association rule mining is applied on the selected independent features. Association rule mining approach creates a antecedent and precedent relationship among the features such that, for r = $\{f_1, f_2, f_3, ..., f_N\}$, the relationship is $f_1(antecedent) \rightarrow f_2(precedent)$ where,
1) $f_1 \subseteq r, f_2 \subseteq r$ and 2) $f_1 \cap f_2 \epsilon \emptyset$
This method is a two-step hybrid feature selection approach using the central point calculation of attributes and Association Rule Mining (ARM).

2.3.5 Deep Learning Methods

Deep Learning methods provide flexibility and ability to discover complex patterns from intrusion dataset. Moreover, deep learning methods possessing the ability to handle large amount

of data, is able to handle abounding number of samples extracted from raw data flow information in intrusion dataset. Deep Learning can also take multi-dimensional data as input. So, a two-dimensional representation for flow consisting of consecutive packets is introduced to detect Distributed Denial of Service attack [56]. Moreover, temporal data processing capability of RNN-LSTM network has been utilized [57]. Some of the related paper achieved benchmark result using Deep Learning approach is discussed in this section in detail.

DeepDefense: Identifying DDoS Attack via Deep Learning [56]

This paper have leveraged the RNN-LSTM architecture for DoS attack detection. The authors have introduced sequential multi-dimensional representation of successive network traffic.
In the first step, the method performs data pre-processing where categorical features are replaced by nominal values and the long strings feature values of ICSX dataset is processed by bag of words. After preprocessing, the dataset becomes of shape (m x n') where n' is the number of features after transformation. Then T successive traffic is combined into a single multi-dimensional feature input of size T x n' where the label of two-dimensional input stream is the label of the last traffic packet sample. This new feature set contains the successive traffic history which brings multi-dimensional sequential representation to the traffic records.

The primary dataset of m number of samples and n features (mxn shape) is processed to a dataset of shape (T-m)xTxn' where each of the sample contains the description of itself along with previous (T-1) records. Then the data is feed to a RNN-LSTM neural network which learns the sequential nature between the successive traffic. Finally a new traffic with history of (T-1) number of previous traffic samples, can be evaluated using the learned model.

Edge-Detect: Edge-centric Network Intrusion Detection using Deep Neural Network [57]

This paper has follows deep neural based LSTM-RNN network introduced by Yuan et al. [56] for intrusion detection on UNSW-NB15 dataset. The authors designed this method based on the basic characteristic of DoS attack, 'Frequent data packet incoming to an edge can cause DDoS attack'. So the method aims at detecting edge under attack by assessing sequential scanning of incoming traffic. A data science pipeline is proposed for sequential assessing of incoming data packet to scan and detect 'edge under attack'.

Two stage of learning is used in this method. In stage-1,feature selection and feature reduction is performed by using an sliding window of size T . The feature selection method is same as the paper by Moustafa et al. [55] and worked with only 11 features of UNSW-NB15 dataset selected by the paper [55]. From the **n** no of features, **n'** features is selected in feature selection

step. The sliding window of size T contains the features of T'th traffic information along with previous T-1 traffic record. Each of the traffic sample is converted to 2-D feature set containing information of previous T traffics and n' feature each. At the end of pre-processing stage, from the initial size of (m×n) where in total of m sample of traffic each of having n features, is converted to (m T + 1) number of windows samples where each of the sample is of 2-D shape having a size (T × n').
In stage-2, the neural network in effect is created. This stage is designed on GRU is a variant of LSTM. It leverages the advantage of LSTM to remember the value of the variables in sequential traffic flow to have a deep analysis about an attempt to generate a DDoS attack. These LSTM layers are followed by a dense layer of 128 cells and finally the output layer is flattened and result in possibility of being a DDoS atack. The activation functions used are 'tanh' for LSTM, 'ReLU' for dense layer and 'sigmoid' for output layer in the proposed model.

This method shows an extremely good accuracy in detecting DDoS attack by performing sequential analysis of traffic. However, the method is not applicable for other types of attacks as all other attacks has to be detected by statistical inspection of header information of the traffic flow rather than the sequential flow of packets. Moreover, this method consider feature named 'destination port' which ease up detecting the dos attack irrespective of the information of traffic.

MSTREAM: Fast Anomaly Detection in Multi-Aspect Streams [58]

From a incoming stream of data packet, this method detects the DoS attack analyzing the traffic packet through hashing of features information and observing overlapping of similar information of incoming stream of traffic. The method is called as 'group-anomaly'. A flow of stream to be marked as 'group-anomaly' should possess three properties: 1) Similarity in categorical attributes of the stream of traffic, 2) similarity in real-valued attributes of the flow of traffic, 3) arriving these flow suddenly over a short period of time. The working methodology of MStream can be divided into two step: Hashing of attributes (both categorical and real-valued) and temporal scoring. After scoring the each of traffic record, it decides about whether the traffic is attempt to DoS attack or not.

The diagram of proposed method is given in Fig-2.10. At first dimensionality reduction is performed. In this approach, the authors have used three different unsupervised dimensionality reduction approaches: Principle Component Analysis (PCA), Information Bottleneck (IB) and Autoencoder (AE). The next step is hashing. In case of hashing, two different methods of hashing is proposed: **FeatureHash** and **RecordHash**. For hashing, a 'b-bucket' method is used where the feature information of the records are hashed to b number of integers.

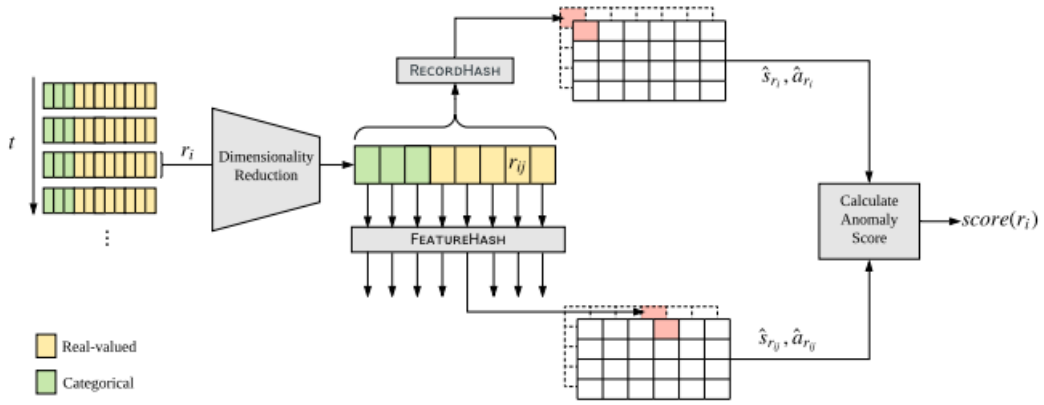


Figure 2.10: Diagram of Proposed Mstream

In Feature Hash Approach, hashing on individual features of given record is performed and stored in the respective bucket. In Record Hash approach, the entire record is divided into categorical and real-valued feature and hashing is performed on the two portions. Then the remainder of sum of these two portion by bucket number is taken as hashed value. Finally both the hash value is passed through a temporal scoring approach where depending on the score of previous T frame, the score for the current record is calculated. From the score of the current traffic, the decision about the traffic record is taken.

Chapter 3

Proposed Method

We propose a correlation based feature extraction method for Intrusion Detection System (IDS). The Intrusion Detection System follows anomaly detection approach where a normal traffic profile is created based on the data distribution of normal traffic in train set. The deviated traffic from the normal traffic profile is marked as anomaly.

In our proposed method, the normal traffic profile is characterized by the highly-related group of features in normal traffic. The highly related group of features in normal traffic is identified in a graph-based approach. The normal traffic profile is denoted by the mean and covariance matrix of the features in each of the groups. A distance function is used to calculate the deviation of traffic instances from the normal traffic profile. The distance from each of the groups is considered as the extracted features. Our proposed framework for intrusion detection has three steps: Normal Traffic Profile Generation, feature extraction and classification.

3.1 Normal Traffic Profile Generation

3.1.1 Filtering Normal Traffic

Anomaly is something which is deviated from the normal. An anomaly detection technique identifies anomalies by monitoring system activities and evaluating this activity with respect to normal characteristics. In our proposed intrusion detection system, we first broadly categorized network traffic data into two types. One is normal traffic data and another is intrusion traffic data. Here, we consider the normal network traffic samples as standard data samples and intrusion traffic samples as anomaly or data with unexpected behaviors. In this regard, we have intended to generate a normal traffic profile so that it can represent the standard properties of expected network traffic data. Due to generate a standard profile, the normal traffic instances are filtered from the train set.

F1	F2	F3	F4	F5	C
v0	v1	v2	v3	v4	N
v5	v6	v7	v8	v9	A
v10	v11	v12	v13	v14	N
v15	v16	v17	v18	v19	A
v20	v21	v22	v23	v24	N
v25	v26	v27	v28	v29	A
v30	v31	v32	v33	v34	A
v35	v36	v37	v38	v39	N
v40	v41	v42	v43	v44	N

Filter

F1	F2	F3	F4	F5	C
v0	v1	v2	v3	v4	N
v10	v11	v12	v13	v14	N
v20	v21	v22	v23	v24	N
v35	v36	v37	v38	v39	N
v40	v41	v42	v43	v44	N

Figure 3.1: Filtering Normal Traffic Instances

3.1.2 Constructing Groups of Highly Related Features

In our proposed method, the highly related group of features in normal traffic instances is identified by a graph based approach. Initially the mutual relationship status among features is calculated by a correlation function. We have used Pearson Correlation as relationship function among two features. The Pearson correlation between two features is denoted by:

$$r = \frac{\sum_i (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_i (x_i - \overline{x})^2 \sum (y_i - \overline{y})^2}} \tag{3.1}$$

where,
r = correlation coefficient
x_i = values of the x-feature in an instance
$\overline{x}$ = mean of the values of the x-feature
y_i = values of the y-feature in an instance
$\overline{y}$ = mean of the values of the y-feature

The mutual correlation value between all possible feature pairs are illustrated in Fig. 3.2 using a complete graph K_n. Here the weighted graph $G = (V, E)$ is formed considering the features as nodes V and the mutual correlation value between feature pairs as edges E.

From the complete graph K_n, a maximum spanning tree T_n is formed (Fig. 3.3). The maximum spanning tree consists of n nodes and (n-1) edges where the edges are maximum correlation values of feature pairs retain the graph connected. The extracted tree is minimally connected graph denotes the strong mutual correlation between features in normal traffic.

To form the groups of features, a cut-off value is defined. The edges having absolute weight

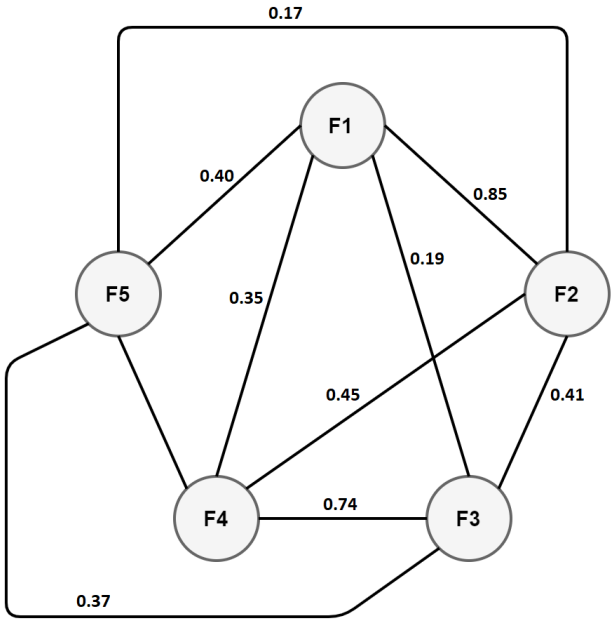


Figure 3.2: Complete graph of mutual correlation between feature pairs

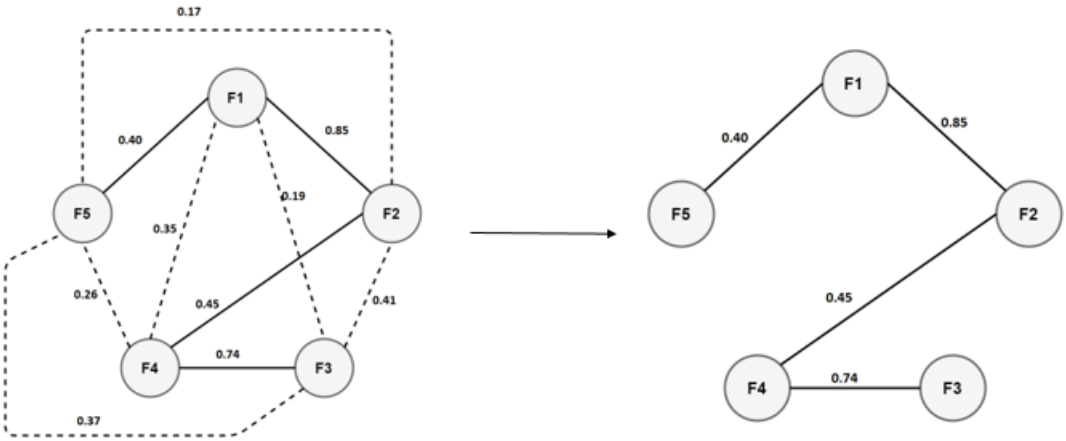


Figure 3.3: Formation of maximum spanning tree

below of the cut-off value is discarded. In this example (shown Fig. 3.4), we have considered the cut-off correlation value to be 0.5. Therefore, the edges containing weight below of 0.5 is discarded which result in three connected components. A connected component can be consist of multiple features or a single feature as well. Each of the connected components incorporates the features which are highly correlated.

From the highly correlated feature groups, we need to compute the mean and covariance matrix from all feature groups so that the ideal normal traffic can be characterized. To measure the deviation from any incoming traffic instance from the ideal normal traffic profile, some distance metrics can be used considering the mean and covariance of the groups.

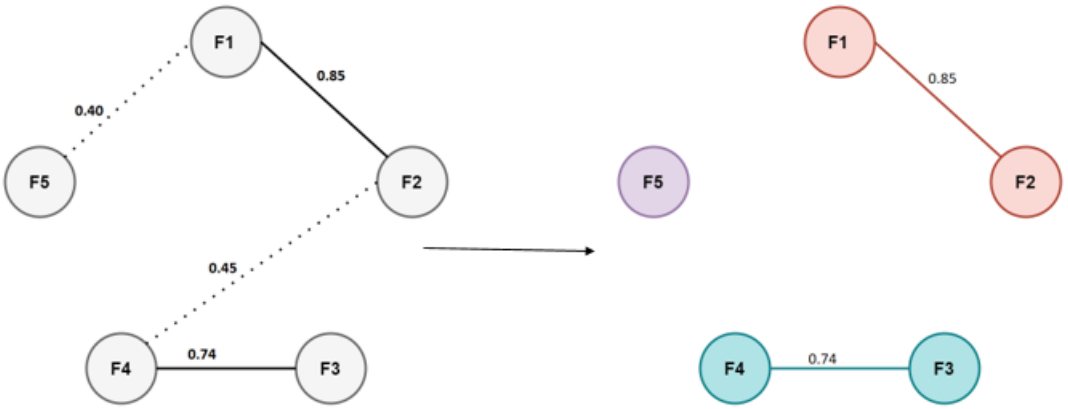


Figure 3.4: Formation of groups of features

3.2 Feature Extraction

The extracted features of our framework is the deviation from the ideal normal traffic profile. The number of features is equal to the number of highly correlated groups in normal traffic characteristics. Mahalanobis distance is used to calculate the deviation from normal traffic. Mahalanobis Distance between x_i and $\overline{x_j}$ can be denoted by the following equation.

$$f_j = \sqrt{(x_i - \overline{x_j})^T cov^{-1} (x_i - \overline{x_j})} \tag{3.2}$$

where,
f_j = j'th extracted feature
x_i = values of the x-features in an instance where the features belong to group-j
$\overline{x_j}$ = mean of the values of the x-feature belong to group-j
cov^{-1} = inverse of covariance matrix of the group of features

The mahalnobis distance for the normal traffic instances is expected to be close to zero whereas for the attack groups, the deviation is more. f groups in normal traffic profile forms f number of extracted features for instances. For example (Fig.-3.5), there are three groups of features in normal traffic profile. Now the amount of extracted features will also be three. The first extracted feature is the deviation of traffic instances from the features f1 and f2 participating in the group. Whereas the other features are the distance from respective feature groups.

The train set includes both the normal and attack instances. The extracted multidimensional features of train and test set are obtained by computing distance for each of the instances in train and test set from the normal traffic profile generated using the normal traffic of train set.

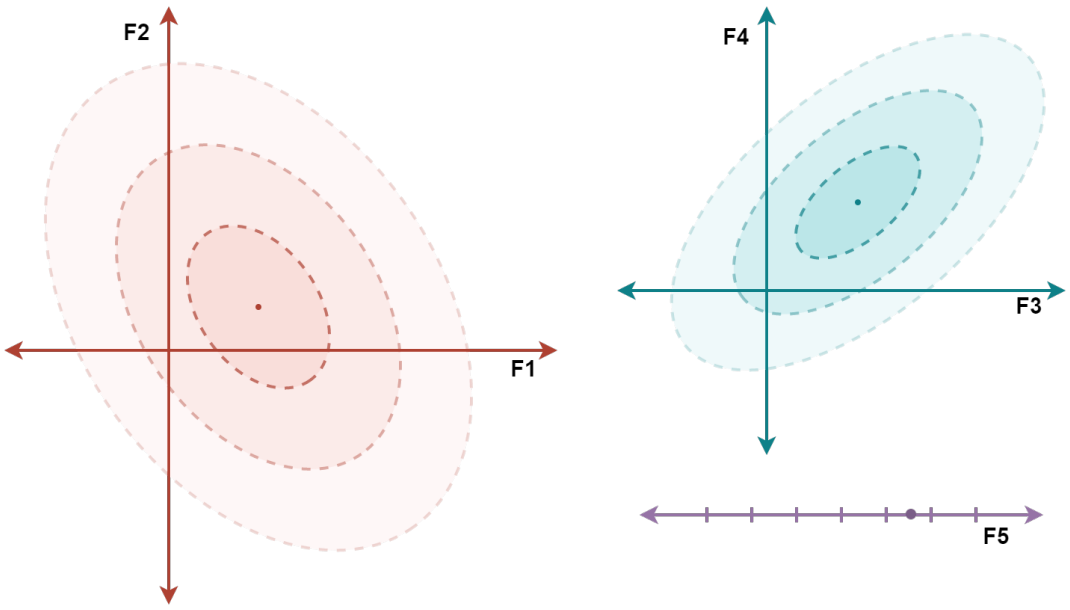


Figure 3.5: Distance from each of the group of features

F1	F2	F3	F4	F5	C		CD-1	CD-2	CD-3	C
v0	v1	v2	v3	v4	N	→	p1	p2	p3	N
v5	v6	v7	v8	v9	A	→	p4	p5	p6	A
v10	v11	v12	v13	v14	N	→	p7	p8	p9	N
v15	v16	v17	v18	v19	A	→	p10	p11	p12	A
v20	v21	v22	v23	v24	N	→	p13	p14	p15	N
v25	v26	v27	v28	v29	A	→	p16	p17	p18	A
v30	v31	v32	v33	v34	A	→	p19	p20	p21	A
v35	v36	v37	v38	v39	N	→	p22	p23	p24	N
v40	v41	v42	v43	v44	N	→	p25	p26	p27	N

Figure 3.6: Extracted Set of Features

3.3 Classification

The anomaly-detection is basically classifying an incoming traffic as normal or attack. A machine learning classification algorithm is used for the classification purpose. We fit the classifier with the extracted training data and evaluate the performance on extracted test data set.We apply **Random Forest** classifier as our classification algorithm. Random Forest classifier requires less amount of time for training multi-dimensional data with reduced overfitting compared to other classification algorithm. As intrusion detection system requires to be up-to-date, the efficiency in training time is beneficial.

Chapter 4

Result Analysis

As stated before, the proposed method aims to develop an efficient and generalized Network Intrusion Detection System. For this purpose, a correlation based feature extraction method is proposed. The performance of the proposed system is evaluated on five benchmark dataset. The experiments are performed by Python notebook on computer with 3.5 GHz Intel Core i7-7500 processor and 16GB RAM.

4.1 Benchmark Datasets

Using benchmark dataset for the evaluation of network intrusion detection system is a great challenge. Most often continuous traffic flow may contain normal traffic in major and can cause imbalance in dataset. So the researchers have moved towards synthetic dataset by a qualitative analysis of attack traffic behavior.

A dataset named KDD-CUP99 was released in 1999 where real-life normal traffic with simulated attack traffic was considered as sample of the dataset. By removing redundant traffic of KDD-CUP99 dataset, a modified version was released later named NSL-KDD dataset. In 2015, Kolias et al. published Aegean WiFi Intrusion Dataset (AWID) dataset, which includes real life normal traffic and wifi intrusive traffic. In 2015, the network security lab of University of New South Wales published another well-known synthetic dataset UNSW-NB15. In 2017, the Canadian Institute for Cybersecurity (CIC) published another intrusion detection system dataset named CIC-IDS2017.

4.1.1 KDD-CUP99 Dataset

The KDD-CUP99 dataset was published in 1999 and be the widely used dataset for intrusion detection system. The dataset contains 41 features where 32 of them are continuous and rest 9 are discrete. The dataset contains normal traffic and attack data of four major categories: Denial of Service (DoS), User to Root (U2R), Remote to Local (R2L) and Probe attack. KDD-cup99 is stored as three datasets. The complete and the largest one is called "Whole KDD". Whole-KDD contains about 4 million records. This is the original dataset gathering all the raw data records collected by the sniffer tool. The second dataset is called 'KDD-ten percent'. As the amount of data in original dataset is too high, it is computationally expensive to handle. So the ten percent dataset is released which contains 10% of the 'Whole-KDD'. As the feature values of KDD-CUP99 dataset contains raw traffic information, there are many missing values. So the third dataset was introduced called 'KDD-99 corrected' which contains a processed dataset. For our experiment, we have worked with 30% data of 'KDD-99 corrected'.

Attack	**No of Instances**	**Description**
Normal Traffic	582183	Legitimate network traffic samples
Denial of Service (DoS)	2330855	the attacker makes the computing resources too busy to handle traffic containing legitimate requests
User to Root (U2R)	31	the attacker starts from a normal user account and exploits some vulnerability to gain root access to the system
Remote to Local (R2L)	651	the attacker starts as packet sender to the system and exploits vulnerabilities to gain local access as an user of that system
Probe	24599	the attacker surveillance the system to get security information of the system

Table 4.1: Dataset Record Distribution of KDD-CUP99 dataset

4.1.2 UNSW-NB15 Dataset

The UNSW-NB15 dataset was published in 2015 by the network security lab of University of New South Wales. Most of the Intrusion Detection System(IDS) follows anomaly detection approach. Anomaly Detection works by learning possible characteristics and data distribution of normal traffic and the deviation of anomalies. However, most of the dataset prepared by scanning the real-world traffic seems to be dominated by normal traffic. To be an ideal dataset of experimentation, it is a balanced dataset is required.Moustafa et. al. [59] prepared the UNSW-NB15 by generating different attack and normal traffic in a simulated environment. The instances of this dataset is synthetic traffic from qualitative analysis of network attacks.

The total number of records is 2,540,044 which are stored in the four CSV files. Each of the instances contains 47 features. The dataset contains normal traffic and attack samples of nine categories: Fuzzers, Analysis, Backdoors, Denial of Service(DoS), Exploits, Generic, Reconnaissance, Shellcode and Worms.

Attack	No of Instances	Description
Normal Traffic	2218764	Normal and Legitimate transaction data
Fuzzers	24246	Attempt to suspend the system feeding randomly generated data
Analysis	2677	It includes html penetration or port scan attack
Backdoors	2329	Bypassing system security mechanism
DoS	16353	Attempt to make network resource unavailable or suspended
Exploits	44525	Attempt to attack by utilizing the vulnerabilities of the system
Generic	215481	A technique against block-cipher without considering about the structure
Reconnaissance	13987	Surveillance the system to get security and General information of the system
Shellcode	1511	Small piece of code used as payload to exploit software vulnerabilities
Worms	174	Malware that spreads copies of itself by replicating from computer to computer.

Table 4.2: Dataset Record Distribution of UNSW-NB15 dataset

4.1.3 NSL-KDD Dataset

The NSL-KDD dataset was published in 2009 as a modified version of the original KDDCup'99 dataset. This dataset solves several drawbacks of KDD-CUP99 dataset : redundant records and extreme imbalance of samples.As in the original KDD-CUP99, the Denial of Service (DoS) dominates other type of attacks which hinders machine learning classifier to learn the data distribution of other attack categories.The NSL-KDD contains reduced and balanced amount of sample for the attack categories. NSL-KDD dataset contains two subset: KDDTrain+ and KDDTest+. The KDDTrain+ subset contains total 125,973 samples where 58,630 instances of attack traffic and 67,343 instances of normal traffic. The KDDTest+ set contains total 22,544 samples.The dataset contains normal traffic and attack data of four categories same as KDD-CUP99. Each of the instances contains 41 features.

Attack	No of Instances	Description
Normal Traffic	77054	Legitimate network traffic samples
Denial of Service (DoS)	53385	the attacker makes the computing resources too busy to handle traffic containing legitimate requests
User to Root (U2R)	252	the attacker starts from a normal user account and exploits some vulnerability to gain root access to the system
Remote to Local (R2L)	3749	the attacker starts as packet sender to the system and exploits vulnerabilities to gain local access as an user of that system
Probe	14077	the attacker surveillance the system to get security information of the system

Table 4.3: Dataset Record Distribution of NSL-KDD dataset

4.1.4 Aegean WiFi Intrusion Dataset (AWID) Dataset

AWID dataset was published in 2015 as a collection of normal and intrusive traffic data from Wifi traffic of real network environments. Each of the instances contains 155 attributes. AWID-CLS dataset groups the instances into normal traffic and three main classes of attack including flooding, impersonation, and injection attack. We have conducted our experiment on AWID-CLSR dataset which contains 575,643 instances in total.

Attack	No of Instances	Description
Normal Traffic	530457	Benign and Legitimate traffic
Impersonation	20079	A malicious party impersonating to be another user to access the system to spread malware or steal data
Injection	16682	Attempt to inject malicious code into the system to get access to the database of the system
Flooding	8097	Attackers send a very high volume of data to make system resource suspended or unavailable for legitimate traffic

Table 4.4: Dataset Record Distribution of AWID dataset

4.1.5 CIC-IDS2017 Dataset

The CIC-IDS2017 dataset was published in 2017 by Canadian Institute for Cybersecurity(CIC).This dataset contains 2,830,743 instances divided into 8 files and each of the record sample has 78 features.CIC-IDS is a recent and one of the most up-to-date intrusion dataset which contains traffic from Normal and 14 attack categories.

Attack	No of Instances	Description
Benign	2272688	Normal and Legitimate traffic
DoS Hulk	230124	HTTP flooding DoS attack which overwhelms web servers
Portscan	158930	Scan for open or unused port of a server and exploit a known vulnerability of the network service
DDoS	128027	Flooding a targeted computer resource using more than one unique ip address
DoS GoldenEye	10293	HTTP flooding using GoldenEye tool. GoldenEye is a tool to tune parameters of the traffic randomly
FTP-Patator	7938	Brute-force attack to guess the login password of FTP
SSH-Patator	5897	Brute-force attack to guess the login password of SSH
DoS slowloris	5796	HTTP flooding using Slow Loris tool. Slow Loris makes new connections at a time interval but tries to make them open as long as possible
DoS Slowhttptest	5499	Open multiple HTTP connections to exceed the capacity of the server
Bot	1966	Uses trojans to breach security and then takes the control of victim computer remotely by Bot
Web Attack-Brute Force	1507	Trail-and-Error based BruteForce approach
Web Attack-XSS	652	The attackers attempt to inject malicious script on server
Infiltration	36	Surveillance the victim system
Web Attack-Sql Injection	21	Code injection in victim system database through entry-field of SQL request
Heartbleed	11	Attackers access OpenSSL memory exploiting vulnerabilities of OpenSSL protocol

Table 4.5: Dataset Record Distribution of CIC-IDS2017 dataset

4.2 Evaluation Metrics

The performance of our proposed model on each of the datasets is evaluated in terms of Accuracy, True Positive Rate, False Positive Rate, Precision, Recall, AUC-ROC, F-measures and Matthews Correlation Coefficient (MCC).

Accuracy: Accuracy is the ability of the trained classifier to correctly classify a traffic instance

as normal or attack. The accuracy is measured by:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.1}$$

True Positive Rate (TPR)/ Sensitivity/ Recall: Rate of the intrusions correctly classified as intrusion by the classifier.The True Positive Rate(TPR) is measured by:

$$TPR = \frac{TP}{TP + FN} \tag{4.2}$$

False Positive Rate (FPR)/ False Alarm: Rate of the normal traffic classified as intrusion by the classifier.The False Positive Rate(FPR) is measured by:

$$FPR = \frac{FP}{FP + TN} \tag{4.3}$$

Precision: Precision represents exactness in the detected attacks.Precision is measured by:

$$Precision = \frac{TP}{TP + FP} \tag{4.4}$$

Area Under Curve-Receiver Operating Characteristic(AUC-ROC):AUC-ROC represents the exact detection for both normal and attack traffic.

F-measure:F-measure is the weighted harmonic mean of the precision and recall measures of the trained classifier.

$$F - Measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4.5}$$

Matthews Correlation Coefficient(MCC):MCC [60] is a measure of the quality of classification. MCC is calculated using:

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{4.6}$$

4.3 Performance Evaluation

4.3.1 Intrusion Detection Report

Dataset	Detection Rate (Normal)	Detection Rate (Attack)	Accuracy	AUC-ROC	Recall / TPR	FAR	Precision	F-measure	MCC
KDD-CUP99	0.9998	0.9996	0.9996	0.9999	0.9996	0.0001	0.9999	0.9997	0.9989
UNSW-NB15	0.9945	0.9681	0.9912	0.9995	0.9681	0.0055	0.9623	0.9652	0.9601
NSL-KDD	0.9987	0.9995	0.9991	0.9999	0.9995	0.0012	0.9986	0.9991	0.9982
AWID	0.9922	0.9323	0.9875	0.9877	0.9323	0.0077	0.9102	0.9211	0.9144
CIC-IDS2017	0.9942	0.9912	0.9937	0.9989	0.9912	0.0057	0.9768	0.9840	0.98

Table 4.6: Intrusion Detection Report of Our Proposed Method on Benchmark Datasets

Table-4.6 summarizes the performance of our proposed method based on five benchmark datasets. Our proposed method achieves a very good detection rate and accuracy in both percentile and Area-Under the curve metric. Moreover, a very low False Alarm Rate (FAR) is achieved with high Recall, Precision and F-Measure value for all of the benchmark datasets. However, the performance of our proposed method is comparatively lower in case of AWID-dataset than four other benchmark datasets.

4.3.2 Detection Performance by Class

KDD-CUP99 Dataset

Class	Correctly Classified	Misclassified	Detection Rate
Normal Traffic	582119	64	0.9998
Denial of Service (DoS)	2330676	179	0.9999
User to Root (U2R)	10	21	0.3225
Remote to Local (R2L)	537	114	0.8248
Probe	24018	581	0.9763

Table 4.7: Intrusion Detection on KDD-CUP99 dataset

Table - 4.7 represents detection rate for each of the classes of KDD-CUP99 dataset. Our proposed method performs extremely well for DoS, Probe attack and Normal traffic. However, the

performance degrades for the attack class having only few number of samples. As the number of samples of U2R is negligible compared to the other classes of traffic, the classifier falis to learn the characteristics of U2R properly.

UNSW-NB15 Dataset

Class	Correctly Classified	Misclassified	Detection Rate
Normal Traffic	2206574	12190	0.9945
Generic	215375	106	0.9995
Exploits	43517	1008	0.9773
Fuzzers	16125	8121	0.6650
DoS	16106	247	0.9848
Reconnaissance	13913	74	0.9947
Analysis	2198	479	0.8210
Backdoors	2317	12	0.9948
Shellcode	1313	198	0.8689
Worms	171	3	0.9827

Table 4.8: Intrusion Detection on UNSW-NB15 dataset

Table - 4.8 demonstrates detection rate for each of the classes of UNSW-NB15 dataset. Our proposed method produces significant performance for the classes of traffic. However, the detection rate for 'Fuzzers' class is comparatively low as Fuzzers attack can mimic the normal network traffic.

NSL-KDD Dataset

Class	Correctly Classified	Misclassified	Detection Rate
Normal Traffic	76958	96	0.9987
Denial of Service (DoS)	53375	10	0.9998
User to Root (U2R)	248	4	0.9841
Remote to Local (R2L)	3734	15	0.9959
Probe	14074	3	0.9997

Table 4.9: Intrusion Detection on NSL-KDD dataset

Table-4.9 represents detection rate of the classes of NSL-KDD dataset. NSL-KDD dataset is the balanced version of KDD-CUP99. Hence, our proposed method has achieved good performance even for the attack classes it performed poor in KDD-CUP99(Table-4.7) due to extreme imbalance in dataset.

AWID Dataset

Class	Correctly Classified	Misclassified	Detection Rate
Normal Traffic	526332	4125	0.9922
Impersonation	19165	914	0.9544
Injection	16187	495	0.9703
Flooding	6471	1626	0.7991

Table 4.10: Intrusion Detection on AWID dataset

AWID dataset was created from direct pcap files of wifi traffic. This dataset suffers from inconsistent feature values as well as imbalance of data. However, our proposed method has achieved a considerable performance for AWID Dataset (Table-4.10).

CIC-IDS2017 Dataset

Class	Correctly Classified	Misclassified	Detection Rate
Benign	2259623	13065	0.9942
DoS Hulk	228834	1290	0.9943
Portscan	158887	43	0.9997
DDoS	127916	111	0.9991
DoS GoldenEye	10118	175	0.9829
FTP-Patator	7898	40	0.9949
SSH-Patator	5783	114	0.9806
DoS slowloris	5345	451	0.9221
DoS Slowhttptest	5138	361	0.9343
Botnet	1559	407	0.7929
Web Attack-Brute Force	275	1232	0.1824
Web Attack-XSS	31	621	0.0475
Infiltration	13	23	0.3611
Web Attack-Sql Injection	13	8	0.6190
Heartbleed	10	1	0.9090

Table 4.11: Intrusion Detection on CIC-IDS2017 dataset

Table - 4.11 represents detection rate for each of the classes of CIC-IDS2017 dataset. Our proposed method performs good for most of the attacks.But still faces issue of imbalance in dataset and performs poor for the attack classes having comparatively small amount of instances. Moreover, as Brute Force-Web Attack takes the form of normal traffic, the detection rate for Brute Force-Web Attack is extremely poor.

4.3.3 Comparison with Literature

This section provides a comparative analysis between our proposed method and the state of the art methods in literature. As we have evaluated our proposed method on five different benchmark datasets, we present here comparative analysis with respective method only.

Result Comparison on KDD-CUP99 Dataset

In Fig.4.1 we compare our proposed method with state-of the art ABC-method [20]. Our proposed method outperforms the ABC-method in terms of detection rate of attack for most of the attack. However, our method performs poorer in case of the U2R attack. The ABC method

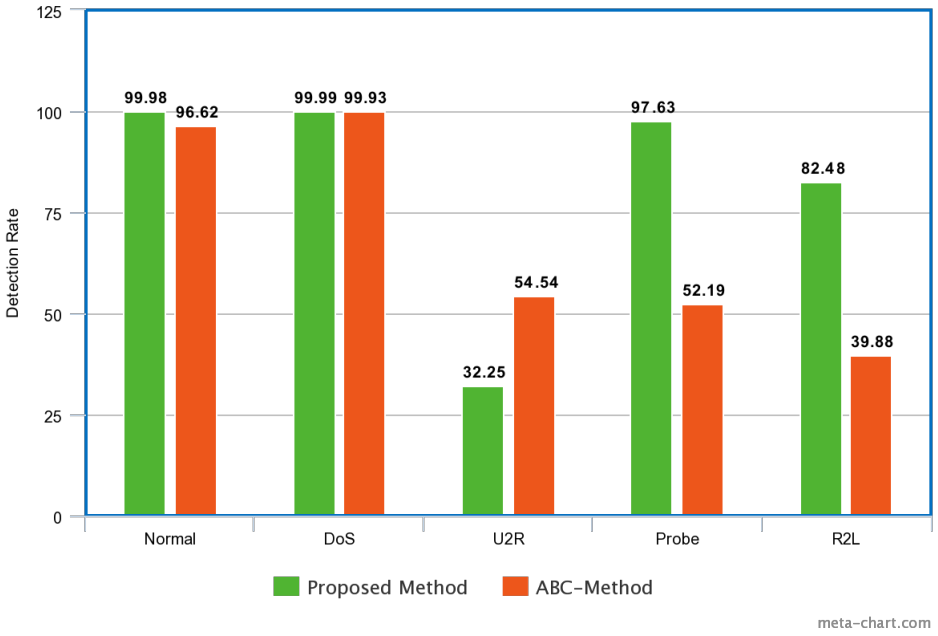


Figure 4.1: Performance comparison between proposed method and ABC method on KDD-CUP99 Dataset

is trained on the normal traffic only whereas our proposed method considers both the normal traffic as well as the deviations for the attack classes. As the dataset is extremely imbalance and the number of instances of U2R is almost negligible compared to the other attack categories, the classifier of our proposed method fails to learn the U2R attack properly.

Result Comparison on UNSW-NB15 Dataset

We observe from table-4.12 that our proposed method outperforms all of the methods in literature in terms of accuracy and detection rate. Moreover, our method achieves the minimum false alarm rate compared to other methods in literature.

Method	Accuracy	Detection Rate	False Alarm Rate
EM-Clustering [61]	78.47	-	-
DT [61]	85.56	-	-
CASCADE-ANN [62]	86.40	86.74	13.10
M-Stream [58]	-	90.5	-
ABC-Method [20]	83.76	85.67	-
ICVAE-DNN [63]	89.08	95.68	19.01
Proposed Method	**99.12**	**96.81**	**0.55**

Table 4.12: Comparison of detection performance with methods in literature on UNSW-NB15 Dataset

Our method achieved accuracy score of 99.12% and attack detection rate of 96.81% whereas the closest performance in literature is of ICVAE-DNN [63] method which used Deep Learning approach and achieved detection reate of 95.68%.

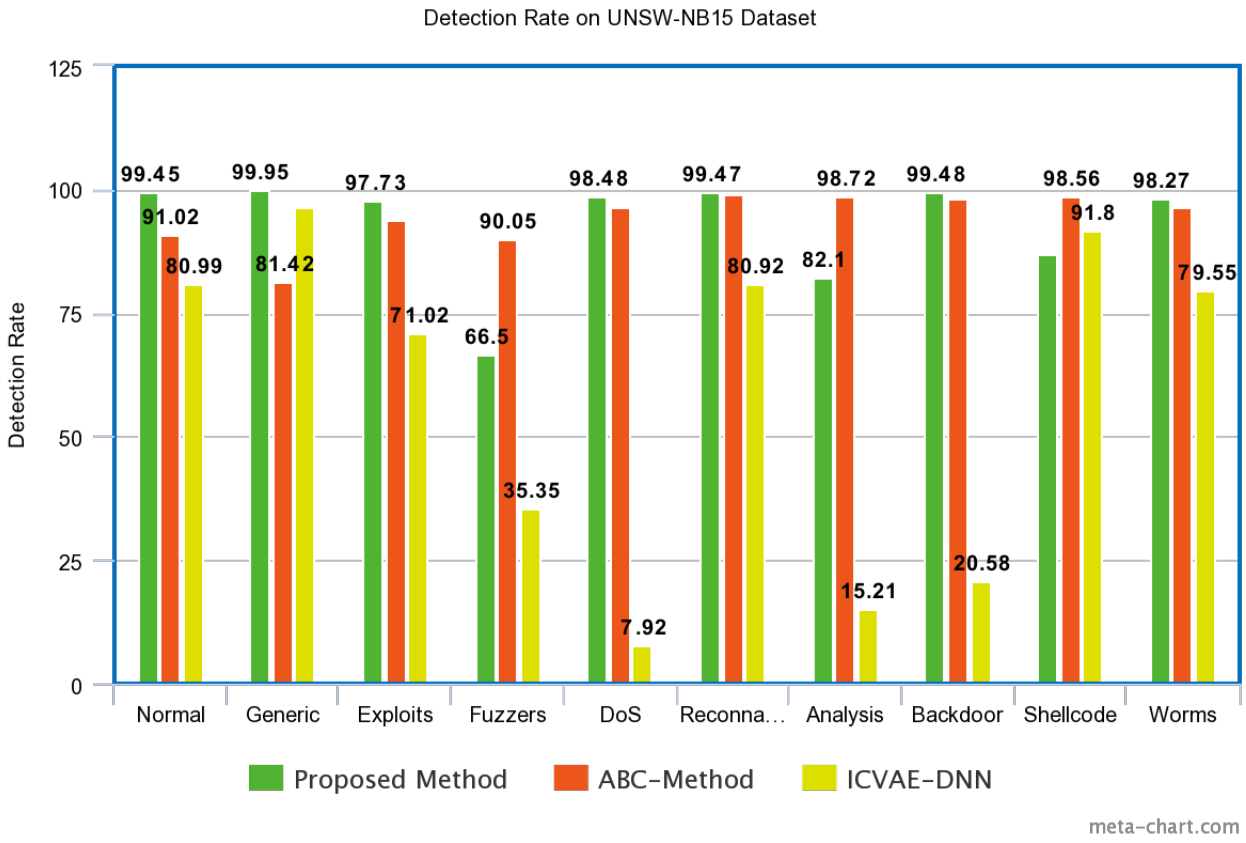


Figure 4.2: Class specific performance comparison between proposed method, ABC method and ICVAE-DNN on UNSW-NB15 Dataset

Fig.-4.2 illustrates the comparison between our proposed method, ABC method [20] and ICVAE-DNN [63] in terms of detection rate for individual traffic classes. Our proposed method outper-

forms the performance of the referenced methods for most of the attack classes. However,our proposed method underperforms in case of 'Fuzzers' attack.Fuzzers attack can mimic the normal network traffic keeping the data distribution in the range of normal traffic and tuning the parameters only.

Result Comparison on NSL-KDD Dataset

Method	Accuracy	Detection Rate	False Alarm Rate
STL [64]	74.38	62.99	7.21
RNN-IDS [65]	81.29	-	-
ICVAE-DNN [63]	85.97	77.43	2.74
DL-based IOT attack [66]	98.2	98.23	1.73
Proposed Method	**99.91**	**99.95**	**0.12**

Table 4.13: Comparison of detection performance with methods in literature on NSL-KDD Dataset

Table-4.13 represents that our proposed method outperforms the methods in literature in terms of accuracy and detection rate with the least False Alarm Rate.

Fig.-4.3 illustrates the comparison between our proposed method, DL-IOT [66] and ICVAE-

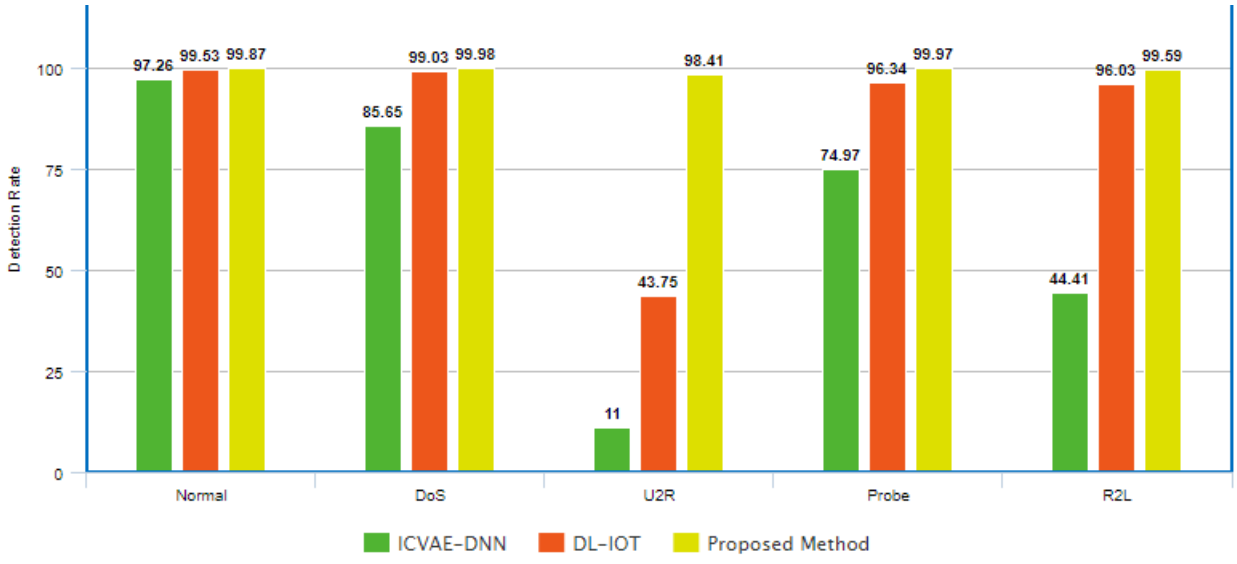


Figure 4.3: Class specific performance comparison between proposed method, DL-based IOT and ICVAE-DNN on NSL-KDD Dataset

DNN [63] in terms of detection rate for individual traffic classes. Our proposed method outperforms the State-of-the-art method DL-IOT for all of the traffic classes. As NSL-KDD attains balance in dataset, our method overcomes the poor performance for U2R attack on KDD-CUP99 dataset.

Result Comparison on AWID Dataset

Method	Accuracy	Detection Rate	False Alarm Rate
Empirical-Wifi-Anomaly [67]	96.19	**96.2**	**0.437**
DL-Wifi-Anomaly [68]	98.66	-	-
Proposed Method	**98.75**	93.23	0.77

Table 4.14: Comparison of detection performance with methods in literature on AWID Dataset

For AWID-dataset, our proposed method overperforms the methods in literature in the scale of accuracy. However, still Empirical-Wifi-Anomaly method [67] holds the best detection and False Alarm Rate(FAR).

Fig.-4.4 represents the comparison between our proposed method, Empirical-Wifi-Anomaly

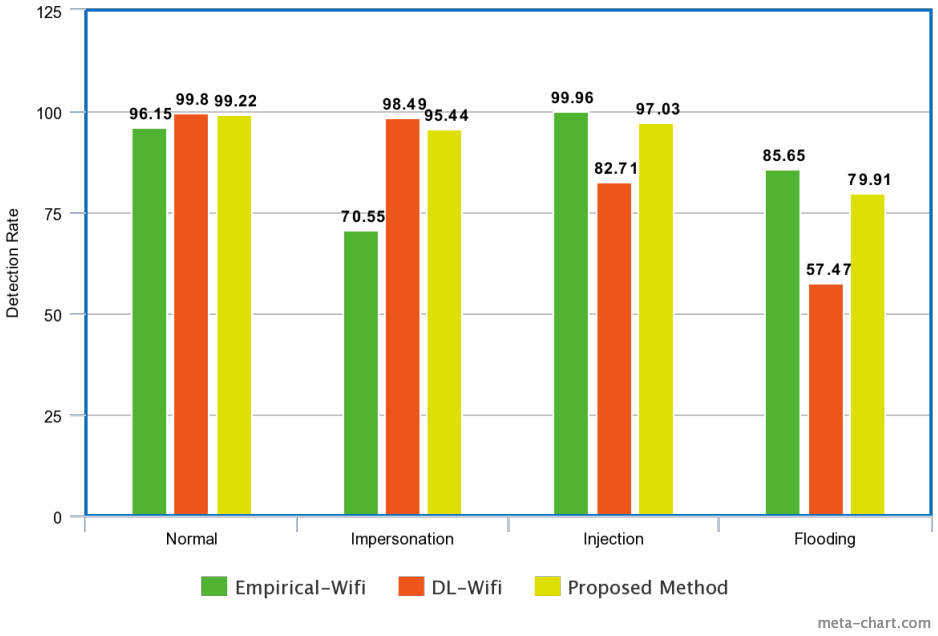


Figure 4.4: Class specific performance comparison between proposed method, Empirical-Wifi-Anomaly and DL-Wifi-Anomaly on AWID Dataset

[67] and DL-Wifi-Anomaly [68] in the scale of detection rate for all the traffic classes. Though our proposed method perform standalone for all the attacks , it can't outperform the SOTA performance of other methods.

Result Comparison on CIC-IDS-2017 Dataset

Table-4.15 represents the comparative analysis of our proposed method with the methods in literature using CIC-IDS2017 dataset. Our proposed method achieved comparatively better accuracy whereas Three-Layer-Architecture still ahead in Detection rate and False Alarm Rate.

Method	Accuracy	Detection Rate	False Alarm Rate
Hierarchical-IDS [69]	96.66	94.47	1.145
Three-Layer-Architecture [70]	99.3	**99.36**	**0.22**
Proposed Method	**99.37**	99.12	0.57

Table 4.15: Comparison of detection performance with methods in literature on CIC-IDS2017 Dataset

Fig.-4.5 illustrates the comparison between our proposed method and the methods in literature.

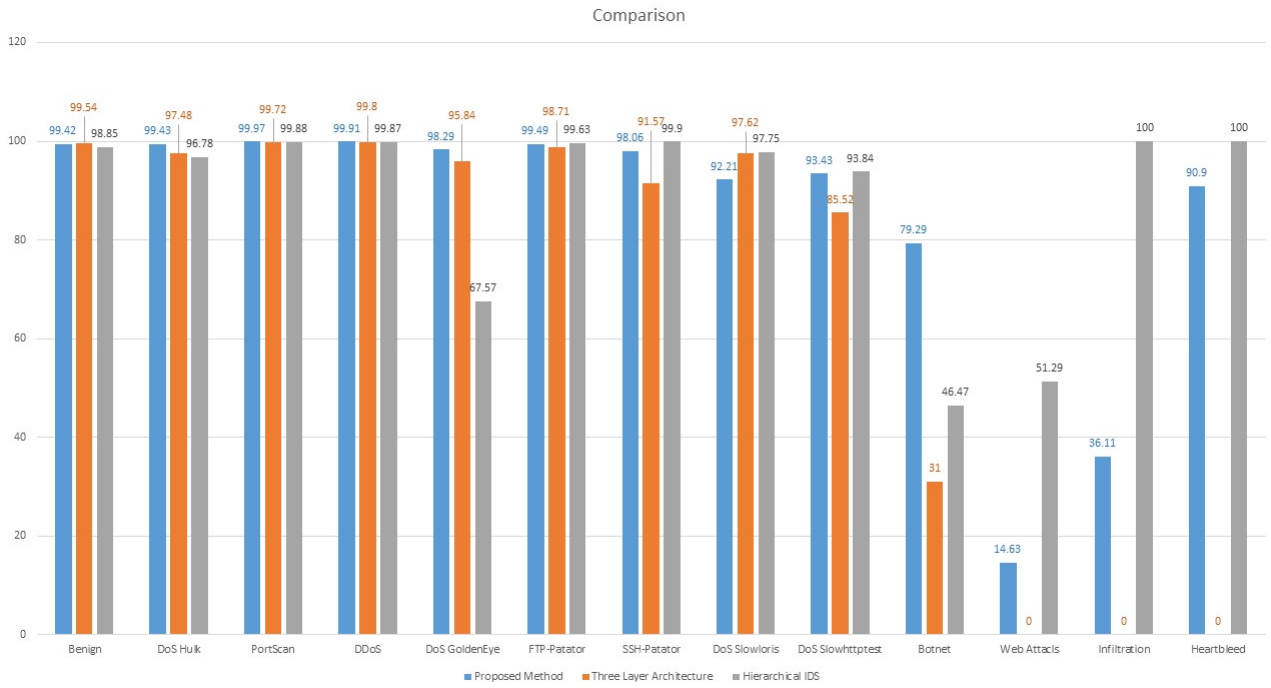


Figure 4.5: Class specific performance comparison between proposed method, Three-Layer-Architecture and Hierarchical-IDS on CIC-IDS2017 Dataset

Our proposed method achieved a high detection rate for all the attack classes except Infiltration and Web Attacks. The imbalance in dataset is the probable reason behind this decreasing in performance.

4.3.4 Effect of Imbalance in dataset

Most of the benchmarking IDS datasets are imbalanced in data distribution. The machine learning algorithms fail to learn the classes with small amount of instances compared to the classes with large amount of sample. As our proposed method uses Random Forest classifier algorithm for anomaly detection and consider the attack classes while training, the method suffers from imbalance in IDS datasets. However, the distance based anomaly detection classifiers learn the

characteristics of normal traffic only and don't take the data distribution of attack categories into consideration. So, the distance based classifiers are less prone to the imbalance in attack instances. Fig. - 4.6 represents the performance of our proposed method and a distance-based

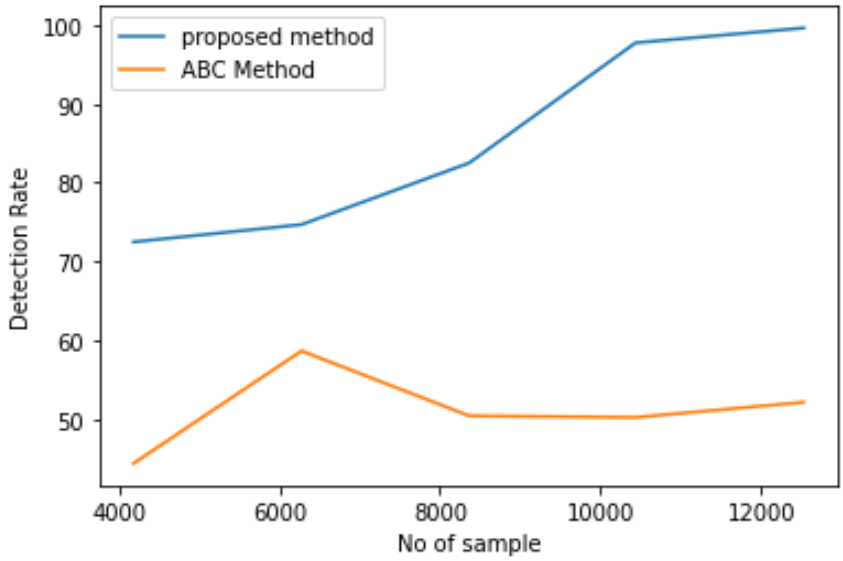


Figure 4.6: Effect of Number of Samples on Detection Rate

anomaly detection classifier ABC-method [20] for different number of instances of Probe attack in KDD-CUP99 dataset. The performance of our proposed method increases with the increase in number of samples. On the contrary, the performance of ABC-method remains almost same irrespective of the number of instances in attack categories.

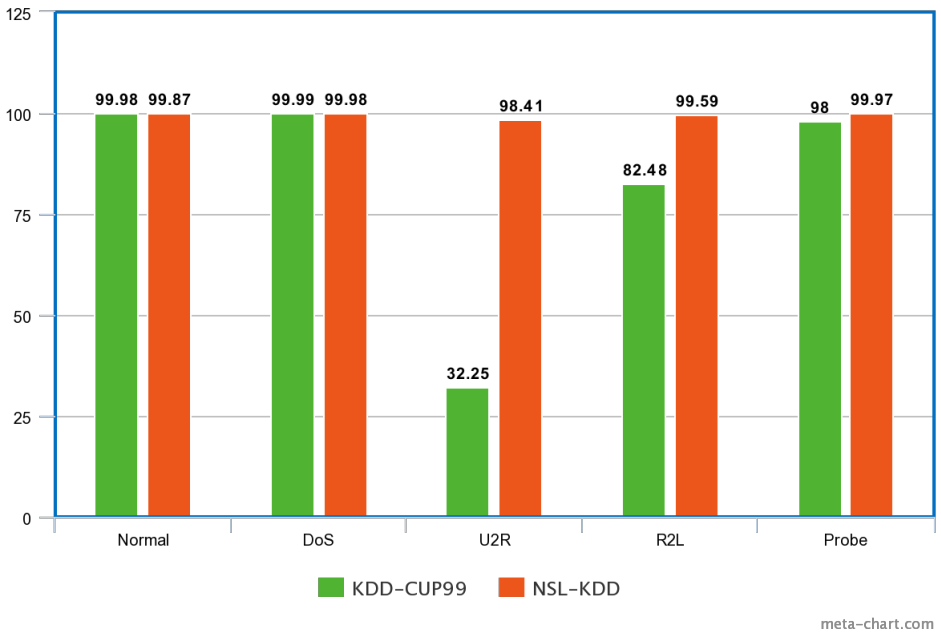


Figure 4.7: Comparison of proposed method in KDD-CUP99 and NSL-KDD dataset

The NSL-KDD dataset was published as a balanced subset of KDD-CUP99 dataset. So, our

proposed method gained an immense increase in detection rate for the attack categories it was performing poor in KDD-CUP99 dataset. Fig.-4.7 illustrates the comparative performance of our proposed method in KDD-CUP99 and NSL-KDD dataset.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

Our proposed framework is a novel approach of anomaly detection which passes through two step of learning: normal traffic profile generation and training classifier with the deviation of instances from normal traffic profile. The proposed frameworks detects the anomalous traffic more efficiently which is evaluated on five benchmark datasets.

Our proposed method of feature extraction and intrusion detection framework is generalized and efficient. It outperforms the existing intrusion detection approaches in literature in terms of detection rate. Moreover, the existing multi-variate feature extraction approach for anomaly detection (AD) suffer from the curse of dimensionality on the contrary our proposed reduces the number of features.

5.2 Future Work

The result analysis illustrates that our proposed method is sensitive to the imbalance of classes of traffic in benchmark datasets. Therefore, our method can be improved incorporating any oversampling or data imbalance handling methods.

References

[1] "Cyber security report." https://docs.broadcom.com/doc/istr-22-2017-en. Accessed: 2017-07-10.

[2] "Cyber security summary-2020." https://symantec-enterprise-blogs.security.com/blogs/feature-stories/symantec-security-summary-june-202 Accessed: 2020-06-04.

[3] M. Husák, J. Komárková, E. Bou-Harb, and P. eleda, "Survey of attack projection, prediction, and forecasting in cyber security," *IEEE Communications Surveys Tutorials*, vol. 21, pp. 640–660, 2019.

[4] I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, 12 2019.

[5] T. Porter and M. Gough, "Chapter 7 - active security monitoring," in *How to Cheat at VoIP Security* (T. Porter and M. Gough, eds.), How to Cheat, pp. 185–206, Burlington: Syngress, 2007.

[6] H.-J. Liao, C.-H. Richard Lin, Y.-C. Lin, and K.-Y. Tung, "Intrusion detection system: A comprehensive review," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 16–24, 2013.

[7] C. Pascoal, M. R. Oliveira, R. Valadas, P. Filzmoser, P. Salvador, and A. Pacheco, "Robust feature selection and robust pca for internet traffic anomaly detection," *2012 Proceedings IEEE INFOCOM*, pp. 1755–1763, 2012.

[8] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," *2010 IEEE Symposium on Security and Privacy*, pp. 305–316, 2010.

[9] R. Zuech and T. Khoshgoftaar, "A survey on feature selection for intrusion detection," pp. 150–155, 01 2015.

[10] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, pp. 491–502, 2005.

[11] Y. Li, J.-L. Wang, Z.-H. Tian, T.-B. Lu, and C. Young, "Building lightweight intrusion detection system using wrapper-based feature selection mechanisms," *Computers Security*, vol. 28, no. 6, pp. 466–475, 2009.

[12] I. Guyon, S. Gunn, M. Nikravesh, and L. Zadeh, "Feature extraction: foundations and applications," 01 2006.

[13] H. T. Nguyen, K. Franke, and S. Petrovic, "Feature extraction methods for intrusion detection systems," 2012.

[14] R. Agrawal, T. Imieli'nski, and A. Swami, "Mining association rules between sets of items in large databases," vol. 22, pp. 207–216, 01 1993.

[15] H. Mannila and H. Toivonen, "Discovering generalized episodes using minimal occurrences," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96)* (E. Simoudis, J. Han, and U. Fayyad, eds.), (United States), pp. 146–151, AAAI Press, Aug. 1996.

[16] K. Wang, J. Parekh, and S. Stolfo, "Anagram: A content anomaly detector resistant to mimicry attack," vol. 4219, pp. 226–248, 09 2006.

[17] N. Ye, S. M. Emran, Q. Chen, and S. Vilbert, "Multivariate statistical analysis of audit trails for host-based intrusion detection," *IEEE Trans. Computers*, vol. 51, pp. 810–820, 2002.

[18] D. Yeung, S. Jin, and X. Wang, "Covariance-matrix modeling and detecting various flooding attacks," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 37, pp. 157–169, 2007.

[19] Z. Tan, A. Jamdagni, X. He, P. Nanda, and R. Liu, "A system for denial-of-service attack detection based on multivariate correlation analysis," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, pp. 447–456, 02 2014.

[20] Q. Li, Z. Tan, A. Jamdagni, P. Nanda, X. He, and W. Han, "An intrusion detection system based on polynomial feature correlation analysis," *2017 IEEE Trustcom/BigDataSE/ICESS*, pp. 978–983, 2017.

[21] I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, 12 2019.

[22] F. Gottwalt, E. Chang, and T. Dillon, "Corrcorr: A feature selection method for multivariate correlation network anomaly detection techniques," *Comput. Secur.*, vol. 83, pp. 234–245, 2019.

[23] X. J. Zhou and T. S. Dillon, "A statistical-heuristic feature selection criterion for decision tree induction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 8, pp. 834–841, 1991.

[24] J. Viinikka, H. Debar, L. Mé, A. Lehikoinen, and M. Tarvainen, "Processing intrusion detection alert aggregates with time series modeling," *Information Fusion*, vol. 10, no. 4, pp. 312–324, 2009. Special Issue on Information Fusion in Computer Security.

[25] Qingtao Wu and Zhiqing Shao, "Network anomaly detection using time series analysis," in *Joint International Conference on Autonomic and Autonomous Systems and International Conference on Networking and Services - (icas-isns'05)*, pp. 42–42, 2005.

[26] N. Walkinshaw, R. Taylor, and J. Derrick, "Inferring extended finite state machine models from software executions," *Empirical Software Engineering*, vol. 21, pp. 811–853, 2013.

[27] I. Studnia, E. Alata, V. Nicomette, M. Kaâniche, and Y. Laarouchi, "A language-based intrusion detection approach for automotive embedded networks," *Int. J. Embed. Syst.*, vol. 10, pp. 1–12, 2018.

[28] G. Kim, S. Lee, and S. Kim, "A novel hybrid intrusion detection method integrating anomaly detection with misuse detection," *Expert Systems with Applications*, vol. 41, no. 4, Part 2, pp. 1690–1700, 2014.

[29] P. S. Kenkre, A. Pai, and L. Colaco, "Real time intrusion detection and prevention system," in *Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014* (S. C. Satapathy, B. N. Biswal, S. K. Udgata, and J. Mandal, eds.), (Cham), pp. 405–411, Springer International Publishing, 2015.

[30] S. Dua and X. Du, *Data Mining and Machine Learning in Cybersecurity*. USA: Auerbach Publications, 1st ed., 2011.

[31] K. Bajaj and A. Arora, "Improving the intrusion detection using discriminative machine learning approach and improve the time complexity by data mining feature selection methods," *International Journal of Computer Applications*, vol. 76, pp. 5–11, 08 2013.

[32] S. Thaseen and C. Kumar, "An analysis of supervised tree based classifiers for intrusion detection system," *2013 International Conference on Pattern Recognition, Informatics and Mobile Engineering*, pp. 294–299, 2013.

[33] A. Khraisat, I. Gondal, and P. Vamplew, "An anomaly intrusion detection system using c5 decision tree classifier," in *Trends and Applications in Knowledge Discovery and Data Mining* (M. Ganji, L. Rashidi, B. C. M. Fung, and C. Wang, eds.), (Cham), pp. 149–155, Springer International Publishing, 2018.

[34] L. Koc, T. A. Mazzuchi, and S. Sarkani, "A network intrusion detection system based on a hidden naïve bayes multiclass classifier," *Expert Systems with Applications*, vol. 39, no. 18, pp. 13492–13500, 2012.

[35] M. S. Hoque, M. Mukit, and M. A. N. Bikas, "An implementation of intrusion detection system using genetic algorithm," *International Journal of Network Security Its Applications*, vol. 4, pp. 109–120, 03 2012.

[36] S. Elhag, A. Fernández, A. Bawakid, S. Alshomrani, and F. Herrera, "On the combination of genetic fuzzy systems and pairwise learning for improving detection rates on intrusion detection systems," *Expert Systems with Applications*, vol. 42, no. 1, pp. 193–202, 2015.

[37] Y. Li, J. Xia, S. Zhang, J. Yan, X. Ai, and K. Dai, "An efficient intrusion detection system based on support vector machines and gradually feature removal method," *Expert Systems with Applications*, vol. 39, no. 1, pp. 424–430, 2012.

[38] C. Annachhatre, T. Austin, and M. Stamp, "Hidden markov models for malware classification," *Journal of Computer Virology and Hacking Techniques*, vol. 11, 05 2014.

[39] W.-C. Lin, S.-W. Ke, and C.-F. Tsai, "Cann: An intrusion detection system based on combining cluster centers and nearest neighbors," *Knowledge-Based Systems*, vol. 78, pp. 13–21, 2015.

[40] R. A. R. Ashfaq, X.-Z. Wang, J. Z. Huang, H. Abbas, and Y.-L. He, "Fuzziness based semi-supervised learning approach for intrusion detection system," *Information Sciences*, vol. 378, pp. 484–497, 2017.

[41] N. B. R. S. S. P. P. Rath, L. F. S. Davoodkhani, and A. T. M. Ahmed, "A prototype multiview approach for reduction of false alarm rate in network intrusion detection system," *Journal of Computer Networks and Communications*, vol. 5, pp. 49–59, 2017.

[42] J. Lyngdoh, M. I. Hussain, S. Majaw, and H. Kalita, *An Intrusion Detection Method Using Artificial Immune System Approach: Second International Conference, ICAICR 2018, Shimla, India, July 14–15, 2018, Revised Selected Papers, Part II*, pp. 379–387. 01 2019.

[43] M. Goldstein, "Fastlof: An expectation-maximization based local outlier detection algorithm," *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pp. 2282–2285, 2012.

[44] H. Sadreazami, A. Mohammadi, A. Asif, and K. Plataniotis, "Distributed-graph-based statistical approach for intrusion detection in cyber-physical systems," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 4, pp. 137–147, 2018.

[45] A. A. Aburomman and M. B. Ibne Reaz, "A novel svm-knn-pso ensemble method for intrusion detection system," *Applied Soft Computing*, vol. 38, pp. 360–372, 2016.

[46] M. Jabbar, R. Aluvalu, and S. S. Reddy S, "Rfaode: A novel ensemble intrusion detection system," *Procedia Computer Science*, vol. 115, pp. 226–234, 2017. 7th International Conference on Advances in Computing Communications, ICACC-2017, 22-24 August 2017, Cochin, India.

[47] D. Gaikwad and R. C. Thool, "Intrusion detection system using bagging with partial decision treebase classifier," *Procedia Computer Science*, vol. 49, pp. 92–98, 2015. Proceedings of 4th International Conference on Advances in Computing, Communication and Control (ICAC3'15).

[48] N. Moustafa, B. Turnbull, and K. R. Choo, "An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4815–4830, 2019.

[49] N. Paulauskas and J. Auskalnis, "Analysis of data pre-processing influence on intrusion detection using nsl-kdd dataset," in *2017 Open Conference of Electrical, Electronic and Information Sciences (eStream)*, pp. 1–5, 2017.

[50] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Computer Networks*, vol. 174, p. 107247, 2020.

[51] M. Abdullah, A. Balamash, A. Al-Shannaq, and S. Almabdy, "Enhanced intrusion detection system using feature selection method and ensemble learning algorithms," *International Journal of Computer Science and Information Security,*, vol. 16, pp. 48–55, 02 2018.

[52] H. Hota and A. Shrivas, "Decision tree techniques applied on nsl-kdd data and its comparison with various feature selection techniques," *Smart Innovation, Systems and Technologies*, vol. 27, pp. 205–212, 01 2014.

[53] C. Khammassi and S. Krichen, "A ga-lr wrapper approach for feature selection in network intrusion detection," *Computers Security*, vol. 70, pp. 255–277, 2017.

[54] H. H. Pajouh, R. Javidan, R. Khayami, A. Dehghantanha, and K. R. Choo, "A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detec-

tion in iot backbone networks," *IEEE Transactions on Emerging Topics in Computing*, vol. 7, no. 2, pp. 314–323, 2019.

[55] N. Moustafa and J. Slay, "A hybrid feature selection for network intrusion detection systems: Central points," *ArXiv*, vol. abs/1707.05505, 2017.

[56] X. Yuan, C. Li, and X. Li, "Deepdefense: Identifying ddos attack via deep learning," *2017 IEEE International Conference on Smart Computing (SMARTCOMP)*, pp. 1–8, 2017.

[57] P. Singh, J. Jaykumar, A. Pankaj, and R. Mitra, "Edge-detect: Edge-centric network intrusion detection using deep neural network," *ArXiv*, vol. abs/2102.01873, 2021.

[58] S. Bhatia, A. Jain, P. Li, R. Kumar, and B. Hooi, "Mstream: Fast anomaly detection in multi-aspect streams," 2021.

[59] N. Moustafa and J. Slay, "Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set)," in *2015 Military Communications and Information Systems Conference (MilCIS)*, pp. 1–6, 2015.

[60] Y. Liu, J. Cheng, C. Yan, X. Wu, and F. Chen, "Research on the matthews correlation coefficients metrics of personalized recommendation algorithm evaluation," *International Journal of Hybrid Information Technology*, vol. 8, pp. 163–172, 2015.

[61] N. Moustafa and J. Slay, "The evaluation of network anomaly detection systems: Statistical analysis of the unsw-nb15 data set and the comparison with the kdd99 data set," *Information Security Journal A Global Perspective*, vol. 25, pp. 1–14, 01 2016.

[62] M. Baig, M. Awais, and E. El-Alfy, "A multiclass cascade of artificial neural network for network intrusion detection," *J. Intell. Fuzzy Syst.*, vol. 32, pp. 2875–2883, 2017.

[63] Y. Yang, K. Zheng, C. Wu, and Y. Yang, "Improving the classification effectiveness of intrusion detection by using improved conditional variational autoencoder and deep neural network," *Sensors (Basel, Switzerland)*, vol. 19, 2019.

[64] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," vol. 3, 12 2015.

[65] C. Yin, Y. Zhu, J. Fei, and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, pp. 21954–21961, 2017.

[66] Q. Abu Al-Haija and S. Zein-Sabatto, "An efficient deep-learning-based detection and classification system for cyber-attacks in iot communication networks," *Electronics*, vol. 9, no. 12, 2020.

[67] C. Kolias, G. Kambourakis, A. Stavrou, and S. Gritzalis, "Intrusion detection in 802.11 networks: Empirical evaluation of threats and a public dataset," *IEEE Communications Surveys Tutorials*, vol. 18, pp. 184–208, 2016.

[68] V. L. L. Thing, "Ieee 802.11 network anomaly detection and attack classification: A deep learning approach," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, 2017.

[69] A. Ahmim, L. Maglaras, M. A. Ferrag, M. Derdour, and H. Janicke, "A novel hierarchical intrusion detection system based on decision tree and rules-based models," in *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pp. 228–233, 2019.

[70] L. Zhiqiang, L. Zhijun, G. Ting, S. Yucheng, and M.-U.-D. Ghulam, "A three-layer architecture for intelligent intrusion detection using deep learning," in *Proceedings of Fifth International Congress on Information and Communication Technology* (X.-S. Yang, S. Sherratt, N. Dey, and A. Joshi, eds.), (Singapore), pp. 245–255, Springer Singapore, 2021.