INTRUSION DETECTION SYSTEM USING FEATURE SELECTION

A Project Report

Submitted by

Himanshu Chopra **111508074**

Vaibhav Murudkar **111508078**

in partial fulfillment for the award of the degree

of

B.Tech

Information Technology

Under the guidance of

Rahul B. Adhao

College of Engineering, Pune

DEPARTMENT OF COMPUTER ENGINEERING

AND

INFORMATION TECHNOLOGY,

COLLEGE OF ENGINEERING, PUNE-5

April, 2019

DEPARTMENT OF COMPUTER ENGINEERING

AND

INFORMATION TECHNOLOGY,

COLLEGE OF ENGINEERING, PUNE

CERTIFICATE

Certified that this project, titled "INTRUSION DETECTION SYSTEM USING FEATURE SELECTION" has been successfully completed by

Himanshu Chpora **111508074**

Vaibhav Murudkar **111508078**

and is approved for the partial fulfillment of the requirements for the degree of "B.Tech. Information Technology".

SIGNATURE

Rahul B. Adhao
Project Guide
Department of Computer Engineering and Information Technology, College of Engineering Pune, Shivajinagar, Pune - 5.

SIGNATURE

Dr. Vahida Attar
Head
Department of Computer Engineering and Information Technology, College of Engineering Pune, Shivajinagar, Pune - 5.

Abstract

As advances in networking technology help to connect the distant corners of the globe and as the Internet continue to expand its influence as a medium for communications and commerce, the threat from spammers, attackers and criminal enterprises has also grown accordingly. It is the prevalence of such threats that has made intrusion detection system join ranks with firewalls as one of the fundamental technologies for network security. Organizations host and store huge sets of data for security, troubleshooting and maintaining resources. Only a few of these features are relevant in identifying security threats and attacks. This causes a need of training model to selectively identify these features and decrease resources required for the operation. In machine learning and statistics, feature selection is the process of selecting a subset of relevant features (variables, predictors) for use in model construction. Feature selection techniques can be implemented on selected datasets to simplify the model and make it easier to interpret by researchers/users, to reduce the training and threat detection time in model and to avoid the curse of dimensionality in these datasets. Feature selection can improve classification accuracy and decrease the computational complexity of classification. The central premise when using a feature selection technique is that the data contains some features that are either redundant or irrelevant, and can thus be removed without incurring much loss of information. Redundant and irrelevant features in data have caused a long-term problem in network traffic classification. These features not only slow down the process of classification but also prevent a classifier from making accurate decisions, especially when coping with big data. In this project, we propose a method that analytically selects the optimal features for classification using Rough Set Theory and Genetic Algorithms.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

An intrusion detection system continually compares recent activity to known intrusion scenarios in order to ensure that attackers are not attempting to exploit known vulnerabilities. Misuse detection identifies intrusions by matching observed data with predefined descriptions of intrusive behavior.

1.1 INTRUSION DETECTION SYSTEM

An intrusion detection system gathers and analyzes information from various areas within a computer or a network to identify possible security breaches or threats. Intrusion Detection Systems (IDS) are capable of performing different functions such as monitoring and analysis of user as well as system activities, analysis of system configuration and vulnerabilities, analysis of abnormal activity patterns, recognizing patterns of typical attacks, tracking of user policy violations, etc. An intrusion detection system is a monitoring entity that complements the static monitoring abilities of a firewall. An intrusion detection system monitors traffic in a network in promiscuous mode, very much like a network sniffer. The network packets that are collected are analyzed for rule violations by a pattern recognition algorithm. IDS are based on the knowledge of system vulnerabilities and known attack patterns (sig-

natures). These systems are widely used and are capable of detecting known attack patterns. Misuse detection is concerned with finding intruders who are attempting to break into a system by exploiting some known vulnerabilities.

1.2 FLOW BASED INTRUSION DETECTION

Flow based intrusion detection systems take network flow records as input and classify the traffic as normal or malicious[1]. A network flow or a flow is defined as a set of packets or frames passing an observation point in the network during a certain time interval. All packets belonging to a particular flow have a set of common properties.

1.2.1 Motivation

In recent years, the rapid spread of the Internet, the number of devices, and the communication tools have created large amounts of data that have been stored without any clear potential use. Traditionally, intrusion detection systems use deep packet inspection to detect attacks in the network traffic. Deep packet inspection is difficult to implement when network traffic is encrypted. Also, inspecting the complete payload is computationally expensive and can become a performance bottleneck in high-speed IP networks. Due to limitations of the packet and protocol-based IDS, researchers are focusing on alternative approaches to protect IP networks. A solution for securing IP networks from unauthorized access is the use of flow-based intrusion detection. The observation points in the network can be flow enabled network devices. The information of network flow is stored in a flow record.

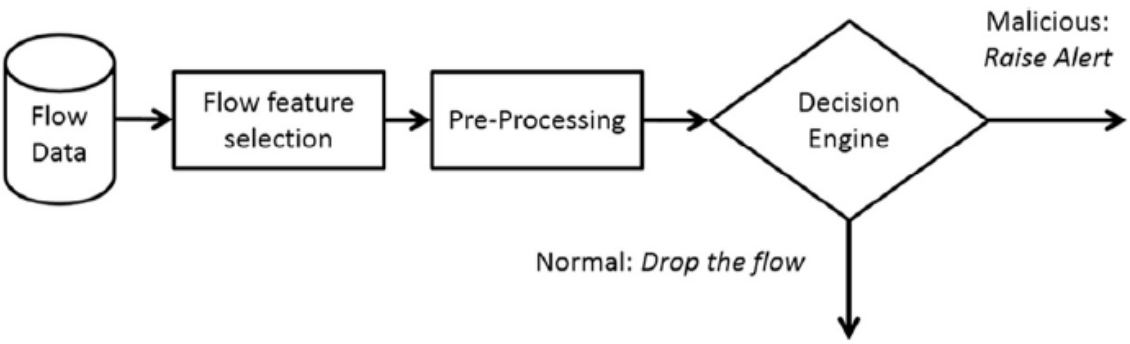

Figure 1.1: Flow based detection overview

1.2.2 Architecture of flow based intrusion detection system

The system takes flow records as input. Not all of the attributes of flow records will be required in the attack detection decision. The feature selection phase only selects relevant attributes required for decision making[1]. A pre-processing / Metering phase converts the flow records in an acceptable format. It is shown in figure 1.1. In the detection phase, the algorithm marks the flow records as malicious or normal. If the flow is normal, it is considered safe and dropped with no subsequent action while malicious flow can raise an alert and become the subject of further inspection.

1.2.3 Pros and cons of flow-based intrusion detection

Flow-based intrusion detection has a number of advantages over traditional intrusion detection systems. The flow-based IDS only analyzes network flow records[1]. The flow records contain aggregated information of packet headers. The network traffic information is summarized in the form of IP flows and the amount of data processed by the IDS is reduced. Flow-based intrusion detection is, therefore, best suited for protection of backbone links where processing of complete network traffic is computationally difficult.

Flow-based intrusion detection also has some drawbacks. The IP flow records used for intrusion detection contain generalized network information. It is therefore difficult for the flow-based IDS to distinctly detect an attack using the generalized information. Flow-based techniques do not scan the packet payload. Therefore flow-based techniques cannot detect the network attacks hidden in the packet payload and are not as accurate as packet-based detection.

1.3 FEATURE SELECTION

Feature selection (FS) is a part of dimensional reduction which is known as the process of choosing an optimal subset of features that represents the whole dataset[2].

1.3.1 Introduction

FS has been used in many fields, such as classification, data mining, object recognition and so forth, and has proven to be effective in removing irrelevant and redundant features from the original dataset. Given a feature set, the FS problem tries to find a minimal feature subset (fewer attributes) that enables the construction of the best classifier with high accuracy without loss of much information.

1.3.2 Feature search

Feature search is a strategy that determines the optimal subset of features. The optimal solution can be found through an exhaustive search or the branch and bound algorithm, sequential search methods (e.g. sequential forward selection, sequential backward elimination, and bidirectional selection), and

random search methods (e.g. genetic algorithms and random mutation hill climbing).

1.3.3 Feature selection methods

Feature selection methods can be classified into two approaches[3]: individual evaluation and subset evaluation. Individual feature evaluation assesses each feature individually according to its relevance which leads at the end to a feature ranking. The drawback of individual evaluation is that it is incapable of eliminating redundant features because they have the same rank. Differently, subset feature evaluation can overcome the inconvenience of individual evaluation, it uses certain search strategies to select and evaluate a subset of features according to certain evaluation measures and then compares it with the previous best one[4]. From this classification, three main approaches can be identified based on the relationship among the inductive learning method and the feature selection algorithm[3]: filters, wrappers, and embedded methods

Filter methods

Filter methods, which are more common in statistics, are feature selection algorithms totally independent from any predictors. Applied directly on the training data, the filter approach is based on feature ranking techniques that use an evaluation criterion and a threshold to determine the feature relevance and decide whether to keep it or discard it. Filter algorithms are usually computationally less expensive than the other methods. A common drawback for filter methods is that they are adequate only for independent features; otherwise, features will be redundant.

Wrapper methods

Distinct from the filter methods, wrapper methods are feature selection based on three components: a search strategy, a predictor, and an evaluation function. The search strategy determines the subset of features to be evaluated. The predictor (considered as a black box) can be any classification method and its performance is used as the objective function to evaluate the subset of features defined by the search strategy so as to find the optimum subset that gives the best accuracy of it. The wrapper approach outperforms the filter approach but it is more time consuming and requires more computational resources.

Embedded methods

In contrast to wrapper methods, embedded methods incorporate an interaction between feature selection and learning process. Therefore, the solution is reached faster than wrappers because they make better use of the available data and avoid retraining the predictor for every selected feature subset.

1.4 ROUGH SET THEORY

Rough Set Theory [5] is a mathematical tool to deal with imprecise and insufficient knowledge.In rough set theory, membership is not the primary concept unlike fuzzy sets. It deals with inconsistency, uncertainty, and incompleteness by imposing an upper and a lower approximation to set membership. The advantage of rough set theory is that it does not require any preliminary or additional information about data like probability in statistics or grade of membership or the value of possibility in fuzzy set theory. In-discernibility relation is a central concept in Rough Set theory. In-discernibility relation is

an equivalence relation, where all identical objects of a set are considered as elementary. In this approach, the uncertainty and imprecision are expressed by a boundary region of a set not by a partial membership.

Let (X, A) be an information system where X is the universe of discourse and A is a non-empty finite set of attributes such that $a : X \to V$, a for every $a \in A$ .The set V a is called the "value set of a".

Given $B \subseteq A$ there is an associated equivalence relation R_B[5] :

$$R_B = (x, y) \in X_2 \mid \forall\, a \in B,\ a(x) = a(y)$$

If $(x, y) \in R_B$, then x and y are indiscernible by attributes from B. The equivalence classes of the B-indiscernibility relation are denoted by [x]B .

Let A be a subset of X so that A can be approximated using the information contained within B by constructing the B-lower and B-upper approximations of A as denoted by

$$R_B \downarrow A = x \in X \mid [x]B \subset A \ \ and$$

$$R_B \uparrow A = x \in X \mid [x]B \cap A = \phi$$

The tuple $(R_B \downarrow A, R_B \uparrow A)$ is called a rough set.

1.5 GENETIC ALGORITHM

Genetic Algorithm (GA) was first proposed by Professor Holland in the University of Michigan of the United States[6]. This is a stochastic method for function optimization based on the mechanics of natural genetics and biological evolution. It is an adaptive heuristic search technique for finding global optimal solution. It simulates genetic and evolutionary process of natural evolution. It uses the search technique which is not along a single direction of search space. It considers a number of individual solutions and tests for

convergence within the overall scope of the search space, thus it has greater possibility of finding the global optimal solution. It is very useful for solving optimization problems as it works properly even if the input parameters are slightly changed, or even in the presence of reasonable noise. Also, the method offers significant benefits while searching for a solution in a large state-space, multi-modal state-space, or n-dimensional surface, over more other search of optimization techniques like linear programming, depth-first, breath-first and so on. Genetic algorithms operate on a population of individuals to produce better and better approximations[6]. At each generation, a new population is created by the process of selecting individuals according to their level of fitness in the problem domain, and recombining them together using operators borrowed from natural genetics. The offspring might also undergo mutation. This process leads to the evolution of populations of individuals that are better suited to their environment than the individuals that they were created from, just as in natural adaptation.

1.5.1 Notion of natural selection

The process of natural selection starts with the selection of fittest individuals from a population. They produce offspring which inherit the characteristics of the parents and will be added to the next generation. If parents have better fitness, their offspring will be better than parents and have a better chance at surviving. This process keeps on iterating and at the end, a generation with the fittest individuals will be found. This notion can be applied for a search problem. We consider a set of solutions for a problem and select the set of best ones out of them.

1.5.2 Concept of GA based optimization

The GA based optimization starts with a population of randomly generated chromosomes where each chromosome represents a candidate solution of the concrete problem being solved.[6] In each generation, the fitness of each chromosome is evaluated and the more fitted solutions are selected to form a mating pool. The most used method for fitness assignment is known as "Rank based". With this method, the selection errors of all the individuals are sorted. Then, the fitness assigned to each individual only depends on its position in the individuals rank and not on the actual selection error. The fitness value assigned to each individual with the Rank based method will be: (i) = R(i) i = 1,...,N. Here is a constant called selective pressure, and its value is fixed between 1 and 2. Greater selective pressure values will make the fittest individuals to have more probability of recombination. The parameter R(i) is the rank of individual i. Two parents are randomly selected from the pool, and undergo cycle crossover and mutation to form two offspring. This process of selection, crossover and mutation is repeated until the new population is generated. One of the most used selection methods is roulette wheel, also called stochastic sampling with replacement. This method places all the individuals on a roulette, with areas proportional to their fitness. Then, the roulette is turned and the individuals are selected at random. The corresponding individual is selected for recombination. Now, In order to decide if a feature will be mutated, we generate a random number between 0 and 1. If this number is lower than a value called the mutation rate, that variable is flipped. The mutation rate is usually chosen to be 1/m, where m is the number of features. With that value for the mutation rate, we will mutate one feature of each individual (statistically). The mutation operation is helpful to avoid premature convergence and to explore broader search

space. Thus, the process searches for the better solutions in each generation and is continued until the population converges to a globally optimal solution in the solution space. The overall process continues until either a predefined number of generations are completed, stagnation, or termination criteria are satisfied. Configurable parameters in the implementation include termination criterion, tournament size to select parents, crossover probability, and mutation probability. There are two types of genetic algorithm[1], in single objective GA the real world optimization problem is built involving only one objective function for finding the unique optimal solution. The main goal of using single objective genetic algorithm is to find the optimal solution, which corresponds to the optimum value of the single objective function. On the other hand in multi objective GA the optimization problem involves more than one competing or conflicting objective functions for finding many optimal solutions. Most of the real-world optimization problems involve multiple objectives, and if they are conflict in nature then there may be a number of pareto optimal solutions instead of a single optimal solution. Generally, all pareto optimal solutions are treated as equally good and the goal may be to find a representative set of pareto optimal solutions or finding a single solution by the decision maker based on the application[7].

Chapter 2

Literature Survey

We try to put together all the works we have studied in this section in brief in sections.

2.1 INTRUSION DETECTION

One of the earliest work that proposed intrusion detection by identifying abnormal behavior can be attributed to Anderson [8]. In his report, Anderson presents a threat model that classifies threats as external penetrations, internal penetrations, and misfeasance, and uses this classification to develop a security monitoring surveillance system based on detecting anomalies in user behavior. External penetrations are defined as intrusions that are carried out by unauthorized computer system users; internal penetrations are those that are carried out by authorized users who are not authorized for the data that is compromised; and misfeasance is defined as the misuse of authorized access both to the system and to its data. Recent IDS uses misuse detection, which is a technique for intrusion detection that relies on a predefined set of attack signatures. Analysing for specific patterns, the signature detection-based intrusion detection systems try to match incoming packets to the signatures of known attacks. Thus, decisions are made based on the knowledge acquired

from the model of the intrusive process and the observed trace that it has left in the system.

Legal or illegal behavior can be defined and compared with observed behavior. Such a system tries to collect evidence of intrusive activity irrespective of the normal behavior of the system. One of the chief benefits of using signature detection is that known attacks can be detected reliably with a low false positive rate. If the audit data in the log files do not contain the attack signature, no alarm is raised. Another benefit is that the signature detection system begins protecting the computer/network immediately upon installation. One of the biggest problems with signature detection systems is maintaining state information of signatures in which an activity spans multiple events, the complete attack signature spans multiple packets. Another drawback is that the signature detection system must have a signature defined for all of the possible attacks that an attacker may launch. This requires signature updates to keep the signature database up-to-date.

2.2 FEATURE SELECTION

As we have already seen what is feature selection we will try to emphasis on existing works on feature selection.

2.2.1 Literature

William et al. [9] found that feature selection using correlation-based feature selection can greatly improve computational performance, and meanwhile the classification accuracy is not significantly degraded. Moore et al. [10] used FCBF feature selection method to filter the redundancy features and evaluated the classification performance (accuracy metric) using Nave Bayes for

searching an optimal number of features. Zander et al. [11] used wrapper feature selection method, sequential forward selection, which selects features by evaluating the performance of classifier using datasets characterized by feature subset. Ultimate goal is to select an optimal feature subset for this classifier. Dai L. et al. [12] proposed ChiSquared-C4.5 feature selection method for Internet traffic classification. It filters out a feature subset using Chi-squared. These two methods are wrapper feature selection methods.The previous feature selection methods (especially the wrapper feature selection methods) lead to the multi-class imbalance problem. Because that their performance metrics (accuracy or intra-class homogeneity) used for classifiers are dependent on the prior distribution. And, the classifiers will be overwhelmed by majority classes. En-Najjary et al.[13] build logistic regression model for every application class, and select a feature subset for each logistic regression model using parameter estimation. They handle the class imbalanced problem through transferring the multi-class classification into two-class classification. Another method was proposed using machine learning by Liu zhen [14]. But it is not computationally efficient. NSGA II implemented the feature selection problem referring to a simple coding scheme and used binary chromosomes to present even the feature was selected or not. Two competing objectives were : the minimization of the number of used features and the classification error. The classification error was computed on each database using the 1-NN classifier to evaluate the discrimination value of the each selected set. In this they used 1-NN as classifier to evaluate the performance of a given feature set identified by a chromosome on different training and testing sets. The 1-NN classifier returns a real variable belonging to (0,1) and denoting the classification error on a testing data set. This improvement was implemented by means of the integration of a data structure completing

the chromosome encoding containing the indexes of the selected sequence of features. This saves the computational time of the whole classification process. The NSGA II was recently implemented by Deb to improve the first version of the algorithm. Deb has show that the new algorithm outperforms the first version of NSGA and has a lower computational complexity. In table 2.1, we shown the key features and limitations of different feature selection approaches used recently. Few of them are compared in the table.

Table 2.1: Comparison of different approaches for feature selection

Name of algorithm	Key features	Limitations
FSMSD[15]	Decomposing the feature space according to the selected categorical features to improves performance Handle both binary and multi-class mixed-type classification problems by selecting a subset of features	High-dimensional mixed-type datasets used in this paper is the biomedical datasets, originally only included numerical features It was slow
Roughinement QuickReduct[16]	RQR is able to select a subset of features that allows to obtain at least the same accuracy obtained with the unreduced dataset	Lot of work is still has to be done using this type of approach
Genetic algorithm in MapReduce[17]	Investigated the impact of the number of attributes and the number of instances on the sequential and MapReduce implementations MapReduce implementation has consistent results as we move towards higher dimensional spaces	Only domain specific work has done
GA-LR wrapper[18]	The selection of the best subset is based on maximising the accuracy of classification and minimising the number of features Three decision tree classifiers namely C4.5, RF and NBTree to evaluate the generated subsets of features	Used only for single objective problem

2.2.2 Feature selection tools

WEKA 3[19] : Weka is a free software written in Java and developed at the University of Waikato, New Zealand. It integrates most of the machine learning and data mining techniques used for knowledge discovery. Despite its ability to perform wrapper approaches for feature selection, Weka presents a weakness which is the process time. Features : Data Pre-processing, Data Classification, Data Regression, Data Clustering, Data Association Rules, Attribute Selection, Data Visualization

ROSE2 : It provides functions like data pre-processing, including discretization of numerical attributes, performing a standard and an extended rough set based analysis of data, search of a core and reduction of attributes permitting data reduction, inducing sets of decision rules from rough approximations of decision classes, evaluating sets of rules in classification experiments using sets of decision rules as classifiers.

ROSETTA[20] : One of the majorly used tool for feature selection. It provides partial integration with DBMSs via ODBC. Exporting of rules, reduction, tables, graphs and other objects to various formats, including XML, C++ and Prolog. Toolkit for analyzing tabular data within the framework of rough set theory. Features - Pre-processing, Computation, Post-processing, Validation and analysis.

SCIKIT Python : Built on NumPy, SciPy, and matplotlib. Features : Classification, Regression, Clustering, Dimensionality, reduction and Model selection, Pre-processing.

RSESLIB : Rough set and machine learning open source in JAVA, Available in WEKA. Functionalities are Classification, Regression, Clustering, Computation, Post-processing.

DEAP : DEAP is a novel evolutionary computation framework for rapid

prototyping and testing of ideas. It seeks to make algorithms explicit and data structures transparent. It works in perfect harmony with parallelisation mechanism such as multiprocessing and SCOOP. The following documentation presents the key concepts and many features to build your own evolutions.

2.2.3 RST based data discritization

Rough set theory [5] is proposed by professor Pawlak at Warsaw University of Information Technology and Management to deal with incomplete information. It does not require any priori information, and can effectively analyze and deal with incomplete, inconsistent, inaccurate data. It defines the knowledge from a new perspective: taking the knowledge as the partition of universe, where the equivalence relations are used to formally express the classification. Through large amounts of data analysis, this method deletes some relative information in accordance with the relationship of the two equivalence relations in the universe and extracts potential valuable knowledge of the rules. This method has been widely used in knowledge acquisition, rule extraction, machine learning, decision analysis, pattern recognition, data mining and other fields, it is very suitable for the safety rules learning and detection. Information theory is a branch of application mathematics and electrical engineering involving the quantification of information. It was developed by Claude E. Shannon to find fundamental limits on compressing and reliable storing and communicating data. Suppose U is a universe, P,Q are the equivalence relation in it (i.e.: knowledge), according the information theory, the entropy of knowledge, P is defined as [21]:

$$H(P) = -\sum_{i=1}^{n} p(Xi) \log p(Xi)$$

The conditional entropy of knowledge Q relative to P, is defined as [21]:

$$H(Q|P) = -\sum_{i=1}^{n} p(Xi) \sum_{j=1}^{m} p(Yj|Xj) \log p(Yj|Xj)$$

Among them:

$$p(Y|X) = card(Y \cap X)/card(X)$$

Mutual Information is defined as [21]:

$$I(R, D) = H(D)H(D|R)$$

$$SGF(a, R, D) = I(R \cup a; D) - I(R; D) = H(D|R) - H(D|R \cup a)$$

According to the definition of mutual information, feature important degree is defined as above equation[21].

Chapter 3

Problem statement

In intrusion detection, the datasets used are characterized by their large amounts and their high dimensionality. Thus, it is necessary to proceed with a dimensionality reduction step to improve the classification accuracy and reduce the computational time. In high-dimensional feature space, some of the features may be redundant or irrelevant. Removing these redundant or irrelevant features is very important for effectiveness. Effective feature selection is an important research challenge for classifying intrusion data as normal or intrusive. This has tremendous impact on the performance of any intrusion detection system.

Chapter 4

Methodology

One of the toughest challenges while building intrusion detection systems is uncertainty handling and dealing with unbalanced data set. It is hard to distinguish between normal and abnormal behavior of users in a network because the boundaries cannot be well defined. Thus, use of suitable techniques to address this aspect of intrusion detection is critical. Rough Set theory has the potential to deal with such imprecision and uncertainties in discrete and noisy data. Fuzzy sets and rough sets are two important, complementary characteristics of imperfect data and knowledge. While fuzzy set has the capability to deal with incomplete, noisy, or imprecise data, rough set can deal with imperfect knowledge.

Rough sets have been applied successfully to attribute selection and rule induction in various fields. The rough set theory[5] (RST) was first introduced by Pawlak in 1982 as a tool for handling uncertainty resulting from noisy or incomplete information systems. In contrast to other methods, attribute selection based on rough set theory detects the attribute dependencies using the decision table. A core part of rough set theory is the identification of the minimum reduct (i.e. minimal subset of conditional attributes) that has similar discernibility properties of the full attribute set. This problem has been described as an optimization problem, which is known to be NP-hard. The

brute-force search approach checks the merits of all possible combinations of attributes, which is very time consuming and hence becomes ineffective as the data size grows. To tackle this problem, a heuristic or meta-heuristic search method is typically used with attribute evaluation. Among these methods is genetic algorithms (GA). However, the traditional implementations of GA do not demonstrate the full potential of GA capabilities.

The optimization problems generally have one or more feasible solutions obtained using one or more objective functions. One of the advantage of using genetic algorithms for feature selection is that they dont need specific knowledge about the problem under study. Above this they usually perform better than traditional feature selection techniques and they can manage data sets with many features. But on the other hand these algorithms can take a long time to converge, since they have a stochastic nature and might be expensive in computational terms. In order to overcome the problems of dimension reduction of a high dimensional data set, combination of Genetic Algorithm (GA) and Rough Set Theory (RST) is used for their outstanding ability of overall searching, to obtain the minimal feature subset sufficient for representing the dataset and improve convergence speed.

Table 6.4 shows convergence speeds of buildtimes for CICIDS2017 models using RST based discretization technique.

Chapter 5

System design

In order to improve detection accuracy and efficiency, a Feature Selection method based on Rough Sets and Genetic Algorithms is proposed. Firstly, the features are filtered by virtue of the Rough Sets theory; then Genetic Algorithm applied to find best subset.

This algorithm includes Four Modules: Data Discretization, Applying evolutionary algorithm, Subset Calculation and Feature Ranking. The following is detailed description of these modules:

A. Data Discretization : Using ROSETTA's Nave Scaler algorithm, the continuous features are discretized

B. Applying evolutionary algorithm : Genetic Algorithm is launched using DEAP library in python using specified configurations.

Fitness function is used for evaluating subsets in each generation.

The fitness function depends on these aspects:

a) The number of features in feature subset N , the less the higher fitness;

b) The number of features which have high feature importance-M, the larger the higher fitness; Accuracy on generated attribute subset is used for evaluation.

C. Feature Subset : Final generation of previous step generates HallOfFame feature list which are evaluated against validation set to give best subset(s)

D. Feature Ranking : Features selected through each generation are compiled and ranked.

Best individuals in HallOfFame object are evaluated to assess each feature individually according to its relevance in each generation which leads at the end to a feature ranking.

Chapter 6

Experimental setup

In this chapter we explained all the details about experimentation, datasets we used and parameters we consider for checking the performance.

6.1 EXPERIMENTAL DATA

Details about various dataset we used will be discussed in this section.

6.1.1 NSL KDD Dataset

The Security Lab-Knowledge Discovery and Data Mining (NSL-KDD) dataset is firstly used in the experiment[22]. In the NSL-KDD, redundant records have been removed and attacks are labeled and sorted into different levels by detection difficulty. Also, selected record numbers from each level is inversely proportional to the percentage of these records in the NSL-KDD dataset. Taking into account these characteristics, NSL-KDD can be considered a good approximation of known attacks. It is a reduced version of the original KDD 99 data set. The data set consists of 41 feature attributes out of which 38 are numeric and 3 are symbolic. Total number of records in the data set is 125,973 out of which 67,343 are normal and 58,630 are attacks. The data set contains different attack types that can be classified

into following categories : Remote to Local, Denial of Service, User to Root and Probing.

6.1.2 CICIDS2107 Dataset

CICIDS2017 dataset [23] contains benign and the most up-to-date common attacks, which resembles the true real-world data (PCAPs). It also includes the results of the network traffic analysis using CICFlowMeter with labeled flows based on the time stamp, source and destination IPs, source and destination ports, protocols and attack (CSV files).

Since CICIDS2017 is intended for network security and intrusion detection purposes, it should cover a diverse set of attack scenarios. In this dataset, we observe six attack profiles based on the last updated list of common attack families as :

Brute Force Attack: This is one of the most popular attacks that only cannot be used for password cracking, but also to discover hidden pages and content in a web application. It is basically a hit and try attack, then the victim succeeds.

Heartbleed Attack: It comes from a bug in the OpenSSL cryptography library, which is a widely used implementation of the Transport Layer Security (TLS) protocol. It is normally exploited by sending a malformed heartbeat request with a small payload and large length field to the vulnerable party (usually a server) in order to elicit the victims response.

Botnet: A number of Internet-connected devices used by a botnet owner to perform various tasks. It can be used to steal data, send spam, and allow the attacker access to the device and its connection.

DoS Attack:The attacker seeks to make a machine or network resource unavailable temporarily. It typically accomplished by flooding the targeted

machine or resource with superfluous requests in an attempt to overload systems and prevent some or all legitimate requests from being fulfilled.

DDoS Attack: It typically occurs when multiple systems, flood the bandwidth or resources of a victim. Such an attack is often the result of multiple compromised systems (for example, a botnet) flooding the targeted system with generating the huge network traffic.

Web Attack: This attack types are coming out every day, because individuals and organizations take security seriously now. We use the SQL Injection, which an attacker can create a string of SQL commands, and then use it to force the database to reply the information, Cross-Site Scripting (XSS) which is happening when developers dont test their code properly to find the possibility of script injection, and Brute Force over HTTP which can tries a list of passwords to find the administrators password.

Infiltration Attack: The infiltration of the network from inside is normally exploiting a vulnerable software such as Adobe Acrobat Reader. After successful exploitation, a backdoor will be executed on the victims computer and can conduct different attacks on the victims network such as IP sweep, full port scan and service enumerations using Nmap.

6.1.3 CICIDS2017 Panigrahi Dataset

Various shortcomings of the CICIDS[23] have been studied and outlined; Solutions to counter the following issues have been provided under this dataset. Shortcomings are as follows :

Scattered Presence : We have seen in table 6.1 that the data of CICIDS2017 dataset is present scatter across eight files. Processing individual files are a tedious task. Therefore, we combined those files to form a single file that contains a total of 3119345 instances of all the files

Table 6.1: Description of files containing CICIDS2017 dataset

Name of Files	Day Activity	Attacks Found
Monday-WorkingHours.pcap_ISCX.csv	Monday	Benign (Normal human activities)
Tuesday-WorkingHours.pcap_ISCX.csv	Tuesday	Benign, FTP-Patator, SSH-Patator
Wednesday-workingHours.pcap_ISCX.csv	Wednesday	Benign, DoS GoldenEye, DoS Hulk, DoS Slowhttptest, DoS slowloris, Heartbleed
Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX.csv	Thursday	Benign, Web Attack Brute Force, Web Attack Sql Injection, Web Attack XSS
Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX.csv	Thursday	Benign, Infiltration
Friday-WorkingHours-Morning.pcap_ISCX.csv	Friday	Benign, Bot
Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv	Friday	Benign, PortScan
Friday-WorkingHours-Afternoon-DDos.pcap_ISCX.csv	Friday	Benign, DDoS

Huge Volume of Data : The combined dataset contains data of all the possible recent attack labels at one place. But, at the same time the size of a combined dataset becomes huge. This huge volume of data itself becomes a shortcoming, it consumes more overhead for loading and processing.

Missing Values : The combined CICIDS2017 dataset contains 288602 instances having missing class label and 203 instances having missing information. These unwanted instances have been removed to form a dataset that contains unique 2830540 instances.

High Class Imbalance : High class imbalance is a situation in a dataset where if the dataset is used for training of a classifier or detector, in such a case the detector biased towards the majority class. As a result, the detector shows lower accuracy with higher false alarm. This is a major drawback in CICIDS2017 dataset.

The problem of scattered presence has been handled by combining various

data files if CICIDS2017. Also, the missing values has also been removed. Though, huge volume of data is a short-coming for a dataset but at the same time it is inherent to any typical dataset that contains typical information. This shortcoming of high volume can be overcome by sampling the dataset before actual detection process starts.

Table 6.2: Characteristics of attack labels with prevalence rate in dataset

New Labels	Old Labels	Instances	% of prevalence
Normal	Benign	2359087	83.34
Botnet ARES	Bot 1966	0.06	
Brute Force	FTP-Patator, SSH-Patator	13835	0.48
Dos/DDos	DDoS, DoS GoldenEye, DoS Hulk, DoS Slow-httptest, DoS slowloris, Heartbleed	294506	10.4
Infiltration	Infiltration	36	0.001
PortScan	PortScan	158930	5.61
WebAttack	Web AttackBrute Force,Web AttackSql Injection,Web Attack-XSS	2180	0.07

6.2 PERFORMANCE METRICS

An intrusion detection model can be evaluated by its ability to make accurate prediction of attacks. Intrusion detection system mainly discriminate between two classes, normal class and attack class. The confusion matrix reports True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

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

TN (True Negative): The number of legitimate (normal) records that are

Table 6.3: Confusion Matrix

		Predicted Class: Normal	Predicted Class: Attack
Actual Class	Normal	TN	FP
	Attack	FN	TP

correctly classified.

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

FN (False Negative): The number of records that are incorrectly classified as legitimate activities though those are actually malicious.

The proposed model is evaluated based on these performance measures : Accuracy Rate, Precision, Recall (True Positive Rate)

Accuracy Rate(AR) measures the fraction of all instances that are correctly categorized; it is the ratio of the number of correct classifications to the total number of correct or incorrect classifications.

$$AR = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision is the fraction of relevant instances among the retrieved instances.

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

Recall / True Detection Rate (TDR) measures the proportion of actual positives that are correctly identified.

$$Recall = \frac{TP}{TP + FN}$$

Higher values of Accuracy, Precision, Recall show better classification performance for IDSs.

6.3 FEATURE SELECTION PROCESS

6.3.1 Experimental Configuration

Experimental configuration is as follows: Minimum threshold value for feature importance is 0.0001; Size of initial population is 100; Number of Generations is 10; Mutation rate is 0.2; Crossover rate is 0.5.
Experimental environment is as follows: Intel Core i7-6700HQ; CPU 2.60GHz*8; OS type 64-bit; 8 GB Memory; Ubuntu 16.04 LTS.

6.3.2 Result Analysis

NSL KDD Dataset

The data set consists of 41 feature attributes out of which 38 are numeric and 3 are symbolic. Total number of records in the data set is 125,973 out of which 67,343 are normal and 58,630 are attacks.

Table 6.4: Feature Selection on NSLKDD

Dataset	Features	Detection-time(secs)	Accuracy-Rate(%)	Precision(%)	Recall(%)
Original-Dataset	41	3.76	86.55	81.63	64.48
Reduced-Subset	23	1.88	91.26	92.85	82.29

Model-Build-Time : 18.01 secs

Best Informative Features : count, num_outbound_cmds, serror_rate, service, same_srv_rate, num_shells, duration, dst_host_same_srv_rate, num_compromised, rerror_rate.

CICDS2017

The dataset is spanned over eight different files containing five days normal

and attacks traffic data of Canadian Institute of Cybersecurity. The whole shape of the dataset contains 3119345 instances and 83 features containing 15 class labels (1 normal + 14 attack labels).

Table 6.5: Model Build Time using GA and RST+GA in secs

Dataset	BuildTime-GA	BuildTime-RSTGA
Tuesday-WorkingHours.pcap_ISCX.csv	1886.36	1291.82
Wednesday-workingHours.pcap_ISCX.csv	2109.01	1785.40
Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX.csv	1929.89	1317.93
Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX.csv	1562.01	1267.206
Friday-WorkingHours-Morning.pcap_ISCX.csv	1436.78	1182.06
Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv	1554.89	1172.96
Friday-WorkingHours-Afternoon-DDos.pcap_ISCX.csv	1552.48	1251.49

Table 6.6: Feature Selection on CICDS2017

	Features	Test-time(secs)	Test-Time-Subset(secs)	Accuracy-Rate(%)	Precision(%)	Recall(%)
Tuesday	30	167.43	88.00	94.93	96.41	89.50
Wednesday	42	199.49	142.58	97.54	90.94	84.44
Thursday-Webattacks	38	97.35	57.54	99.65	90.43	92.83
Thursday-Infiltration	47	147.59	107.42	99.99	100.00	100.00
Friday-MWH	42	91.64	39.33	99.87	91.78	98.32
Friday-PortScan	43	103.14	51.32	99.87	88.12	95.32
Friday-DDos	43	104.43	43.35	99.84	92.45	98.43

Comparative Study

To evaluate our proposed model, we compare it with some well known classifiers and some recent ones namely J48, Naive Bayes, Random Forest. Table 6.6 summarizes the performance of our model compared to the other classifiers. Our model gives the highest accuracy for two of the files (Wednesday-WH, Thursday-Infiltration), moreover our model is very close to the highest rate for two file performances namely Friday-PortScan and Friday-DDos. For the rest, our model gives an average performance compared to the other models.

Table 6.7: Performance of our model and other classifiers on Accuracy Rate %

	Our Model	J48	Random Forest	Naive Bayes
Tuesday-WH	96.93	99.72	95.54	99.45
Wednesday-WH	97.54	93.75	93.87	82.67
Thursday-WebAttacks	91.65	92.408	98.48	93.87
Thursday-Infilteration	99.99	96.66	83.33	83.33
Friday-WHM	99.17	99.54	98.72	99.18
Friday-PortScan	99.87	98.56	99.88	98.49
Friday-DDos	98.84	99.78	99.81	93.87

Panigrahi CICIDS2017

Shortcomings of CICIDS2017 are eliminated by combining dataset of CICIDS2017 files for better classification and detection of any future intrusion detection engine.

Table 6.8: Feature Selection Results

Dataset	Features	Detection-time(secs)	Accuracy-Rate(%)	Precision(%)	Recall(%)
Original-Dataset	85	1744	90.60	90.05	90.05
Reduced-Subset	36	1044	93.56	96.28	96.28

Our model generates reduced feature subset with higher accuracy, precision and recall rates on combined dataset.

Chapter 7

Conclusion

An approach for feature selection based on rough set theory and genetic algorithm has been presented. Handled dataset contains continuous/real valued and discrete feature attributes. Discretizing the real valued attributes, in order to obtain a dataset composed only by crisp values, improves convergence of evolutionary algorithm applied in place. The genetic algorithm repeatedly modifies a population of individual solutions and over successive generations, the population evolves toward an optimal solution. The results of experiments show that the feature selection algorithm is better at Accuracy rate, Precision rate and Detection rate than classical classifiers, meanwhile the number of features is less. Based on observations of individual solutions over the generations, attributes are ranked according to their selection probability naively. Building more rigorous mathematical formulas to design more reliable and faster converging model will be the work forward.

Chapter 8

References

[1] Muhammad Fahad Umer, Muhammad Sher, Yaxin Bi. "Flow-based intrusion detection: Techd challenges". Computers & security 70-238254, 2017.

[2] S. Theodoridis, K. Koutroumbas. "Pattern Recognition". Academic Press, 2009.

[3] K.M.Shazzad, J.S.Park."Optimization of Intrusion Detection through Fast Hybrid Feature Selection", Proceedings of the Sixth International Conference on Parallel and Distributed omputing, Applications and Technologies (PDCAT05), 2005.

[4] D.Koller, M.Sahami. "Toward optimal feature selection", Proceedings of the international Conference on Machine Learning , 1996.

[5] Pawlak.Z, "Rough Sets and Intelligent data analysis", Computer and Information Science, 2002.

[6] Holland.J.H, "Adaptation in Natural and Artificial Systems". University of Michigan Press, Ann Arbor, 1975.

[7] Asit K. Das, Shampa Sengupta,ha Bhattacharyya. "A group incremental feature selection for classification using roughset theory based genetic

algorithm". Applied Soft Computing 65-400411, 2018.

[8] D. Anderson, T. Frivold, A. Tamaru, and A. Valdes."Next generation intrusion detection expert system (nides), software users manual". Computer Science Laboratory, SRI International, 333 Ravenswood Avenue, Menlo Park, CA 94025-3493, User Manual SRICSL9507, 1994.

[9] Nigel Williams, Sebastian Zander, Grenville Armitage. "A preliminary performance comparison of five ML Algorithms for practical IP traffic flow classification". ACM SIGCOMM Computer Communication Review, Vol.36, No.5, pp. 5-16, 2006.

[10] A.W.Moore, D.Zuev. "Internet traffic classification using Bayesian analysis techniques". Proc. of the 2005 ACM SIGMETRICS international confe7rence on Measurement and Modeling of Computer Systems (SIGMETRICS 05), pp.50-60, 2005.

[11] S. Zander, T. Nguyen, G. Armitage. "Automated Traffic Classification and Application Identification using Machine Learning". Proc. of the IEEE 30th Conference on Local Computer Networks (LCN 05), pp.250-257, 2005.

[12] Dai Lei, Yun Xiaochun, Xiao Jun. "Optimizing Traffic Classification Using Hybrid Feature Selection". Proc. of the Ninth International Conference on Web-Age Information Management (WAIM 08), pp. 520-525,2008.

[13] Najjary Taoufik, Urvoy Keller Guillaume, Pietrzyk Marcin, Costeux Jean Laurent. "Application based feature selection for internet traffic classification". Proc. of 22nd International Teletraffic Congress (ITC 2010), 2010.

[14] K. Xu, Z. L. Zhang, S. Bhattacharyya."Internet traffic behavior profiling for network security monitoring". IEEE/ACM Trans. Networking, Vol.16, No. 6, pp. 1241-1252, 2008.

[15] Kyung Jun Kim, Chi-Hyuck Jun. "Rough set model based feature selection for mixed-type data with feature space decomposition". Expert Systems With Applications 103-196205, 2018.

[16] Alessio Ferone. "Feature selection based on composition of rough sets induced by feature granulation". International Journal of Approximate Reasoning 101-276292, 2018.

[17] El Sayed M. El Alfy, Mashaan A. Alshammari. "Towards scalable rough set based attribute subset selection for intrusion detection using parallel genetic algorithm in MapReduce". Simulation Modelling Practice and Theory 64-1829, 2016.

[18] Chaouki Khammassi, Saoussen Krichen. "A GA-LR wrapper approach for feature selection in network intrusion detection". Computers & security 70-255277, 2017.

[19] WEKA SOFTWARE. https://www.cs.waikato.ac.nz/ml/index.html . Last accessed on: 05/03/2019.

[20] ROSETTA SOFTWARE. Http://tosetta.kb.uu.se/general . Last accessed on: 05/03/2019.

[21] Yanhuai Ma. "Data Mining Methods based on Rough Set Theory ". PhD thesis, Chinese Academy, 2003.

[22] KDD CUP 1999 DataSet. http://kdd.ics.uci.Edu/databases/kddcup99/task.html . Last accessed on: 08/03/2019.

[23] Ranjit Panigrahi, Samarjeet Borah. "A detailed analysis of CICIDS2017 dataset for designing intrusion detection system". International Journal of Engineering & Technology, 7 (3.24) 479-4, 2018.

[24] Han Jiawei, Micheline Kamber. "Data mining: concepts and techniques". 2nd ED. Beijing:China Machine Press, 2006.

[25] C.J.C.Burges. "A tutorial on support vector machines for pattern recognition". Data Mining and knowledge Discovery, 2(2):121-167, 1998.

[26] Iman Sharafaldin, Arash Habibi Lashkari, Ali A. Ghorbani, "Intrusion Detection Evaluation Dataset (CICIDS2017)", Canadian Institute of Cybersecurity, http://www.unb.ca/cic/datasets/ids-2017.html, Last accessed on: 13/04/2019.

[27] Depren O., Topallar M., Anarim E. & Ciliz M. K. "An intelligent intrustion system for anomaly and misuse detection in computer networks". Expert Systems with Applications, 29(4), 713722, 2005.

[28] J. Handl, J. Knowles. "Feature subset selection in unsupervised learning via multiobjective optimization". Int. J. Comput. Intell. 2 (3) 217238, 2003.

[29] T. Hastie, T. Tibshirani, J. Friedman. "The Elements of Statistical Learning". Data Mining, Inference and Prediction, second ed., Springer, 2009.