International Journal of Computer Applications Technology and Research
Volume 9–Issue 03, 115-124, 2019, ISSN:-2319–8656

Model for Intrusion Detection Based on Hybrid Feature Selection Techniques

Joseph Mbugua Chahira
Department of Information and Computer Science
Garissa University, Kenya

Abstract
In order to safeguard their critical systems against network intrusions, organisations deploys multiple Network Intrusion Detection System (NIDS) to detect malicious packets embedded in network traffic based on anomaly and misuse detection approaches. The existing NIDS deal with a huge amount of data that contains null values, incomplete information, and irrelevant features that affect the detection rate of the IDS, consumes high amount of system resources, and slowdown the training and testing process of the IDS. In this paper, a new feature selection model is proposed based on hybrid feature selection techniques (information gain, correlation, chi squere and gain ratio) and Principal Component Analysis (PCA) for feature reduction. This study employed data mining and machine learning techniques on NSL KDD dataset in order to explore significant features in detecting network intrusions. The experimental results showed that the proposed model improves the detection rates and also speed up the detection process.
Key words cyber attacks, Intrusion detection, feature selection, data mining.

Introduction

Network Intrusion Detection System (IDS) [1] monitors the use of computers and networks over which they communicate , searching for unauthorised use, anamolous behaviour, and attempt to deny users, machines or portions of networks access to the services. Although the intrusion detection systems are increasingly deployed in the computer network, they deal with a huge amount of data that contains null values, incomplete information, and irrelevant features. The analysis of the large quantities of data can be tedious, time-consuming and error-prone. Data mining and machine learning[2] provides tools to select best relevance features subset which improves detection accuracy and removes distractions.

Feature selection problem can be characterised in the context of machine learning [3][4], [5]. Assume that T = D(F,C) is a training dataset with m instances and n features, where D = o_1, o_2, . . . , om and F = f_1, f_2, . . . , fn are the sets of instances and features. C = c_1, c_2, . . . , ck refers to the set of class labels. For each instance oj ∈ D, it can be denoted as a value vector of features, i.e., o_j = (v_{j1}, v_{j2}, . . . , v_{jn}), v_{ji} is the value of o_j corresponding to the feature f_i.. Therefore, feature selection plays an important role in alert correlation through reduction in the amount of data needed to achieve learning, improved predictive accuracy, learned knowledge that is more compact and easily understood and reduced execution time.

The existing feature selection techniques in machine learning can be broadly cassified into two categories i.e wrappers and filters. Wrappers selection techniques evaluate the worth of features using the learning algorithm applied to the data while filters evaluate the worth of features by using heuristics based on general characteristics of the data. Feature selection algorithms can be further differentiated by the exact nature of their evaluation function, and by how the space of feature subsets is explored. Wrappers often give better results in terms of the final predictive accuracy of a learning algorithm than filters because feature selection is optimized for the particular learning algorithm used. However, since a learning algorithm is employed to evaluate each and every set of features considered, wrappers are prohibitively expensive to run, and can be intractable for large databases containing many features. Furthermore, since the feature selection process is tightly coupled with a learning algorithm, wrappers are less general than filters and must be re-run when switching from one learning algorithm to another.

The advantages of filter approaches in feature selection outweigh their disadvantages. Flters execute many times faster as compaired to wrappers and therefore applicable in databases with a large number of features [6]. They do not require re-execution for different learning algorithms and can provide an intelligent starting feature subset for a wrapper incase improved accuracy for a particular learning algorithm is required[7]. Filter algorithms also exhibited a number of drawbacks. Some algorithms do not handle noise in data, and others require that the level of noise be roughly specified by the user a-priori [3], [7]. In some cases, a subset of features is not selected explicitly; instead, features are ranked with the

final choice left to the user. In other cases, the user must specify how many features are required, or must manually set a threshold by which feature selection terminates. Some algorithms require data to be transformed in a way that actually increases the initial number of features. This last case can result in a dramatic increase in the size of the search space[3].

The rest of the paper is organized as follows: Section II presents some related researches on intrusion detection which cover the feature selection and data mining. Section III briefly describes the KDD dataset used in this research. Section IV explains the details of the dataset pre-processing phase of the proposed model. The proposed model is presented in Section V. Finally, the experimental results and analysis are presented in Section 6 followed by some conclusions in the final section.

RELATED WORK

Recent study indicates that machine learning algorithms can be adversely affected by irrelevant and redundant training information [8]. The simple nearest neighbour algorithm is sensitive to irrelevant attributes, its sample complexity (number of training examples needed to reach a given accuracy level) grows exponentially with the number of irrelevant attributes[9][10]. Sample complexity for decision tree algorithms can grow exponentially on some concepts (such as parity) as well. The naive Bayes classifier can be adversely affected by redundant attributes due to its assumption that attributes are independent given the class [11] . Decision tree [12], [13] algorithms such as C4.5 overfit training data, resulting in large trees. In many cases, removing irrelevant and redundant information can result in C4.5 producing smaller trees.

As a result, most researchers combines the feature selection and classification algorithms to improve the detection accuracy and make intelligent decisions in determining intrusions. Siraj et al. [16] proposed new, automated and intelligent hybrid clustering model called Improved Unit Range and Principal Component Analysis with Expectation Maximization (IPCA-EM) to aggregate similar alerts as well as to filter the low quality alerts. Panda et al. [2] proposed a hybrid intelligent approach using combination of classifiers in order to make the decision intelligently, so that the overall performance of the resultant model is enhanced. These two models use hybrid classifiers to make intelligent decisions and the filtering process is applied after adding supervised or unsupervised learning techniques to obtain the final decision. Agarwal et al. [47] proposed hybrid approach for anomaly intrusion detection system based on combination of both entropy of important network features and support vector machine.

Madbouly et al.(2014), proposed a relevant feature selection model that selects a set of relevant features to be used in designing a lightweight, efficient, and reliable intrusion detection system. Although, the model achieved good overall detection result; detection results for PROBE, U2R, R2L attack types were low.

Lin et al. (2015) studied the importance of feature representation method on classification process. They proposed cluster centre and nearest neighbour (CANN) approach as a novel feature representation approach. In their approach, they measured and summed two distances. The first distance measured the distance between each data sample and its cluster centre. The second distance measured the distance between the data and its nearest neighbour in the same cluster. They used this new one-dimensional distance to represent each data sample for intrusion detection by a k-nearest neighbour (k-NN) classifier. The proposed approach provided high performance in terms of classification accuracy, detection rates, and false alarms. In addition, it provided high computational efficiency for the time of classifier training and testing

Zhao et al. (2015) proposed a new model based on immune algorithm (IA) and BPNN. The new developed method is used to improve the detection rate of new intruders in coal mine disaster warning internet of things. IA was used to preprocess network data, extract key features and reduce dimensions of network data by feature analysis. BPNN is adopted to classify the processed data to detect intruders. Experiments' results showed the feasibility and effectiveness of the proposed algorithm with a detection rate above 97%.

Methodology

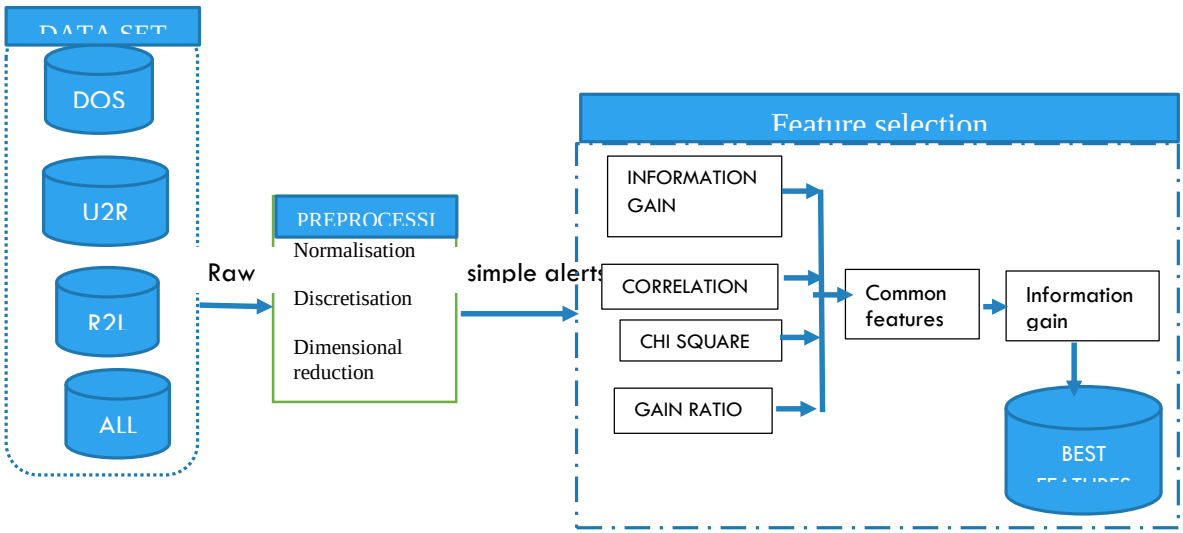

Figure 1 best feature selecion process

The proposed model has four phases as shown in figure 1 above:

Phase 1 data pre-processing

Phase 2 Dimension Reduction.

Phase 3 best feature selection.

Phase 4 Evaluation.

Data Preprocessing

To make efficient use of the available dataset for analysis the data preprocessing is required to provide solutions to Clean the data to remove noise and duplicate information and then deal with any incomplete or missing data an efficient algorithm based on normalization and discretization techniques. Data normalizaion is a process of scaling the value of each feature into a well-proportioned range, so that the bias in favor of features with greater values is eliminated from the dataset [14]. Every attribute within each record is scaled by the respective maximum value and falls into the same range of [0-1]. Normalization follows equation 1,

$$\text{Normalized}(x_i) = \frac{(X_i - X_{min})}{(X_{Max} - X_{min})} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 1$$

where X_{min} is the minimum value for variable X, X_{max} is the maximum value for variable X. For a specific symbolic feature, we assigned a discrete integer to each value and then used equation 1 to normalize it.

Discretization transforms continuous valued attributes to nominal [15][16]. The main benefit is that some classifiers can only take nominal attributes as input, not numeric attributes and also some classifiers can only take numeric attributes and hence can achieve improved accuracy if the data is discretized prior to learning.

Recently, organizations use cooperative NIDSs to provide a better detection and global view of intrusion activities. This contributes to the diversity of output formats. In order to correlate alerts such diversified formats have to be converted into a unified standard representation. Intrusion Detection Messaging Format (IDMEF) define a common data formats and exchange procedures for shairing important information to intrusion detection and response system.

Dimension Reduction Using Principal Component Analysis (PCA)

Data reduction algorithms reduce massive data-set to a manageable size without significant loss of information represented by the original data [22]. PCA is a useful technique for dimension reduction and multivariate analysis where the extracted components are statistically orthogonal to each other [13]. This enables speedup of training and robust convergence and hence can be applied in the intrusion alerts dataset to find the principal components of the alerts, i.e., the attributes vector that can describe the alerts exactly and sufficiently, but not redundantly. Mathematically, will establish the principal components of the distribution of the alerts, or the eigenvectors of the covariance matrix of the set of the alerts [23], [24],[21].

The PCA algorithm consists of 5 main steps :

1. Deduct the mean: deduct the mean from each of the data dimensions.
2. Calculate the covariance matrix: Cmxn = (Ci,j, Ci,j = cov(Dimi,Dimj) .....(1) Where Cmxn is a matrix in which all entry is the result of computing the covariance between two distinct dimensions.
3. Compute the eigenvectors and eigenvalues of the covariance matrix.
4. Select components and form a feature vector: once FeatureVector = (eig1, eig2, ..., eigN) .....(2)
5. Derive the new data set. Consider the transpose of the FeatureVector and multiply it on the left of the main data set, transposed:

FinalData = RowFeatureVector x RowDataAdjusted ... (3)

where RowFeatureVector is the matrix with the eigenvectors in the columns transposed and RowDataAdjusted is the mean-adjusted data.

Feature Selection Techniques

The feature selection techniques help to identify some of the important attributes in a data set, thus reducing the memory requirement, increase the speed of execution and improves the classification accuracy[17]. The purpose of this work is to find out which data feature selection algorithm gives better results with decision trees classifiers. Several feature subset selection techniques have been used in data mining.

i. Correlation based feature selection (CFS)

CFS is considered as one of the simplest yet effective feature selection method which is based on the assumption that features are conditionally independent given the class, where feature subsets are evaluated according to a correlation based heuristic evaluation function.[18]. A good feature subset is one that contains features highly correlated with the class, yet uncorrelated with each other. The major advantage of CFS, it is a filter algorithm, which makes it much faster compared to a wrapper selection method since it does not need to invoke the learning algorithms [19],[20].

$$\rho(X,Y) = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\left[\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2\right]^{\frac{1}{2}}} \quad \text{.....(2)}$$

Pearson's correlation coefficient (2), where all variables have been standardized shows that the correlation between a composite and an outside variable is a function of the number of component variables in the composite and the magnitude of the inter-correlations among them, together with the magnitude of the correlations between the components and the outside variable.

ii. Information Gain

Information gain is used as a measure for evaluating the worth of an attribute based on the concept of entropy (1), the higher the entropy the more the information content. Entropy can be viewed as a measure of uncertainty of the system. The largest mutual information between each feature and a class label within a certain group is then selected (2). The performance evaluation results show that better classification performance can be attained from such selected features [21],[19].

$$-\sum_i P(c_i)\log_2 P(c_i). \quad \text{.....(3)}$$

$$IG(A) = I(D) - \sum_{j=1}^{p} \frac{|D_j|}{|D|} I(D_j^A) \quad \text{.....(4)}$$

Algorithm 1: Feature selection according to information gains

Input: A training dataset T = D(F,C), number of features to be selected L

Output: Selected features S

1. Initialize relative parameters: F← fi, i =1, 2, ...n, C←'class labels', S =? ;
2. for each feature fi ∈ F do
   a. Calculate its information gain IG(fi) ;
   b. insert fi into S in descending order with regard to IG(fi) ;
3. Retain first L feature in S, and delete the others ;
4. Return Selected features: S.

iii. Chi-square

Chi-square [19] test is commonly used method, which evaluates features individually by measuring chi- square statistic with respect to the classes. The statistic is

$$\sum_{i=1}^{k} \sum_{i=1}^{k} \frac{(A_{ij} - E_{ij})^2}{E_{ij}} \quad \text{.....5}$$

Where,

k = No. of attributes,

n = No. of classes,

Aij = number of instances with value i for attribute and j for the class,

Eij = the expected No. of instances for Aij.

The larger value of the χ^2, indicates highly predictive to the class.

Data Set

The experiments will be conducted on MIT Lincoln's Lab's DARPA 2000 Scenario Specific NSL-KDD, 2014 which contains simulated attack scenarios in a protected environment an off-site server. KDD"99 testing set includes 37 attack types that are included in the testing set.

The NSL-KDD dataset has the following advantages over the original KDD dataset [4], [25].

i. It does not include redundant records in the train set, so the classifiers will not be biased towards more frequent records.

ii. The number of selected records from each difficulty level group is inversely proportional to the percentage of records in the original KDD data set. As a result, the classification rates of distinct machine learning methods vary in a wider range, which makes it more efficient to have an accurate evaluation of different learning techniques.

iii. The numbers of records in the train and test sets are reasonable, which makes it affordable to run the experiments on the complete set without the need to randomly select a small portion. Consequently, evaluation results of different research works will be consistent and comparable.

The simulated attacks in the NSL-KDD dataset fall in one of the following four categories[9], [24], [26], [27].

i. Denial of service attack (Dos), where attempts are to shut down, suspend services of a network resource remotely making it unavailable to its intended users by overloading the server with too many requests to be handled. e.g. syn flooding. Relevant features includes source bytes and percentage of packets with errors. Examples of attacks includes back,land, neptune, pod, smurf, teardrop

ii. Probe attacks, where the hacker scans the networkof computers or DNS server to find valid IP, active ports, host operating system and known vulnerabilities with the aim discover useful information. Relevant features includes duration of connection and source bytes. Examples includes Ipsweep, nmap, portsweep, satan

iii. Remote-to-Local (R2L) attacks, where an attacker who does not have an account with the machine tries to gain local access to unauthorized information through sending packets to the victim machine exfiltrates files from the machine or modifies in transist to the machine. Relevant features includes number of file creations and number of shell prompts invoked. Attacks in this category includes ftp_write, guess_passwd, imap,multihop, phf, spy, warezclient, warezmaster

iv. User-to-Root (U2R) attacks, where an attacker gains root access to the system using his normal user account to exploit vulnerabilities. Relevant features includes Network level features – duration of connection and service requested and host level features - number of failed login attempts. Attacks includes buffer_overflow, loadmodule, perl,rootkit

Using this format each generated alert is characterised by a set of attributes.

i. Basic attributes. that represent an alert and they are in IDMEF format. Examples of these attributes include timestamp, signature identifier, messages associated with alerts, protocol, IP source and IP destination addresses, source port and destination address, Time to live and identification field.

ii. Content features: The features of suspicious behavior in the data portion should be captured in order to detect attacks. E.g. number of failed login attempts. Those features are called content features. The R2L and U2R attacks normally don't appear in intrusion frequent sequential patterns, as they have been embedded in the data portions of packets and only request a single connection. While the DoS and Probing attacks involve many connections to hosts and show the attribute of intrusion frequent sequential patterns.

iii. Time-based traffic features: Only the connections in the past two seconds are examined, which have the same destination host/service as the current connection, and of which the statistics related to protocol behavior, service, etc. are calculated.

iv. Connection-based traffic features: Some slow probing attacks scan the hosts/service at an internal much longer than two seconds, e.g. once in every minute, which cannot be

v. detected by the time-based traffic features, as it only examines the connections in the past 2 seconds. In such case, the features of same destination host/service connections can be re-calculated at an interval of every 100 connections rather than a time window

Table 1: Description of basic features, content features, traffic features and host-based features.

vi.	Feature	Description	Type
1	Duration	Length of the connection.	Basic Features
2	protocol type	Connection protocol (e.g. tcp, udp)	Basic Features
3	Service	Destination service (e.g. telnet, ftp)	Basic Features
4	Flag.	Normal or error status of the connection	Basic Features
5	source byte	Bytes sent from source to destination	Basic Features
6	destination bytes	Bytes sent from destination to source	Basic Features
7	Land	1 - Connection is from/to the same host/port; 0 – otherwise.	Basic Features
8	Wrong_Fragment	Number of “wrong” fragments	Basic Features
9	Urgent	Number of urgent packets	Content Features
10	Hot	number of “hot indicators”.	Content Features
12	num_failed_logins	number of failed login attempts	Content Features
13	logged_in	1 - successfully logged in; 0 - otherwise	Content Features
14	num_compromise	number of “compromised” conditions.	Content Features
15	root_shell	number of “compromised” conditions.	Content Features
16	su_attempted	1 - root shell is obtained; 0 – otherwise.	Content Features
17	num_root	number of “root” accesses.	Content Features
18	num_file_creations	number file creation operations	Content Features
19	num_shells	number of shell prompts	Content Features
20	Num_access_files	number of operations on access control files	Content Features
21	Num_outbound_cmds	number of outbound commands in a ftp session	Content Features
22	is_hot_login	1 - the login belongs to the “hot” list; 0 – otherwise.	Content Features
23	is_guest_login	1 - the login is a “guest”login; 0 - otherwise	Content Features
24	Count	number of connections to the same host as the current connection in the past 2 seconds	Time-based Traffic Feature
25	srv_count	number of connections to the same service as the current connection in the past 2 seconds	Time-based Traffic Features
26	serror_rate	% of connections that have “SYN” error	Time-based Traffic Features
27	rerror_rate	% of connections that have “REJ” errors	Time-based Traffic
28	same srv rate	% of connections to the same service	Time-based Traffic
29	diff srv rate	% of connections to different services	Time-based Traffic
30	srv_serror_rate	% of connections that have “SYN” errors	Time-basedTraffic
31	srv_rerror_rate	% of connections that have “REJ” errors	Time-based Traffic
32	srv_diff_host_rate	% of connections to different hosts	Time-based Traffic
33	Dst_host_count	count of connections having the same destination host	Host-based Traffic Feature
34	dst_host_srv_count	count of connections having the same destination host and using the same service	Host-based Traffic Feature

35	dst_host_same_srv_ rate	% of connections having the same destination host and using the same service	Host-based Traffic Feature
36	dst_host_diff_srv_rate	% of different services on the current host	Host-based Traffic Feature
37	dst_host_same_src_port_rat	% of connections to the current host having the same src port	Host-based Traffic Feature
38	Dst_host_ srv_diff_host_rate	% of connections to the same service coming from different hosts	Host-based Traffic Feature
39	Dst_host_srv_rerror_rate	% of connections to the current host and specified service that have an S0 error Cont	Host-based Traffic Feature
40	dst_host_serror_rate	% of connections to the current host that have an S0 error	Host-based Traffic Feature
41	dst_host_srv_serror_rate	% of connections to the current host and specified service that have an S0 error	Host-based Traffic Feature

Experimental Setup Results And Discussion

Experiment Setup

Several data mining techniques which includes data cleaning and pre-processing, clustering, classification, regression, visualization and feature selection have been implemented in WEKA (Waikato Environment for Knowledge Analysis) [28]. Weka also offers some functionality that other tools do not, such as the ability to run up to six classifiers on all datasets, handling multi-class datasets which other tools continue to struggle with tools.

In the experiment, we apply full dataset as training set and 10-fold cross validation for the testing purposes. The available dataset is randomly subdivided into 10 equal disjoint subsets and one of them is used as the test set and the remaining sets are used for building the classifier. In this process, the test subset is used to calculate the output accuracy while the N_1 subset is used as a test subset and to find the accuracy for each subset. The process is repeated until each subset is used as test set once and to compute the output accuracy of each subset. The final accuracy of the system is computed based on the accuracy of the entire 10 disjoint subsets.

All experiments are performed using Windows platform with the following configuration Intel Core-i5 processor, 2.5GHz speed, and 8GB RAM.

For our experiment, we selected attribute set based on the repetition of attribute from four scheme. Existing FS that are employed in experiments are Correlation Feature Selection (CFS) based evaluator with Best-first searching method, Gain Ratio (GR) Attributes based Evaluator with Ranker searching method, Information Gain (IG) based Attributes Evaluator with ranker searching method, and Chi Squared Eval and Ranker searching method we obtained-.

Table 2: The most important features to distinguish between normal network traffic and cyber-attacks.

Feature selection techniques	No of features	Selected Features
Original Dataset	41	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,31,33,34,35, 36,37,38,39,40,41
Information gain, ranker	10	2, 40,3,41,27,26,30,31,32,35
CFS , best first,	8	2,3,9,23,26,27,34,41
Gain ratio and ranker	9	9,23,41,22,36,3,27,35,2
Chi Squared Eval + Ranker	9	2,40,3,41,26,27,30,31,32
Proposed	11	2,3,4,26,27,36,39,41

Table 3: The most important features to distinguish between normal network traffic and DoS attacks.

Feature selection techniques	No of features	Selected Features
Original Dataset	41	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,31,33,34,35, 36,37,38,39,40,41
Information gain, ranker	10	2, 40,3,41,27,26,30,31,32,35

CFS , best first,	8	2,3,9,23,26,27,34,41
Gain ratio and ranker	10	9,23,41,22,36,3,27,35,2,26
Chi Squared Eval + Ranker	10	2,40,3,41,26,27,30,31,32,20
Proposed	9	2,3,9,26,41.4,27

Table 4: The most important features to distinguish between normal network traffic and Probing attacks.

Feature selection techniques	No of features	Selected Features
Original Dataset	41	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,31,33,34,35, 36,37,38,39,40,41
Information gain, ranker	10	2, 40,3,30,,34,9,33,32,31,38
CFS , best first,	9	2,3,9,24,26,30,34,38,40
Gain ratio and ranker	10	25,9,24,3,2,41,38,40,34,26,
Chi Squared Eval + Ranker	10	2,40,3,33,34,30,32,38,31,37
Proposed	7	2,3,9,30,34,38,40

Table 5: The most important features to distinguish between normal network traffic and R2L attacks

Feature selection techniques	No of features	Selected Features
Original Dataset	41	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,31,33,34,35, 36,37,38,39,40,41
Information gain, ranker	10	1,2, 40,3,30,7,33,40,21,20,34,11
CFS , best first,	5	1,2,7,8,33
Gain ratio and ranker	10	1,8,7,19,2,3,33,40,21,20,34,11
Chi Squared Eval + Ranker	10	1,2,7,3,40,33,19,34,30,29,21
Proposed	9	1,2,7,33,3,40,34,30,21

Table 6: The most important features to distinguish between normal network traffic and U2R attacks.

Feature selection techniques	No of features	Selected Features
Original Dataset	41	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,31,33,34,35, 36,37,38,39,40,41
Information gain, ranker	10	11,40,3,30,10,29,14,1,33,21
CFS , best first,	4	6,11,29,30
Gain ratio and ranker	10	6,11,10,14,13,3,29,30,1,33
Chi Squared Eval + Ranker	10	6,11,3,10,14,40,30,29,31,1
Proposed	7	6,11,29,30,3,10,14

Table 7: The best set of relevant features

ALL	8	2,3,4,26,27,36,39,41
DOS	9	2,3,9,26,41,4,26,27,41
PROBE	7	2,3,9,30,34,38,40
R2L	8	1,2,7,33,3,40,34,30,21
U2R	7	6,11,29,30,3,10,14
Best Feature	12	1,2,3,9,26,27,29,30,34,36,39,40

Results Analysis and Discussion

As indicated by Table 7, the 41-features were reduced to 12-features. Features (6,11,29,30,3,10,14) are relevant for U2R, Features (1,2,7,33,3,40,34,30,21) are relevant for R2. Features (2,3,9,30,34,38,40) are relevant for PROBE class. Features (2,3,9,26,41,4,26,27,41) were selected as relevant DOS.

The category with least relevance features for detecting cyber-attacks is the Content Based Features. This results are biased because of the distribution of train and test datasets where the U2R and R2L attacks is less frequent than the most important features for DoS and Probing attacks. This implies that it is not necessary to analyse contents in the network traffic packages to detect cyber-attacks. The aspect of privacy and integrity for employees is then protected. A content feature can be the text in an email, and to store and analyse this kind of content violates the employees' integrity.

The Basic Features are most important to analyse to distinguish between normal network traffic and cyber-attacks.These features describes the number of seconds for the connection, the protocol used for the connection, the network service on the destination, normal or error status of the connection and the number of data bytes sent between source and destination computer. The results indicate the significance of analysing basic network traffic features to detect cyber-attacks.

The most important attributes to detect DoS attacks includes Src_bytes, Diff_srv_rate, Service, Dst_bytes and Flag. The content features are the least important category of features to detect a DoS attack. The reason for this is that the DoS attacks mostly consist of either no content or filled with a large amount of useless information.

The Host-based features like dst_host_srv_count, dst_host_serror_rate is important to detect probing attacks. These attacks takes longer time and in different ports and also seek known vulnerabilities.

For R2L attacks, the most important features duration, Src_bytes, Dst_bytes and srv_count. The duration represents the number of seconds for the connection and several R2L attacks have a duration which is much larger than a normal connections. The time-based feature, srv_count, which represent the number of connections have a low value compared to normal network traffic. During a R2L attack, the attacker tries to gain access to a local user account with a specific service in connections longer than 2 seconds.

To detect U2R attacks the most important feature includes Service, num_failed_logins, root_shell since the U2R attacks involves the use of specific services for remote access, often in combination with a file transfer service. Compared to the other attack categories, the content features are very important to detect U2R attacks. The content features are created by analysing the content in a network connection. The importance of content features to detect U2R attacks are as a result of the remote users actions that can only be noticed when analysing the content in the connection packages.

Conclusions

The most important features to detect cyber-attacks are basic features such as source byte, destination byte, the used service, a flag to indicate the status of the connection. Moreover time-based traffic features is important to analyse and detect cyber-attacks, such as information about the percentage of connections in the past 2 seconds with a different service than current connection. To detect R2L and U2R attacks it is important to study content features.

Refferences

[1] M. M. Siraj, H. Hussein, T. Albasheer, and M. M. Din, "Towards Predictive Real-time Multi-sensors Intrusion Alert Correlation Framework," *Indian J. Sci. Technol. ISSN*, vol. 8, no. 12, pp. 974–6846, 2015.

[2] M. C. Belavagi and B. Muniyal, "Performance Evaluation of Supervised Machine Learning Algorithms for Intrusion Detection," *Procedia Comput. Sci.*, vol. 89, pp. 117–123, 2016.

[3] J. Song, "Feature Selection for Intrusion Detection System Jingping Song Declaration and Statement," p. 132, 2016.

[4] N. A. Biswas, F. M. Shah, W. M. Tammi, and S. Chakraborty, "FP-ANK: An improvised intrusion detection system with hybridization of neural network and K-means clustering over feature selection by PCA," *2015 18th Int. Conf. Comput. Inf. Technol. ICCIT 2015*, pp. 317–322, 2016.

[5] J. H. Assi and A. T. Sadiq, "NSL-KDD dataset Classification Using Five Classification Methods and Three Feature Selection Strategies," vol. 7, no. 1, pp. 15–28, 2017.

[6] M. Othman and T. Maklumat, "Mobile Computing and Communications: An Introduction," *Malaysian J. Comput. ...*, vol. 12, no. 2, pp. 71–78, 1999.

[7] K. Kumar, "Network Intrusion Detection with Feature Selection Techniques using Machine-Learning Algorithms," vol. 150, no. 12, pp. 1–13, 2016.

[8] N. A. Noureldien and I. M. Yousif, "Accuracy of Machine Learning Algorithms in Detecting DoS Attacks Types," vol. 6, no. 4, pp. 89–92, 2016.

[9] A. Thesis, "Using Support Vector Machines in Anomaly Intrusion Detection by," 2015.

[10] P. Verma, "Performance of Detection Attack using IDS Technique," vol. 4, no. 3, pp. 624–629, 2016.

[11] J. Juanchaiyaphum, N. Arch-int, and S. Arch-int, "A Novel Lightweight Hybrid Intrusion Detection Method Using a Combination of Data Mining Techniques," *Int. J. Secur. its Appl.*, vol. 9, no. 4, pp. 91–106, 2015.

[12] P. Manandhar, "A Practical Approach to Anomaly - based Intrusion Detection System by Outlier Mining in Network Traffic By," 2014.

[13] A. I. Madbouly, A. M. Gody, and T. M. Barakat, "Relevant Feature Selection Model Using Data Mining for Intrusion Detection System," *Int. J. Eng. Trends Technol.*, vol. 9, no. 10, pp. 501–512, 2014.

[14] M. A. Ambusaidi, X. He, Z. Tan, P. Nanda, L. F. Lu, and U. T. Nagar, "A Novel Feature Selection Approach for Intrusion Detection Data Classification," *2014 IEEE 13th Int. Conf. Trust. Secur. Priv. Comput. Commun.*, pp. 82–89, 2014.

[15] D. a. M. S. Revathi, "A Detailed Analysis on NSL-KDD Dataset Using Various Machine Learning Techniques for Intrusion Detection," *Int. J. Eng. Res. Technol.*, vol. 2, no. 12, pp. 1848–1853, 2013.

[16] S. K. Sahu, S. Sarangi, and S. K. Jena, "A detail analysis on intrusion detection datasets," *Souvenir 2014 IEEE Int. Adv. Comput. Conf. IACC 2014*, no. December, pp. 1348–1353, 2014.

[17] Z. Dewa and L. A. Maglaras, "Data Mining and Intrusion Detection Systems," *Int. J. Adv. Comput. Sci. Appl.*, vol. 1, no. 1, p. 1:7, 2016.

[18] Y. Wahba, E. ElSalamouny, and G. ElTaweel, "Improving the Performance of Multi-class Intrusion Detection Systems using Feature Reduction," *Ijcsi*, vol. 12, no. 3, pp. 255–262, 2015.

[19] V. Barot, S. Singh Chauhan, and B. Patel, "Feature Selection for Modeling Intrusion Detection," *Int. J. Comput. Netw. Inf. Secur.*, vol. 6, no. 7, pp. 56–62, 2014.

[20] M. B. Shahbaz, X. Wang, A. Behnad, and J. Samarabandu, "On Efficiency Enhancement of the Correlation-based Feature Selection for Intrusion Detection Systems," 2016.

[21] A. AliShah, M. Sikander Hayat Khiyal, and M. Daud Awan, "Analysis of Machine Learning Techniques for Intrusion Detection System: A Review," *Int. J. Comput. Appl.*, vol. 119, no. 3, pp. 19–29, 2015.

[22] I. Syarif, A. Prugel-Bennett, and G. Wills, "Unsupervised clustering approach for network anomaly detection," *Networked Digit. Technol.*, vol. 293, 2012.

[23] S. Mallissery, S. Kolekar, and R. Ganiga, "Accuracy Analysis of Machine Learning Algorithms for Intrusion Detection System using NSL-KDD Dataset," vol. 4, no. 1, 2014.

[24] N. Shahadat, I. Hossain, A. Rohman, and N. Matin, "Experimental Analysis of Data Mining Application for Intrusion Detection with Feature reduction," pp. 209–216, 2017.

[25] L. Dhanabal and S. P. Shantharajah, "A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 4, no. 6, pp. 446–452, 2015.

[26] A. Jain and J. L. Rana, "Classifier Selection Models for Intrusion Detection System (Ids)," *Informatics Eng. an Int. J.*, vol. 4, no. 1, pp. 1–11, 2016.

[27] M. R. Parsaei, S. M. Rostami, and R. Javidan, "A Hybrid Data Mining Approach for Intrusion Detection on Imbalanced NSL-KDD Dataset," vol. 7, no. 6, pp. 20–25, 2016.

[28] M. Govindarajan and R. Chandrasekaran, "Intrusion Detection using an Ensemble of Classification Methods," *Proc. World Congr. Eng. Comput. Sci.*, vol. I, no. October, 2012.