Indian Journal of Science and Technology, Vol 9(11), DOI: 10.17485/ijst/2016/v9i11/84277, March 2016
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Enhanced Intrusion Detection System for Detecting Rare Class Attacks using Correlation based Dimensionality Reduction Technique

Shilpa Bahl* **and Deepak Dahiya**

CSE Department, Ansal University, Gurgaon - 122003, Haryana, India; gerashilpa@gmail.com, deepakdahiya@ansaluniversity.edu.in

Abstract

Background/Objective: With Fast growing internet world the risk of intrusion has also increased, as a result Intrusion Detection System (IDS) is the admired key research field. IDS are used to identify any suspicious activity or patterns in the network or machine, which endeavors the security features or compromise the machine. IDS majorly use all the features of the data. It is a keen observation that all the features are not of equal relevance for the detection of attacks. Moreover every feature does not contribute in enhancing the system performance significantly. The aim of the work done is to find out the smallest subset of most important attributes to design an efficient IDS model. **Methods/Statistical Analysis:** By implementing Correlation Feature Selection (CFS) mechanism using 6 search algorithms, a smallest set of features is selected with all the features that are selected very frequently. **Findings:** The smallest subset of features chosen is the most nominal among all the feature subset found i.e.12 features. Further, the performances using Naïve Bayes and Random Tree classifiers is compared for 7 subsets found by filter model and 41 attributes. **Results:** The outcome indicates a remarkable improvement in the performance metrics used for comparison of the two classifiers. The simulation results with enhanced classifiers accuracy is approx. 82% to 86% for Random tree and 33% to 65% for Naïve Bayes with 41 and 12 features respectively. There is a noticeable improvement in classifiers accuracy and exposure of U2R attacks s for the proposed smallest subset in comparison to other six subsets as shown in the result. **Application:** The proposed work with such an improved detection rate and lesser classification time and larger merits of the minimal subset found will play a vital role for the network administrator in choosing efficient IDS.

Keywords: Correlation based Attribute Selection, Feature Reduction, Intrusion Detection Model, Machine Learning Algorithms, and User to Root Attacks

1. Introduction

The speed of data flow in a local area network is around 100 gigabits per second i.e. millions of packets are flowing in the network per second. Monitoring such a fast and humongous data is a great challenge. However, present Intrusion Detection System (IDS) follow classical networks security mechanisms e.g. client verification, data cryptography and access rights for well known attacks but fails in case of novel intrusions. An IDS has been acknowledged as an interesting field to explore due to involvement of novel intrusions on the networks and on the machine. Intrusion Detection System (IDS) is a promising element to protect the network channels and the machine containing the confidential information from the nuisance of anomalous user behavior and misuse of information. The target of Intrusion Detection System is to detect any abnormality or irregularity in the user's action which breaches the authentication and security features of the networks, machine and database. Three different approaches of machine learning domain can be followed to design an efficient IDS[2] (1) signature misuse detection; (2) anomaly detection and (3) clustering. Classical signature-based misuse detection system comes under pattern classification problem which is not a promising solution for novel attacks as it monitors on

*Author for correspondence

the basis of old existing patterns. However anomaly based methods can easily classify normal data from anomalous traffic examining the present user behavior. Therefore for supervised classification approach, ID model is trained for discriminating the network data into respective attack classes in the testing phase[3].For supervised /semi-supervised anomaly based system a pattern of normal logs is constructed and flagged for the exceptions. Several anomaly based database IDS are compared depending on the scenario through which the intrusion is detected[3-5]. Data clustering or outlier detection is an unsupervised machine learning technique in which a set of objects are assigned to a cluster or outlier. There are a number of soft Computing approaches in literature implied for building IDS i.e. Genetic Algorithms, neural networks, Fuzzy logics, Honey Pot, Data mining and a lot more. This paper addresses basic data mining using neural networks for classifying various attacks into respective attack classes for constructing an efficient and competent ID[6].

The KDD Cup'99 dataset is a well known standard for assessment of IDS methods. There are some discrepancies found in this dataset due to duplicate instances and unbalanced class distribution[29]. In our work, we used a well known NSL-KDD Cup 99 dataset[7]. The data set is comprised of 41 features with 21 unique attacks in the training datasets categorized as Denial of Service Attacks (DoS), probing attacks (Probe), Remote to Local Attacks (R2L) and User to Root Attacks (U2R). The attack classes U2R and R2L are called rare classes as they have very small number of examples in the data set in comparison to other two attack classes. In the literature it is observed that uncovering the aforementioned 2 attacks is below an acceptable level for many misuse detection algorithms[8,9]. Moreover it is revealed in the literature that none of the machine learning approaches could be implemented significantly on KDD Cup99 dataset to achieve satisfactory point of misuse detection for user to Root or Remote to Locals attack classes. Reason being, the signature pattern of these liberal attacks in testing data set are not present in the training dataset[10].

The merit of input data merely affects the accuracy and efficiency of machine learning algorithms. However the quality of input is dependent on the most relevant features in the dataset. The Feature Selection (FS) is a renowned dimensionality reduction techniques for a given feature space. In dimensionality reduction mechanism a subset of the most pertinent features that contributes in machine learning process are chosen and other inappropriate and repetitive features are deleted. A single irrelevant feature in the dataset tends to confuses ML process[11]. Before the learning phase all the irrelevant features are removed in pre-processing phase, to reduce the adverse impact of these unrelated features on the classification algorithms. The Feature reduction techniques have been readily identified in the areas of ML and data mining for years[12-14]. It is shown in the literature that FS algorithm actually improves Machine Learning outcomes, decreasing the computational time and complexity of the model, with need of less storage space[15-17].

The purpose of the work is to find out the smallest set of the best contributing features from the dataset which significantly enhances the classifiers accuracy and efficiency of Intrusion Detection System. Correlation based Feature Subset Selection (CFS) technique is implemented to select six subsets of features. The smallest set of 12 features is introduced to significantly sustain the firmness and efficiency of IDS. Random tree classifier and Naïve Bayes classifiers are implemented for comparison among the reduced set of attributes with complete 41features. The paper shows empirical results for User to Root attack class. In the proposed approach we achieved a noteworthy performance with 12 selected subset of features.

2. Feature Subset Selection and Related Work

2.1 Feature Subset Selection Methodologies

Feature selection is a process of chucking out the irrelevant and redundant features from the total feature space during the Pre-Processing step. Moreover it reduces the negative effect on the actual machine learning algorithms. Feature subset selection methodologies are broadly categorized into, the filter method and the wrapper method.

The selection of feature subset in filter method is entirely dependent on the characteristics of the dataset not on the induction algorithm. Moreover, there are two directional approaches followed by filter method are forward selection and backward selection in sequential order. In Sequential Forward Selection, we initiate with an empty set and insert rest of the features one by one. In Sequential backward selection, we initiate with full set of features and remove them one by one. However wrapper method is entirely dependent on the induction algorithm, i.e. a predestined classifier is implemented to assess the selected set of features. Therefore later is computationally more costly and time consuming comparative to filter

method[11-13,18]. The comparison for 4 attack classes are shown in the literature using ten well known classification algorithms from Baysian Network family, Decision tree and rule based family and SVM. The detection rate of U2R attack class is marked in the range from 0.012 to 0.328. The detection rate of R2L attack class is marked in the range from 0.001 to 0.107[8].

Similarly one more relative study has been performed with 9 ML algorithms for the same 4 attack classes and the normal instances. The detection rate of User to Root attack class is marked in the range from 0.022 to 0.298. The finest outcomes out of the 3 renowned presented outcomes and the winner of the KDD cup 99 ID contest are also the basis for comparison[9].Due to imbalance dataset it is observed that there are some disconnects in the conclusion marked in the literature for the attacks that have very less number of examples. Therefore due to uneven number of instances of User to Root and Remote 2 Local attack types in the standard training and testing dataset the detection rate for the same has been dropped far below the acceptance level. The superior results are shown in the literature by the use of customized dataset rather than regular training and testing dataset[19, 29].

The performance of the classifiers depends on the quality of features given as an input therefore it should be the most relevant and irredundant subset[18].To come to a decision for the good quality of features several FS algorithms have been evaluated in the literature with decision tree family of classifiers[20]. Under the wrapper method of feature selection, enhanced SVM and decision trees for features selection was discussed in literature[21]. The Subset of seventeen and twelve features has been selected using Naïve Bayes and CART classifier respectively[22]. Introducing some hybrid architectures for features subset selection using decision tree and SVM is also there in the literature which again involves ensemble and base classifiers[23].

3. Statistical Analysis

The machine used for simulation has the mentioned configuration i.e. Intel T2080 processor, 1.73GHz, 2GB RAM. The data mining tool used is WEKA 3.7.11, heap size: 1048 MB with default parameter setting.

3.1 Research Methodology

The focus of the work done is to minimize the feature space during the pre-processing phase of the IDS model. CFS with six search methods is implemented to find out smallest subset of features which classifies all the attacks in their respective classes, occurring in the training dataset. The actual training and testing files now contains only the features that are selected from each FS algorithm and rest of the features are deleted forever. Now we are left with 6 sets of training and testing files received from 6 implemented Features Subset selection techniques. For all the six implemented FS algorithms the most vital features were very frequently selected. In the present work these frequently selected features i.e. the features that were selected maximum number of times (Six) were used to construct a new subset. The resulting subset came out to be for 12 most relevant features, which are loosely coupled with each other and contributes in efficient classification process. Two classifiers are used for representing the comparison among seven reduced datasets including the proposed minimal subset with all 41 features on the basis of the performance.

3.2 NSL-KDD 99 Dataset

It is observed from the literature[4] that KDD Cup 99 dataset have various problems of imbalanced classes and redundancy, but it still remains to be the benchmark for building Intrusion detection models. In order to address some of the very serious issues the data was customized to form a new dataset, called NSL-KDD Cup 99 dataset. Therefore NSL-KDD Cup99 dataset[27] has been used in our work. Though there are various sets of data under NSL-KDD i.e. the complete data set, 10% of the complete data set and 20% of the same, we have taken KDD 20% and KDD full data sets for training and testing respectively. The respective count of examples in the training and testing dataset are 25192 and 22544. The details of instances under normal, DoS, Probe, U2R and R2L classes for training and testing files are given in Table 1.

And the numbers of instances for the same attack categories in the testing file are shown in Table 2.

Another key observation for the dataset is that the count of attacks in training and testing files different i.e. 21 in training file and 37 in testing file respectively.There are some liberal attacks which are present in the testing files for which the model was not trained.

There are two more versions of NSL-Kdd Cup99 dataset which contain five and two classes. In this paper we have build the ID model for multiclass i.e. individual attacks are classified falling under four attack categories. All the variants of NSL-Kdd Cup99 dataset have similar

Table 1. List of Attacks with number of instances in Training File

DoS	No.	Probe	No.	R2L	No.	U2R	No.
Neptune	8282	Satan	691	Guess_ Password	10	Buffer_ overflow	6
Teardrop	188	Nmap	301	Warezmaster	7	Loadmodule	1
Land	2	Portsweep	587	Warezclient	181	rootkit	4
Smurf	529	IPSweep	710	Multihop	2		
Pod	38		2289	ftpwrite	1		
Back	196			Imap	5		
				Spy	1		
				Phf	2		

Table 2. List of Attacks with number of instances in Testing file

DoS	No.	Probe	No.	R2L	No.	U2R	No.
Neptune	4657	Satan	735	Guess_ Password	1231	Buffer_ overflow	20
Teardrop	12	Nmap	73	Warezmaster	944	Loadmodule	2
Land	7	Portsweep	157	Sendmail	14	rootkit	13
Smurf	665	IPSweep	141	Multihop	18	Sql Attack	2
Pod	41	Mscan	996	ftpwrite	3	Perl	2
back	359	Saint	319	Imap	1	Ps	15
mailbomb	293			Phf	2	Xterm	13
Processtable	685			Httptunnel	133		
worm	2			Xlock	9		
udpstrom	2			named	17		
Apache2	737			Xsnoop	4		
				Snmpgetattack	178		
				Snmpguess	331		

number of features (41), broadly categorized as basic, content, traffic and similar host features. The details of all the features are shown in column 2 of Table 4.

4. Correlation based Feature Subset Selection

As discussed earlier there are various filter models for ranking the features, which are broadly based on dependency, distance and consistency. Under dependency model it describes information gain, maximum relevance and minimum redundancy and Pearson's Correlation. In this paper we had made use of well-known Pearson's Correlation based Feature Selection (CFS) technique. According to Pearson, features that are extremely coupled with the projecting class and loosely with each other are the most relevant for ML process[24]. The Correlation based Feature Subset Selection is implemented to calculate the merit of a subset of features with k number of features in (1).

$$Ms = R_{FC} = \frac{Kr_{fc}}{\sqrt{K + k(K-1)r_{ff}}} \quad (1)$$

Where R_{FC}= Correlation among the Target class and the attributes.

r_{fc}= Average value of attribute – Target class correlation.

r_{ff} = Average value of attribute-attribute correlation.

Coefficient of Correlation is used to compute the weightage of the selected subset as the promising performance scale. It identifies the finest subset selected. The amount of correlation or dependency is the measure of the coupling among the features and the classes. Therefore features and the target class must be tightly coupled and features among each other should be loosely coupled. However, adding more features would definitely add to the amount of coupling among the target class attribute and the attributes. The latest induced attributes is less coupled with the already chosen attributes and have more dominance above an elevated correlation with the classes[24].

After generating the optimal candidate feature subsets using various search algorithms these subsets are evaluated using different evaluation criteria to find out the best subset for goodness. In this paper 6 search methods with SBE approach are employed, where we start with a complete set of features and eliminate them one by one. The details of the search methods are shown in the second column of Table 3.The details about the implemented search methods are given as follows[2]:

1. **Best First:** This search strategy searches the subsets from feature space by using greedy hill climbing amplified with backtracking.The intensity of backtracking may be controled by locating an amount of successive non-improving nodes. This serach method works both in SFS and SBE mode or may start from any random point and search bidirectionally.Therefore it have various control panels like direction, Serach termination, start set and lookup Cache Size etc.
2. **Greedy-stepwise:** Performs serach of subset from the feature space in forward as well as in backward direction using greedy hill climbing without backtracking facility.It can also generate record of ranked features

by scanning the feature space from one end to other end making the record of the order in which the features were selected.

3. **Genetic Search:** It use the general principal of genetic algorithm for finding out the subset in feature room, following various control panels eg. Crossover probability,number of generations, mutation probability, population size, seed etc.
4. **Scatter Search VI:** It uses sequential scatter search algorithm for finding out the subsets in feature space. It starts with some significant and diverse subsets and stops depending on some threshold value or when no improvement is revealed.Some of the control panels it have are combinations, seed and threshold.
5. **Random Search:** It starts from a random point or a given starting point to search the best subset in the feature room.
6. **Exhaustive Search:** It implements exhaustive search approach to find the subset in feature room. It initiates with no features and reveals the finest subset found.

4.1 Classification Algorithms

Two classifiers are used in this study from two different families of classifiers so that results are biased towards one algorithm.

4.1.1 Decision Tree

In our work, we used random tree classifier from decision tree family to assess the efficiency and effectiveness of IDS on selected sub-sets and complete set of 41attributes. In which the learning functions are represented in a tree like structure with a combination of rules on its nodes. The tree may be binary or have multiple branches. Nodes represent the decision on the features of an instance and branches showing the feature values. Moreover values of the target class are represented by leaves of the tree, emerging from the root node. Class probabilities are estimated using this classifier and no pruning is required[25].

4.1.2 Naïve Bayes Algorithm

It is a general probabilistic classifier which is independent of the class conditions. It works on "Bayes theorem" which calculates subsequent probabilities from the previous probabilities by recording both of them. Counting the frequency of occurrence is used to make an estimate of the two probabilities. Naïve Bayes algorithm is used to show a comparative study among the reduced subsets and complete set of 41 features. As an outcome it gives the class labels having maximum probabilities for making the decision. The algorithm works on the principal of "conditional independence" (Naive) i.e. the probability (occurrence or not) of one attribute is independent of the probability (occurrence or not) of other attributes and also to the known value of the target class attribute. It is an observation reported in the literature that the merit of Naïve Bayes, decision tree family some of the neural network classifiers is equivalent[25].

5. Empirical Outcomes and Discussions

The details of the implemented subset generation algorithms used, chosen sub-set of features, amount of features selected, merit of selected sub-set and the total number of sub-sets found during the model construction are shown in Table 3, from column 2 to 6 respectively. The features that are chosen using six search methods are reflected in Table 4 with feature names and labels in column 2 and 1 respectively. Columns 3 to 8 represents the six search techniques in Table 4.The total number of times a particular feature is selected is called the count of that feature reflected in column 9 of Table 4.It is observed from table 3 that the search method 1 and 2 are more economical than 5 and 6 because they take less computational time for sub-set selection from the feature space.

Table 3. List of Subset generation algorithms

Sr. No.	Search Method	Selected Subset of Features	#Features Selected	Merit Subset	#Subsets Formed
1	Best First	[2,3,4,5,6,8.10,12, 23,25,29,30,35,36, 37,38,40]	17	0.725	680
2	Greedy Stepwise	[2,3,4,5,6,8,10,12, 23,26,29,30,35,36, 37,38,40]	17	0.725	684
3	Genetic Search	[2,3,4,5,6,8,12,13,2 2,23,25,26,27,29,30, 31,32,33,36,37,38]	21	0.708	400
4	Scatter Search V 1	[2,3,4,5,6,8,11,12,14, 23,25,29,30,35,36,37 ,38,40]	18	0.722	17866
5.	Exhaustive Search	[2,3,4,5,6,8,10,12, 14,23,25,29.30,31, 35,36,37,38,40]	19	0.721	860
6.	Random Search	[2,3,4,5,6,8,10,11, 23,25,26,27,29,30, 36,37,38]	17	0.726	896

Table 4. Details of Selected Features

Label	Feature	Best First	Greedy Stepwise	Genetic Search	Scatter Search V1	Exhaustive Search	Random Search	Total
1	Duration							0
2	Protocol-type	√	√	√	√	√	√	6
3	Service	√	√	√	√	√	√	6
4	Flag	√	√	√	√	√	√	6
5	src_bytes	√	√	√	√	√	√	6
6	dst_bytes	√	√	√	√	√	√	6
7	Land							0
8	wrong_fragment	√	√	√	√	√	√	6
9	Urgent							0
10	Hot	√	√			√	√	4
11	num_failed_logins				√		√	2
12	logged_in	√	√	√	√	√		5
13	num_comromised			√				1
14	root_shell				√	√		2
15	su_attempted							0
16	num_root							0
17	num_file_creation							0
18	num_shells							0
19	num_access_files							0
20	num_outbound_cmds							0
21	is_host_login							0
22	is_guest_login			√				1
23	Count	√	√	√	√	√	√	6
24	srv_count							0
25	serror_rate	√		√	√	√	√	5
26	srv_serror_rate		√	√			√	3
27	rerror_rate			√			√	2
28	srv_rerror_rate							0
29	same_srv_rate	√	√	√	√	√	√	6
30	diff_srv_rate	√	√	√	√	√	√	6
31	srv_diff_host_rate			√		√		2
32	dst_host_count			√				1
33	dst_host_srv_count			√				1
34	dst_host_same_srv_rate							0
35	dst_host_diff_srv_rate	√	√		√	√		4
36	dst_host_same_src_port_rate	√	√	√	√	√	√	6
37	dst_host_srv_diff_host_rate	√	√	√	√	√	√	6
38	dst_host_serror_rate	√	√	√	√	√	√	6
39	dst_host_srv_serror_rate							0
40	dst_host_rerror_rate	√	√		√	√		4
41	dst_host_srv_rerror_rate							0
Total		17	17	21	18	19	17	

The features with a highest count of 6 in column 9 in Table 4 are selected among all the six search methods to propose a new subset of features. The proposed minimal subset(12 features) are labeled as 2,3,4,5,6,8,23,29,30,36, 37, 38 in Table 4. This smallest subset of features is most frequently selected by all 6 implemented search techniques based on correlation based feature selection principal. It is a key observation that no single feature belongs to content category i.e. these features does not contribute in detection of attacks. The rest of the fourteen features belong to the third category. The classifier model constructed during the training phase is a multiclass evolutionary model tested on the discussed testing file.

The empirical outcomes of the attack classes in the training file are saved. We have shown summary results of only a rare attack class (U2R) because it has very less number of examples. The result shows the performance comparison with some well known metrics for reduced datasets and complete set of features. Some novel attacks from User to Root class in the test files whose signature patterns are not found in training data-set are Perl, Sqlattack, ps, Xterm as shown in Table 2. They are left undetected by the classifier.

The final experimental outcomes for a complete set of features and for seven reduced subset of features are shown in Table 5, Table 6 and Table 7 for Random Tree and Naïve Bayes classifier. Table 5 and Table 6 shows detection rate (TPR), false positive rate (FPR), Recall, Precision, F-score and area under ROC curve (AUC) of the two classifiers.

The observation found from the results in Table 5 and Table 6, the TPR of load module attack is below the acceptable level, which adversely affects the overall true positive rate for User to Root attacks. Table 7 presents the other additional performance metrics (Classifier Accuracy, RMSE, and time to construct a model) in the column 2, 3 and 8 respectively for the two classifiers. Column 4, 5,

Table 5. Summary results of Random Tree Algorithm

Search Method	Attack type	TPR	FPR	Recall	Precision	F-Score	AUC
41 attributes	Buffer_overflow	0.100	0.0000	0.100	1.000	0.180	0.550
	Loadmodule	0.000	0.0000	0.000	0.000	0.000	0.500
	rootkit	0.150	0.0020	0.150	0.800	0.110	0.570
Best First	Buffer_overflow	0.380	0.0000	0.380	1.000	0.631	0.720
	Loadmodule	0.002	0.0000	0.002	0.007	0.003	0.500
	rootkit	0.164	0.0020	0.160	0.069	0.085	0.560
Greedy Stepwise	Buffer_overflow	0.330	0.0000	0.330	1.000	0.520	0.719
	Loadmodule	0.001	0.0000	0.110	0.005	0.002	0.567
	rootkit	0.150	0.0020	0.150	0.590	0.085	0.500
Genetic Search	Buffer_overflow	0.220	0.0000	0.200	0.660	0.170	0.550
	Loadmodule	0.000	0.0000	0.000	0.000	0.000	0.500
	rootkit	0.350	0.0000	0.350	0.420	0.490	0.730
Scatter Search V 1	Buffer_overflow	0.220	0.0000	0.630	0.540	0.390	0.600
	Loadmodule	0.000	0.0000	0.000	0.000	0.000	0.500
	rootkit	0.190	0.0000	0.540	0.075	0.330	0.500
Exhaustive Search	Buffer_overflow	0.190	0.0000	0.100	1.000	0.340	0.550
	Loadmodule	0.000	0.0000	0.000	0.000	0.000	0.500
	rootkit	0.110	0.0000	0.017	0.330	0.220	0.530
Random Search	Buffer_overflow	0.200	0.0000	0.200	1.000	0.330	0.600
	Loadmodule	0.000	0.0000	0.000	0.000	0.000	0.500
	rootkit	0.150	0.0100	0.155	0.070	0.010	0.560
Proposed selected (12) attributes	Buffer_overflow	0.260	0.0000	0.260	0.400	0.310	0.610
	Loadmodule	0.001	0.0000	0.001	0.010	0.003	0.510
	rootkit	0.390	0.0020	0.390	0.540	0.480	0.750

Table 6. Summarized Outcomes of Naive Bayes Algorithm

Search Method	Attack type	TPR	FPR	Recall	Precision	F-Score	AUC
41 attributes	Buffer_overflow	0.000	0.0070	0.010	0.012	0.020	0.601
	Loadmodule	0.000	0.0000	0.000	0.000	0.000	0.500
	rootkit	0.308	0.0040	0.308	0.053	0.090	0.691
Best First	Buffer_overflow	0.050	0.0400	0.050	0.020	0.070	0.619
	Loadmodule	0.000	0.0000	0.000	0.000	0.000	0.500
	rootkit	0.308	0.1000	0.308	0.040	0.090	0.664
Greedy Stepwise	Buffer_overflow	0.050	0.0400	0.050	0.020	0.070	0.619
	Loadmodule	0.000	0.0000	0.000	0.000	0.000	0.500
	rootkit	0.308	0.1000	0.308	0.040	0.090	0.664
Genetic Search	Buffer_overflow	0.015	0.0010	0.015	0.020	0.030	0.847
	Loadmodule	0.001	0.0020	0.001	0.010	0.010	0.051
	rootkit	0.390	0.0100	0.390	0.040	0.090	0.917
Scatter Search V 1	Buffer_overflow	0.020	0.0010	0.020	0.030	0.040	0.579
	Loadmodule	0.000	0.0000	0.000	0.000	0.000	0.500
	rootkit	0.300	0.0970	0.300	0.040	0.080	0.635
Exhaustive Search	Buffer_overflow	0.015	0.1600	0.150	0.050	0.040	0.700
	Loadmodule	0.001	0.0020	0.001	0.010	0.010	0.500
	rootkit	0.400	0.0400	0.400	0.100	0.080	0.860
Random Search	Buffer_overflow	0.100	0.0260	0.100	0.030	0.050	0.541
	Loadmodule	0.000	0.0000	0.000	0.000	0.000	0.500
	rootkit	0.308	0.0600	0.308	0.040	0.080	0.731
Proposed selected (12) attributes	Buffer_overflow	0.026	0.0100	0.026	0.070	0.040	0.720
	Loadmodule	0.002	0.0100	0.002	0.010	0.001	0.510
	rootkit	0.390	0.0200	0.390	0.100	0.080	0.870

Table 7. Average of Regular performance metrics.

	Search Method	Accuracy	RMSE	TPR	FPR	F-Score	AUC	Time
RANDOM TREE	41 attributes	82.70	0.125	0.083	0.001	0.097	0.540	2.75
	Best First	85.91	0.113	0.182	0.001	0.240	0.593	1.00
	Greedy Stepwise	85.81	0.114	0.160	0.001	0.202	0.595	1.26
	Genetic Search	83.49	0.122	0.190	0.000	0.220	0.593	1.48
	Scatter Search V 1	85.91	0.115	0.137	0.000	0.240	0.533	1.90
	Exhaustive Search	84.91	0.110	0.100	0.000	0.187	0.527	1.73
	Random Search	83.96	0.120	0.117	0.003	0.113	0.553	1.31
	Proposed selected (12) attributes	85.93	0.110	0.217	0.001	0.264	0.623	1.33
NAIVE BAYES	41 attributes	33.91	0.24	0.103	0.004	0.037	0.597	3.90
	Best First	62.53	0.177	0.119	0.047	0.053	0.594	1.16
	Greedy stepwise	62.63	0.178	0.119	0.047	0.053	0.594	1.11
	Genetic Search	60.52	0.198	0.135	0.004	0.043	0.605	2.29
	Scatter Search V 1	40.90	0.180	0.107	.033	0.040	0.571	1.69
	Exhaustive search	63.78	0.213	0.139	0.067	0.043	0.687	1.95
	Random Search	65.43	0.172	0.136	0.029	0.043	0.591	1.34
	Proposed selected (12) attributes	65.43	0.168	0.139	0.013	0.04	0.70	0.95

6 and 7 shows the average out of the labeled performance metrics in Table 7.

It is concluded from table 7 that, the performance of IDS is always improved by implementing Correlation based feature selection techniques. Moreover it is the best with the proposed minimal subset of features. The false alarms are also very small. The major contribution of the work presented in this paper is that we have identified the minimal subset of features and also improved the overall accuracy and TPR of U2R attack class. However we can say that only 12 features are adequate enough to discriminate one attack from the other in the training dataset.

The literature have reported[8,9,22,23] the use of 5 class dataset. They have revealed on the whole total TPR of U2R attack class. However in this paper, the TPR of individual attacks of U2R attack class are presented.

Further these performance metrics are used to build an alert post processing model to predict next attacker scenario by analyzing the severity and priority of alerts[28]. As discussed in the literature that there are many versions of NSL-KDD Cup dataset with 5 classes of attacks, 2 classes and multi class dataset. We have implemented multi-class dataset with detailed outcomes of only User to Root attacks.

For online IDS two of the implemented search techniques (Exhaustive and Random Search) are not at all suggested because their time to build the model is very high. However the two best techniques (Best first and scatter search) are suggested for the fastest network data monitoring by an IDS.

6. Conclusion

The work presented here, implemented correlation based feature selection using 6 search algorithms for constructing an effective and precise Intrusion Detection System (IDS). Six sub-sets of attributes are chosen from the attribute space using six search techniques. An anticipated subset (12 features) is extracted from the six identified subsets using the logic of the most frequent occurrence of features. For comparison two well known classifiers, Random tree classifier and Naïve Bayes algorithm are implemented. The performance comparison for 6 diverse subsets, anticipated subset and complete 41 features is shown in the results. The proposed smallest subset of 12 features had shown a noticeable improvement in the overall performance of IDS for detecting U2R attacks. Moreover the overall computational time for detection of the attacks is considerably small which will help the system administrator to take necessary action against the occurrence of these intrusions.

7. References

1. Kumar V, Srivastava J, Lazarevic A. Managing cyber threats: issues, approaches, and challenges. Springer Science and Business Media; 2006 Mar 30.
2. Han J, Kamber M, Pei J. Data mining: concepts and techniques: concepts and techniques. Elsevier; 2011 Jun 9.
3. Mohammed Masoud Javidi, Marjan Kuchaki Rafsanjani, Sattar Hashemi, Mina Sohrabi. An overview of Anomaly Based Database Intrusion Detection Systems. Indian Journal of Science and Technology. 2012 Oct; 5(10). Doi no:10.17485/ijst/2012/v5i10/30934
4. Tavallaee M, Stakhanova N, Ghorbani AA. Toward credible evaluation of anomaly-based intrusion-detection methods. Systems, Man, and Cybernetics, Part C: Applications and Reviews. IEEE Transactions on. 2010 Sep; 40(5): 516–24.
5. Garcia-Teodoro P, Diaz-Verdejo J, Maciá-Fernández G, Vázquez E. Anomaly-based network intrusion detection: Techniques, systems and challenges. Computers and security. 2009 Mar 31; 28(1):18–28.
6. Visumathi J, Shunmuganathan KL. A Computational Intelligence for Evaluation of Intrusion Detection System. Indian Journal of Science and Technology. 2011 Jan; 4(1). Doi no: 10.17485/ijst/2011/v4i1/29930
7. Revathi S, Malathi A. Network Intrusion Detection Based On Fuzzy Logic. International Journal of Computer Application 2014 February ; 4(1).Available online on http://www.rspublication.com/ijca/ijca_index.htm ISSN: 2250-1797,
8. Nguyen HA, Choi D. Application of data mining to network intrusion detection: classifier selection model. In Challenges for Next Generation Network Operations and Service Management. Springer Berlin Heidelberg; 2008 Jan 1. p. 399–408.
9. Sabhnani M, Serpen G. Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset within Misuse Detection Context. MLMTA; 2003 Jun.p. 209–215.
10. Sabhnani M, Serpen G. Why machine learning algorithms fail in misuse detection on KDD intrusion detection data set. Intelligent Data Analysis. 2004 Sep 1; 8(4):403–15.
11. Chizi B, Maimon O. Dimension reduction and feature selection. Data Mining and Knowledge Discovery Handbook. Springer US; 2005 Jan 1.p. 93–111.
12. van der Maaten LJ, Postma EO, van den Herik HJ. Dimensionality reduction: A comparative review. Journal of Machine Learning Research. 2009 Oct 26; 10(1-41): 66–71.

13. Liu H, Motoda H. Computational methods of feature selection. CRC Press; 2007 Oct. p. 29.
14. Saeys Y, Inza I, Larrañaga P. A review of feature selection techniques in bioinformatics. Bio-informatics. 2007 Oct 1; 23(19):2507–17.
15. Tang J, Alelyani S, Liu H. Feature selection for classification: A review. Data Classification: Algorithms and Applications. 2014 Jul: 37.
16. Stańczyk U. Ranking of characteristic features in combined wrapper approaches to selection. Neural Computing and Applications. 2015 Feb 1; 26(2):329–44.
17. Wang S, Tang J, Liu H. Embedded Unsupervised Feature Selection.Twenty-Ninth AAAI Conference on Artificial Intelligence; 2015 Oct 2.
18. Yu L, Liu H. Efficient feature selection via analysis of relevance and redundancy. The Journal of Machine Learning Research. 2004 Dec 1; 5:1205–24.
19. Engen V, Vincent J, Phalp K. Exploring discrepancies in findings obtained with the KDD Cup'99 data set. Intelligent Data Analysis. 2011 Apr 1; 15(2):251–76.
20. Piramuthu S. Evaluating feature selection methods for learning in data mining applications. European journal of operational research. 2004 Jul 16; 156(2):483–94.
21. Zaman S, Karray F. Feature selection for intrusion detection systems based on support vector machines. Consumer Communications and Networking Conference 6th IEEE; IEEE; 2009 Jan 10. p. 1–8.
22. Peddabachigari S, Abraham A, Grosan C, Thomas J. Modeling intrusion detection system using hybrid intelligent systems. Journal of network and computer applications. 2007 Jan 31; 30(1):114–32.
23. Kermansaravi Z, Jazayeriy H, Fateri S. Intrusion Detection System in Computer Networks using Decision Tree and SVM Algorithms. Journal of Advances in Computer Research. 2013 Aug 1; 4(3):83–101.
24. Hall MA. Correlation-based feature selection for machine learning (Doctoral dissertation, The University of Waikato).
25. Witten IH, Frank E. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann; 2005 Jul 13.
26. KDD Cup 1999 dataset [Online]. Available at: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, (Accessed 2014, Oct.)
27. Nsl-kdd data set for network-based intrusion detection systems [Online]. Available on: http://nsl.cs.unb.ca/KDD/NSL-KDD.html, (Accessed March 2014).
28. Maheyzah Md. Siraj, Hashim Hussein Taha Al basheer, Mazura Mat Din. Towards Predictive Real-time Multi-sensors Intrusion Alert Correlation Framework. Indian Journal of Science and Technology. 2015 June; 8(12). Doi no:10.17485/ijst/2015/v8i12/70658.
29. Staudemeyer, Ralf Colmar. The importance of time: Modelling network intrusions with long short-term memory recurrent neural networks. University of the Western Cape; PhD Thesis. 2012.