ASTES

Advances in Science, Technology and Engineering Systems Journal Vol. 5, No. 6, 370-390 (2020)

www.astesj.com

ASTESJ

ISSN: 2415-6698

Intrusion Detection and Classification using Decision Tree Based Key Feature Selection Classifiers

Manas Kumar Nanda[1,*], Manas Ranjan Patra[2]

[1]*Department of Computer Application, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, 751030, India*

[2]*Department of Computer Science, Berhampur University, Berhampur, 760007, India*

ARTICLE INFO

Article history:
Received: 26 August, 2020
Accepted: 31 October, 2020
Online: 20 November, 2020

Keywords:
Attribute
Confidentiality
Features
Intrusion
Rank
Spam

ABSTRACT

Feature selection method applied on an intrusion dataset is used to classify the intrusion data as normal or intrusive. We have made an attempt to detect and classify the intrusion data using rank-based feature selection classifiers. A set of redundant features having null rank value are eliminated then the performance evaluation using various feature selection algorithms are done to determine the behavior of attributes. We can distinguish the key features which plays an important role for detecting intrusions. There are 41 features in the dataset, out of which some features play significant role in detecting the intrusions and others do not contribute in the detection process. We have applied different feature selection techniques to select the predominant features that are actually effective in detecting intrusions.

1. Introduction

Intrusion Detection System (IDS) works as an application or device which identifies some hostile activities or as policy violations by the intruder in network. IDS is used to analyze the network traffic and detect some possible intrusive activities in the computer network. Mainly misuse detection system and anomaly detection system are the two types of intrusion detection systems. It is capable of detecting probable attacks from the known patterns or signatures, and identify some intrusive activities which deviates from normal behavior in a monitored system, and can detect some unknown attacks [1]. The most popular IDSs are SourceFire, McAfee, and Symantec, which plays an important role for network surveillance and monitoring, and functions like a network security guard. IDS can be categorized as Network based Intrusion Detection System (NIDS) and Host based Intrusion Detection System (HIDS) [2]. In NIDS, the Intrusion Detection System (IDS) is installed before and after the firewall to capture network traffic for the entire network segment, but in HIDS, the Intrusion Detection System (IDS) is applied on a specific host to analyse packets, logs and system calls. As compared to NIDS, HIDS is more suitable for identifying the internal attacks.

We have applied a number of techniques to analyze the intrusion data and build a system that has higher detection rate.

*Corresponding Author: Manas Kumar Nanda, SOA University, 9437296663 & manasnanda@soa.ac.in

2. Data Mining-based Approach

Data mining is a method of discovering a way of systematic relationship of data and an approach of determining the fundamental information of data. It is broadly divided into two categories such as supervised and unsupervised approach. Classifications and clustering are the best examples of supervised and unsupervised algorithms respectively. In a clustering approach, the group of unique objects are based on the characteristics of such data points [3]. Where these data points in a cluster is similar to other data points in the cluster and is dissimilar to the data points in different cluster. By grouping such similar data points into one cluster which shows the abnormality identification. Hence this approach may be responsible for potentially increase of the false alarm rate. The performance of IDS is highly dependent on the low false alarm rate, which may degrade the performance when it generate high false alarm [4]. Classification is one of the best supervised approach used for classifying the benign or anomalous data, for reducing the false alarm rate. It has the ability to differentiate unusual data pattern, which may be suitable for identifying new attack patterns [5]. Classification is widely used for its strong ability in identifying the normal structure very accurately, which contribute towards its reducing false detection [6]. These ensemble techniques are used to combine several classifiers which obtain better prediction for its accuracy in performance [7].

2.1. Classification

A classification technique (also known as classifier) is a systematic approach to build the classification models from an input dataset. Some of the techniques like Decision Tree based classifiers, Rule based classifiers, Neural Networks, Naïve Bayes classifiers and Support Vector Machines etc., each of the technique employs a learning-based algorithm to identify a model that best suits the relationship between the set of attributes and class label of input data. The generated model by learning algorithm should fit both input data well and also correctly predict class labels of records. The primary objective of the learning based algorithm is to build a model with good generalization and capability; i.e., the models that accurately predict the class labels of previously unknown records. This classification is done using a training set which consists of records whose class labels are known and must be provided. To build a classification model, using this is subsequently applied to the test set, which consists of records with unknown class labels.

The performance of a classification model is evaluated based on the classification of test records correctly and incorrectly predicted by the model. The counts of the predicted values are tabulated in a table known to be confusion matrix. For the learning and classification, we have used various machine learning techniques. There are two major categories of machine learning techniques, namely, Supervised and unsupervised technique, supervised technique requires an initial training phase where the algorithm is trained using existing dataset with appropriate classification. The algorithm then uses this knowledge to perform the real-time classification of test data. Conversely, the unsupervised technique does not require any existing classification method and basically use multiple runs to fine tune the classification patterns.

2.2. Decision Tree

Classifying the test record is a straightforward approach once a decision tree is being constructed. Basically starting from a root node, we go on applying the test condition to the records and follow to the appropriate branches based upon the outcome of the test result. This will lead to us either to the internal node, to which the new test condition is being applied or to the leaf node. This class label which is associated with the leaf node is then assigned to the record.

There are many decision trees, which can be constructed from a given set of attributes, where some of the trees are more accurate than others, and finding a optimal tree is computationally infeasible because of exponential size of the size of the search space. A number of algorithms have been developed to induce, with a reasonably accurate and albeit suboptimal decision tree constructed in a reasonable amount of time. Such algorithms employ a greedy strategy that design a decision tree by taking a series of locally optimum decisions about which, attributes are to be used for partitioning the data. One of such algorithm is Hunt's algorithm, which is the basis of most of the existing decision tree induction algorithms, including, C4.5, Classification and Regression Trees (CART), and Iterative Dichotomiser 3 (ID3). An efficient algorithm to build a decision tree is C4.5, used for classification (also known as statistical classifier), can be described as "a landmark of decision tree program that is probably the machine learning workhorse and most widely used in practice to date".

2.2.1. Random Forest

The most versatile tree-based machine learning algorithm, which is used to build several trees (or decision trees), and then combining each of the output to improve the generalization ability of the building model. This method of combining the trees (i.e combining weak learners (or individual trees) to produce a strong learner (a Forest)) is also known as an ensemble method. Random Forest algorithm can be used to solve the regression problems (where the dependent variables are continuous) and classification problems (where the dependent variables are categorical).

In a given data frame, the tree stratifies or partitions the data, based on rules (such as if-else), then these rules divide the dataset into a number of distinct and non-overlapping regions. Such rules are determined by the use of variable's contribution to the homogeneity or pureness of the respective resultant child nodes. In the regression trees (where the output is predicted by the mean of observations at the terminal nodes), the splitting decision is based on minimizing the RSS. The variable, which leads to the greatest possible reduction in RSS, is chosen in the root node. The tree splitting takes a top-down greedy approach (which is also known as recursive binary splitting) , because the algorithm cares to make the best split, at the current step rather than saving a split for better result on future nodes.

2.3. Feature Selection

Feature selection is a process of selecting a subset of M features from a set of N features, so that the feature space is optimally reduced based on a certain evaluation criteria. The objective of feature subset selection method is to find a optimum set of features such that, the resulting probability distribution of the data classes is as close as possible to the original distribution obtained using all the attributes [8]. Feature selection process used to improve the classification performance by searching for a subset of features, which best classifies the training data. In a high dimensional feature space some of the features may be redundant or irrelevant, which may deteriorate the performance of classifiers [9]. It is very much important to remove these redundant or irrelevant features. It is necessary to find the subset of features involves in the selection process to improve the prediction of accuracy or decrease the size of the features in the dataset without significantly decreasing the prediction accuracy of a classifier which is built using only the selected features.

Feature selection is the foundation of machine learning [10], a process of discovering the most useful and prominent features for the learning-based algorithm. It is very much important to extract the set of redundant or irrelevant features need to prevent the classifiers from being biased and required to minimize the feature selecting error so as to improve the abnormal behavior and detection rate. This is because of the application of appropriate algorithm and its effectiveness highly dependent on the feature selection process. Filters and wrappers [11] are the two generalized methods used in feature selection. Filter-based subset evaluation (FBSE) method is used to remove the redundant features inside the filter ranking [12]. This process examines the complete subset in a multivariate way and select the relevant features and explores the degree of relationship between the features.

Henceforth FBSE is a heuristics approach which involves the probability and statistical measures for searching and evaluation of the usefulness of identified features. The wrapper based subset evaluation (WBSE) method use the classifier for estimating the worth of each feature subset. Basically WBSE methods have greater predictive accuracy as compared to FBSE method, because the selection approach is optimized, when evaluating each of the feature subset using classification algorithm. So more or less the WBSE method use a classification based algorithm for evaluation of each set of features. But WBSE method becomes uncontrollable, at the time of dealing with large databases with many features [13]. Hence, WBSE methods are highly associated with the classifier algorithm which makes it more difficult at the time of shifting from one classifier to another classifier because of the re-initiation of selection process. But the FBSE method uses distance measures and correlation functions for selection criteria of features [14]. Where FBSE method do not need re-execution of different learning based classifiers and hence its execution process is more faster than WBSE. So FBSE is favorable for large databases environment which contains many features.

The statistical based detection approach introduced by [15] based on the collection of data for creating normal behavior profile. Here traffic data over a period of time is collected for utilization of intrusion detection. In Packet Header Anomaly Detection (PHAD), the abnormal patterns are recognized using the packet characteristics and behaviors. The normal profile is constructed using statistical measurement of activity history[16]. A set of traffic can be defined as intrusive, that are deviated from normal profile and behaved abnormally. PHAD uses all of the 33 attributes of a packet header that represents information of data link, network and transport layers of 7 layer OSI model without using IP address and port number, The probability of each packet being benign or tending towards abnormal behavior is measured by the information contain in each attribute. For any such dissimilarity detected at the time of matching against the training data, an anomaly score is given. The anomaly score for each packet is summed-up, and if the score surpasses the preset threshold then it is flagged as anomalous.

Network-based and host-based are two different environments in Protocol based Packet Header Anomaly Detection (PhPHAD) of conventional PHAD system [17]. TCP, UDP and ICMP are the three main protocols used to construct normal profile. The Light weight Network Intrusion Detection System (LNID) has been proposed to identify the malicious packets in Telnet traffic. In LNID, the behavior is extracted from the training data to construct the normal profile which is further used for computing the anomaly score. This anomaly score is used to match between training and testing data. Then surpassed preset threshold score packets are treated as malicious packets. To reduce computational cost the insignificant features from the training data are removed during preprocessing phase.

Rank based feature selection: Feature ranking method calculate the score of each attribute and arrange them in descending order according to their score. The performance of six ranking methods used for feature selection is divided into entropy based attribute evaluator and statistical attribute evaluator technique. Entropy based attribute evaluator technique is used in information theory to characterize the purity of an arbitrary collection of samples. Information Gain (IG), Gain Ratio (GR) and Symmetrical Uncertainty (SU) are the entropy based attribute evaluators used to measure system's unpredictability. Whereas One Rule (One R), Chi Squares and Relief-F Attribute evaluators are statistical attribute evaluator techniques.

3. Rank-based Classification

We have used most of the efficient data mining classification algorithms used for IDS. The Best-First Decision Tree based (BFT) classifier, basically used for binary splitting of both normal and numeric valued attributes, the decision tree learner based on imprecise probabilities and uncertainty measures (CDT), the class implementing decision Forest Algorithm (FPA) using bootstrap samples and penalized attributes [18] , the building of Functional Trees (FT) for classification, more specifically functional trees uses logistic regression based functions at inner nodes and leaves. This algorithm can also deal with the binary as well as multiclass target variables, along with nominal attributes and numeric and with missing values [19]. A Hoeffding Tree (VFDT) is incremental based, anytime decision-tree induction algorithm, which is capable of learning from a massive data stream. Hoeffding trees exploit the fact that, an optimal splitting attribute can often be chosen from a small sample [20], the class for generating a pruned or unpruned C4 (J48) [21] and the class for generating a pruned or unpruned C4.5 Consolidated Tree Construction (CTC) algorithm (J48Consolidated) in which a set of subsamples are used to build a single tree, where the Resampling Method (RM) [22] is built with a few new options added to the J48 class, whereas the class for generating a grafted (pruned or unpruned) C4 (J48graft) [23] and the class for generating a multi-class alternating decision tree using LogitBoost strategy (LADTree) [24]. The classifier for building a Logistic Model Trees (LMT), in which the classification trees are the logistic regression functions of the leaves. This algorithm can also deal with the binary as well as multiclass target variables, along with nominal attributes and numeric and with missing values [25]. The class for generating decision tree with Naïve Bayes classifiers at the leaves (NBTree) [26]. A class for constructing a forest with Random Trees (RF) [27] and the class considers K number of randomly chosen attributes at each level of node (RT) for constructing a tree. The. Fast decision tree learner (REPT) which builds a decision tree or regression tree using the information gain or variance and prunes it using the reduced error pruning method with backfitting, and the Implementation of the decision forest algorithm SysFor (SF) [28].

4. Dataset used

The KDDCUP99 dataset is derived from the DARPA98 network traffic data in 1999, which assembled individual TCP packets into TCP connections. Each of the TCP connection having 41 features along with a label that specifies a specific type of attack or normal as a status of a connection. Dataset consists of 38 numeric features and three symbolic features, which are again classified into following four different categories [29]. First nine (f1-f9) features are used to describe each TCP connection. In this category all the attributes are being extracted from a TCP/IP connection, and these features lead to an implicit-delay in detection. The second thirteen (f10-f22) are domain knowledge related content features used to indicate that suspicious behavior in the network traffic having no sequential patterns. But unlike

Table 1: The Various types of Attacks and their Classifications

Attack Category/ Attack Name		Attack Description
Denial of Service (DoS)		In such attack an attacker tries to make the system's computing/ memory resources too busy or full to handle the legitimate requests, or denies such legitimate users to access a system. The most possible ways to launch the DoS attacks are by abusing the computers for legitimate features, and by targeting the bugs, or by exploiting the system's misconfiguration.
	Back	A DoS attack against the apache web server, in which a client requests URL containing many backslashes, that slows down server response
	Land	A DoS attack where remote host sends a UDP packet with same source and destination, freezes the machine
	Neptune	Syn-flood DoS attack on one port or on more ports
	Ping of Death	DoS ping-of-death
	Smurf	A DoS attack in which a large number of ICMP (Internet Control Message Protocol) packets with its intended victim's spoofed source IP may broadcast to computer network using IP broadcast address. By which the victim's computer may slow down for devices on a network intending to send a reply to the source IP, in a flooded traffic of large no of packets.
	Teardrop	A program, sends IP fragments to a machine which is connected to a network or internet. It is a DoS attack that exploit an overlapping IP fragment bug which is present in Windows 95, Windows NT and Windows 3.1 Operating System Machines. This bug causes TCP/ IP fragmentation reassembly code for improperly handle overlapping IP fragments, which needs a reboot for preferred remedy.
Remote to Local (R2L)		In such attack an attacker without having a registered account in a remote machine, that send packets to machine on a network and exploits the vulnerability for illegally gain local access as a user on that machine.
	Ftp_write	The remote FTP user creates .rhost file in the world writeable anonymous FTP directory that obtains local login which gains user access.
	Guess_passwd	An attacker tries to gain access to the user account, by repeatedly guessing possible passwords.
	Imap	A remote buffer-overflow using Imap port that leads to root shell which gains root access.
	Multihop	Multi-day scenario by which a user first breaks into a machine.
	Phf	The exploitable CGI script that allows a client to execute the arbitrary commands on a machine using misconfigured web server.
	Spy	That sends packets to a machine over a network through it which doesn't have an account in the target machine.
	Warezclient	The user used to download illegal software that was previously posted using anonymous FTP by the warezmaster
	Warezmaster	Exploits a system bug associated with FTP Server.
User to Root (U2R)		Using such attack, an attacker used to access to a normal user account attempts to exploit system vulnerabilities to gain root access to the system. A class of such attacks are the regular buffer overflows, that are caused by the regular programming mistakes and the environmental assumptions.
	Buffer_overflow	Such type of attacks are designed to trigger the arbitrary code execution using a program and by sending it to more than that it supposed to receive.
	Load Module	Non-stealthy load module attack, that resets IFS for the normal user and that creates a root shell.
	Perl	The perl attack sets a user-ID to root in a perl script and it creates a root shell.
	Root kit	A Multi-day scenario where the user installs one or more components of a rootkit.
Probes		In such attack an attacker who scans network of computers to get information or to find known vulnerabilities. An attacker with the information of map of machines and services over a network that can be used to exploit. There are different types of probes, few of them abuse computer's legitimate features or social engineering techniques. These class of attacks are the most commonly heard, which requires little technical knowledge or expertise.
	Satan	A publicly available tool which probes the network for security vulnerabilities and for misconfigurations.
	Ipsweep	The surveillance sweep is performing either a port sweep or ping on multiple host which identifies the active machine.
	Nmap	Network mapping using nmap tool, identifies active ports on a machine.
	Portsweep	The Surveillance sweep is performing either a port sweep or ping on multiple host address.

DOS and Probing attacks, the R2L, and U2R attacks don't contain any intrusion frequent sequential patterns, because DoS and Probing attacks involve many connections to the host(s) in a short period of time, so the R2L and U2R attacks are embedded with the data portions of packets and normally involve a single connection. Hence, we need some features to detect such attacks, which may look in the data portion for the suspicious behavior, (as an example a number of failed login attempts) are called content features. The third nine (f23-f31) time-based traffic features are designed to capture the properties that mature over a two second temporal window, (as an example the number of connections to the same host over a two second interval). Final fourth ten (f32-f41) host-based traffic features basically utilize a historical window, which are estimated over a number of connections instead of time, such features are designed to access attacks, with span intervals are longer than 2 seconds [30]. There are 41 feature attributes for each connection record plus one class label, out of which 38 are numeric and 3 are symbolic. Symbolic attributes are protocol type, service, and flag. Discrete data can be numeric but it can also be categorical, continuous data are always numeric and are not restricted to define separate values and can take any value within a range [31]. The various types of attack with their categories are discussed in the table [32]. The severity of the attack and the classification is discussed.

There are 125973 number of records of the dataset is; out of which 53.48 percent are normal and 46.52 percent of records are of intrusive types. There are 24 different types of attack which can be mainly classified into four categories, such as Danial of Services (DoS), Remote to Local (R2L), User to Root (U2R), and Probing.

Table 2: Different Types of Attack and their Class Occurrences

Class Type		Number of Instances	% of Attack Type	% of Attack Class	% of Total Class Occurences
DoS		45927	-	78.33	36.45
1	neptune	41214	89.74	70.3	32.72
2	teardrop	892	1.94	1.52	0.71
3	smurf	2646	5.76	4.51	2.1
4	pod	201	0.44	0.34	0.16
5	back	956	2.08	1.63	0.76
6	land	18	0.04	0.03	0.01
Probes		11656	-	19.88	9.25
1	ipsweep	3599	30.88	6.14	2.86
2	portsweep	2931	25.15	5	2.33
3	nmap	1493	12.81	2.55	1.19
4	satan	3633	31.17	6.2	2.88
R2L		995	-	1.7	0.78
1	warezclient	890	89.45	1.52	0.71
2	guess_passwd	53	5.33	0.09	0.04
3	ftp_write	8	0.8	0.01	0.01
4	multihop	7	0.7	0.01	0.01
5	imap	11	1.11	0.02	0.01
6	warezmaster	20	2.01	0.03	0.02
7	Phf	4	0.4	0.01	0
8	Spy	2	0.2	0	0
U2R		52		0.09	0.04
1	rootkit	10	19.23	0.02	0.01
2	buffer_overflow	30	57.69	0.05	0.02
3	loadmodule	9	17.31	0.02	0.01
4	Perl	3	5.77	0.01	0
Total Attacks		58630	-	-	46.52
Normal		67343	-	-	53.48
Total Instances		125973			100

In the intrusive data set around 36.45 percent data are of DoS type, about 9.25 percent data are of Probes type, 0.78 percent of R2L type and 0.04 percent of U2R type.

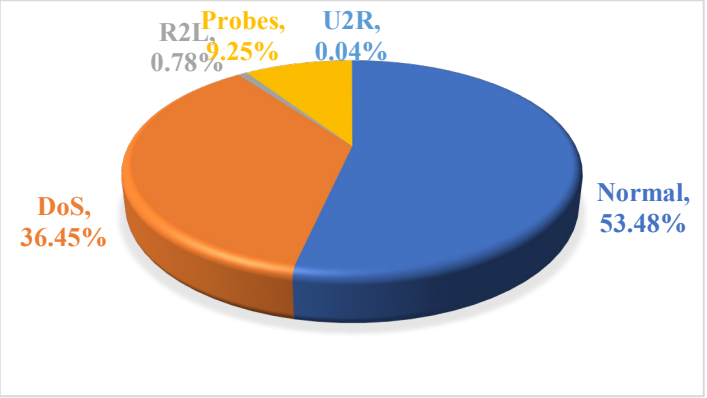

Figure 1: Percentage of Instances of Various Attack Classes

The figure describes various attack categories, around 78.33 percent of data are of DoS type out of which 70.3 percent are of Neptune type, 1.52 percent teardrop, 4.51 percent smurf, 0.34 percent pod, 1.63 percent pod and 0.03 percent of land type attacks. There are 19.88 percent Probes type, where 6.14 percent are ipsweep, 5 percent are portsweep, 2.55 percent are nmap and 6.2 percent are satan type of attack. Nearly 1.7 percent are R2L type of attack, in which 1.52 percent are warezclient, 0.09 percent are guess_passwd, 0.01 percent are ftp_write, 0.01 percent are multihop, 0.02 percent are imap, 0.03 percent are warezmaster, 0.01 percent are phf, and 0.0 percent are spy. In the 0.09 percent of U2R attack, there are 0.02 percent rootkit, 0.05 percent are buffer_overflow 0.05 percent are loadmodule, and there are 0.01 percent perl type of attack.We have applied the rank based feature selection method to compute the rank of the features and the order of the features based on their rank is as follows. The order of the features varies with the different feature selection approach. The rank based feature selection approach applied on the various set of data set of NSL KDD Cup data set is as follows. The rank of the features varies with the

change of rank based feature selection approach applied on various classes of data. The order of the features varies based on their rank values and feature selection mechanism. The rank for different features varies based on their contribution in the selection process is mentioned below. The rank value signify the contribution of the feature to the different class of data.

With the application of rank based feature selection approach, we found the rank value of the features in the above mentioned table is '0' (Null), So these features do not contribute in the process of intrusion classification. The rest of the features mentioned in the below table are responsible in the classification of the Intrusion data.

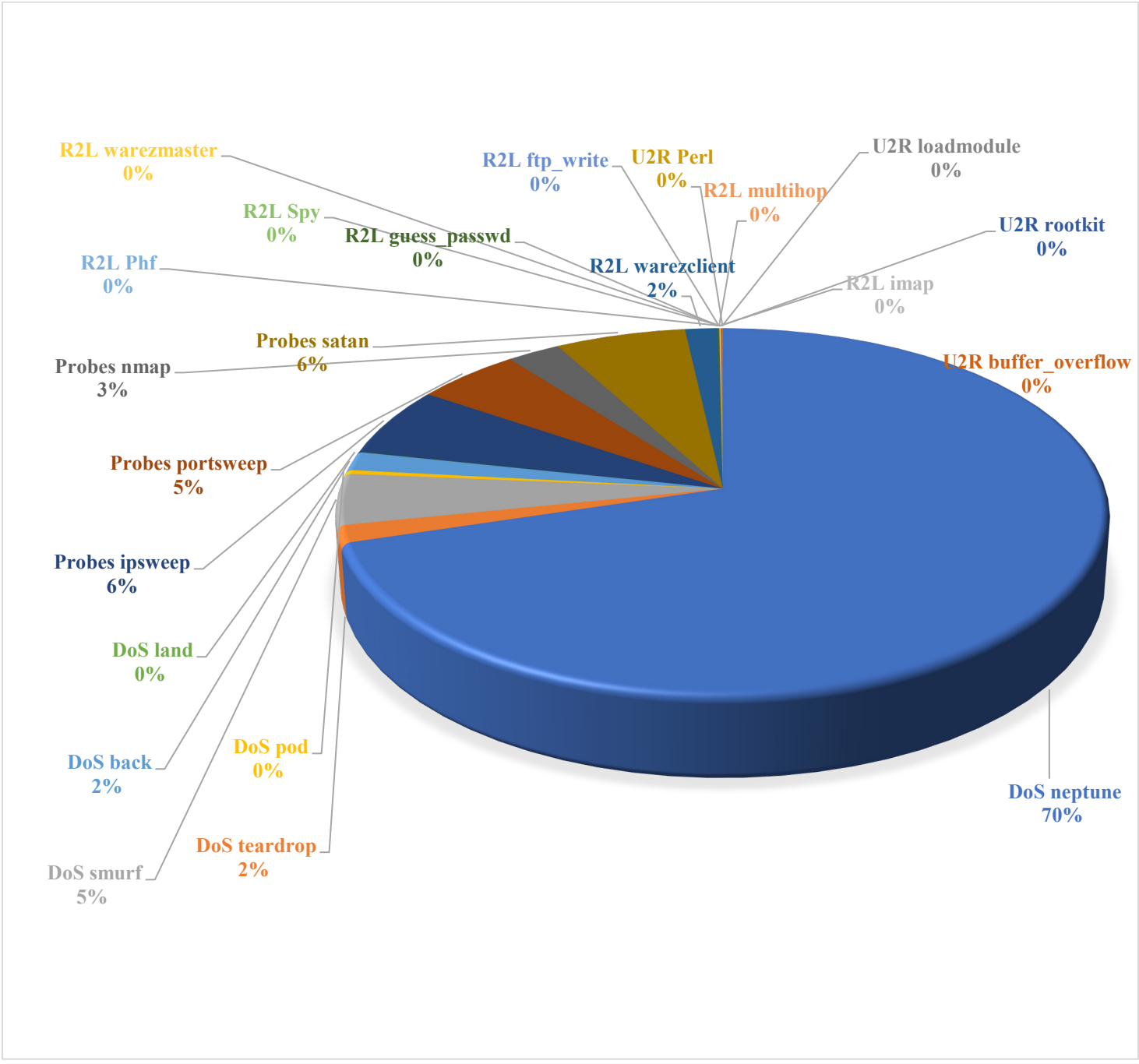

Figure 2: Percentage of Instances of Various Attack Types

Table 3: Number of features do not Contribute in Selection Process

Feature Selection Approach	Number of Features	Feature Selection	Name of Feature	Rank Value
NSL KDD'99 (All)	2	f20, f21	Nu_ob, Is_ho_lg	0
NSL KDD'99 (DOS)	11	f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	N_f_login, Nu_ob, Is_gu_lg, N_shell, Nu_ac_fl, Num_f_cr, R_shell, Urgent, Su_attem, Num_roo, Is_ho_lg	0
NSL KDD'99 (PROBES)	14	f7, f20, f9, f22, f8, f19, f18, f11, f17, f16, f15, f14, f13, f21	Land, Nu_ob, Urgent, Is_gu_lg, W_frag, Nu_ac_fl, N_shell, N_f_login, Num_f_cr, Num_roo, Su_attem, R_shell, Num_com, Is_ho_lg	0
NSL KDD'99 (R2L)	8	f29, f2, f20, f30, f15, f7, f8, f21	Sa_srv_rt, Pro_type, Nu_ob, Di_srv_rt, Su_attem, Land, W_frag, Is_ho_lg,	0
NSL KDD'99 (U2R)	27	f15, f14, f13, f41, f11, f9, f8, f7, f21, f19, f33, f30, f31, f37, f20, f38, f39, f29, f28, f27, f26, f40, f22, f23, f24, f25, f1	Su_attem, R_shell, Num_com, D_hsrr, N_f_login, Urgent, W_frag, Land, Is_ho_lg, Nu_ac_fl, Ds_ho_sr, Di_srv_rt, Sr_di_h0, Ds_d_h_rt, Nu_ob, D_h_sr, Ds__h_r, Sa_srv_rt, Sr_rr_rt, Rer_rt, Se_se_rt, Ds_hrr, Is_gu_lg, Count, Sev_coun, Ser_rate, Duration	0

Table 4 : Number of Features do not Contribute in DOS Attack

	Features do not Contribute			Features Contribute			
	Number	Features	Value	Number	Features	Name	Key Feature
NSL KDD'99 (All)	2	f20, f21	0	39	f5, f4, f2, f3, f29, f36, f30, f24, f35, f34, f23, f33, f38, f39, f25, f26, f40, f8, f41, f12, f6, f10, f13, f32, f28, f27, f31, f37, f7, f1, f11, f22, f18, f19, f17, f14, f9, f15, f16	-	0
NSL KDD'99 (DOS)	11	f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	0	30	f5, f4, f2, f3, f29, f36, f30, f24, f35, f34, f23, f33, f38, f39, f25, f26, f40, f8, f41, f12, f6, f10, f13, f32, f28, f27, f31, f37, f7, f1	-	0
NSL KDD'99 (DOS-neptune)	15	f8, f10, f13, f7, f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	0	26	f5, f4, f2, f3, f29, f36, f30, f24, f35, f34, f23, f33, f38, f39, f25, f26, f40, f41, f12, f6, , f32, f28, f27, f31, f37	-	0
NSL KDD'99 (DOS-teardrop)	21	f39, f26, f41, f12, f10, f13, f28, f31, f37, f7, f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	0	20	f5, f4, f2, f3, f29, f36, f30, f24, f35, f34, f23, f33, f38, f25, , f40, f8, , f6, f32, f27, , f1	Src_bytes	f5
NSL KDD'99 (DOS-smurf)	26	f30, f39, f25, f26, f8, f41, f12, f6, f10, f13, f28, f27, f31, f37, f7, f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	0	15	f5, f4, f2, f3, f29, f36, f30, f24, f35, f34, f23, f33, f38, f39, f25, f26, f40, f8, f41, f12, f6, f10, f13, f32, f28, f27, f31, f37, f7, f1, f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	Same_srv_rate	f29
NSL KDD'99 (DOS-pod)	23	f30, f39, f25, f26, f41, f12, f6, f10, f13, f28, f27, f7, f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	0	18	f5, f4, f2, f3, f29, f36, f24, f35, f34, f23, f33, f38, , f40, f8, f32, f31, f37, f1,	Same_srv_rate	f29
NSL KDD'99 (DOS-back)	15	f35, f8, f37, f7, f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	0	26	f5, f4, f2, f3, f29, f36, f30, f24, f34, f23, f33, f38, f39, f25, f26, f40, f41, f12, f6, f10, f13, f32, f28, f27, f31, f1	Logged_in	f12

NSL KDD'99 (DOS-land)	19	f5, f8, f41, f12, f6, f10, f13, f28, f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	0	22	f4, f2, f3, f29, f36, f30, f24, f35, f34, f23, f33, f38, f39, f25, f26, f40, f32, f27, f31, f37, f7, f1	Land	f7

Table 5: Number of Features do not Contribute in U2R Attack

Feature Selection Approach	Features do not Contribute			Features Contribute			
	Number	Features	Value	Number	Features	Key Feature	Name
NSL KDD'99 (All)	2	f21, f20	0	39	f35, f34, f32, f36, f18, f17, f6, f5, f10, f16, f12, f3, f2, f4, f15, f14, f13, f41, f11, f9, f8, f7, f19, f33, f30, f31, f37, f38, f39, f29, f28, f27, f26, f40, f22, f23, f24, f25, f1	-	Nu_ob, Is_ho_lg
NSL KDD'99 (U2R)	27	f15, f14, f13, f41, f11, f9, f8, f7, f21, f19, f33, f30, f31, f37, f20, f38, f39, f29, f28, f27, f26, f40, f22, f23, f24, f25, f1	0	14	f35, f34, f32, f36, f18, f17, f6, f5, f10, f16, f12, f3, f2, f4	f12, f29	Logged_in, Same_srv_rate
NSL KDD'99 (U2R-buffer_overflow)	15	f35, f18, f15, f11, f9, f8, f7, f21, f19, f31, f20, f38, f39, f26, f22,	0	26	f34, f32, f36, f17, f6, f5, f10, f16, f12, f3, f2, f4, f14, f13, f41, f33, f30, f37, f29, f28, f27, f40, f23, f24, f25, f1	f12	Logged_in
NSL KDD'99 (U2R_loadmodule)	16	f15, f11, f9, f8, f7, f21, f31, f20, f38, f39, f28, f27, f26, f40, f22, f25	0	25	f35, f34, f32, f36, f18, f17, f6, f5, f10, f16, f12, f3, f2, f4, f14, f13, f41, f19, f33, f30, f37, f29, f23, f24, f1	f12	Logged_in
NSL KDD'99 (U2R_perl)	21	f10, f15, f13, f41, f11, f9, f8, f7, f21, f19, f30, f31, f37, f20, f38, f39, f28, f27, f26, f22, f25	0	20	f35, f34, f32, f36, f18, f17, f6, f5, f16, f12, f3, f2, f4, f14f33, f29, f40, f23, f24, f1	f12	Logged_in
NSL KDD'99 (U2R_rootkit)	15	f18, f15, f8, f7, f21, f19, f30, f31, f20, f38, f28, f27, f26, f22, f25	0	26	f35, f34, f32, f36, f17, f6, f5, f10, f16, f12, f3, f2, f4, f14, f13, f41, f11, f9, f33, f37, f39, f29, f40, f23, f24, f1	f29	Same_srv_rate

Table 6: Number of Features do not Contribute in R2L Attack

Feature Selection Approach	Features do not Contribute			Features Contribute			
	Number	Features	Value	Number	Features	Key Feature	Name
NSL KDD'99 (All)	2	f20, f21	0	39	f6, f5, f3, f12, f39, f28, f4, f10, f11, f38, f40, f41, f36, f27, f1, f32, f33, f35, f37, f34, f19, f31, f26, f25, f24, f17, f23, f14, f22, f16, f13, f18, f9, f29, f2, f30, f15, f7, f8	-	-
NSL KDD'99 (R2L)	8	f29, f2, f20, f30, f15, f7, f8, f21	0	33	f6, f5, f3, f12, f39, f28, f4, f10, f11, f38, f40, f41, f36, f27, f1, f32, f33, f35, f37, f34, f19, f31, f26, f25, f24, f17, f23, f14, f22, f16, f13, f18, f9	-	-
NSL KDD'99 (R2L-ftp_write)	17	f39, f28, f11, f38, f40, f41, f27, f26, f25, f14, , f18, f20, f30, f15, f7, f8, f21	0	24	f6, f5, f3, f12f4, f10, f36, f1, f32, f33, f35, f37, f34, f19, f31, f24, f17, f23, f22, f16, f13, f9, f29, f2	f29	Same_srv_rate

NSL KDD'99 (R2L_guess_passwd)	15	f35, f19, f31, f17, f14, f16, f13, f18, f9, f20, f30, f15, f7, f8, f21	0	26	f6, f5, f3, f12, f39, f28, f4, f10, f11, f38, f40, f41, f36, f27, f1, f32, f33, f37, f34, f26, f25, f24, f23, f22, f29, f2, f20	f29	Same_srv_rate
NSL KDD'99 (R2L_imap)	16	f11, f41, f27, f37, f19, f17, f14, f22, f18, f9, f20, f30, f15, f7, f8, f21	0	25	f6, f5, f3, f12, f39, f28, f4, f10, f38, f40, f36, f1, f32, f33, f35, f34, f31, f26, f25, f24, f23, f16, f13, f29, f2	f29	Same_srv_rate
NSL KDD'99 (R2L_multihop)	16	f39, f28, f11, f38, f41, f27, f37, f26, f25, f9, f20, f30, f15, f7, f8, f21	0	25	f6, f5, f3, f12, f4, f10, f40, f36, f1, f32, f33, f35, f34, f19, f31, f24, f17, f23, f14, f22, f16, f13, f18, f29, f2	f29	Same_srv_rate
NSL KDD'99 (R2L_phf)	22	f39, f11, f38, f40, f41, f36, f27, f37, f26, f25, f17, f22, f16, f13, f18, f9, f20, f30, f15, f7, f8, f21	0	19	f6, f5, f3, f12, f28, f4, f10, f1, f32, f33, f35, f34, f19, f31, f24, f23, f14, f29, f2	f29	Same_srv_rate
NSL KDD'99 (R2L_spy)	21	f28, f10, f11, f40, f41, f36, f27, f37, f31, f26, f25, f14, f22, f16, f13, f9, f20, f30, f7, f8, f21	0	20	f6, f5, f3, f12, f39, f4, , f38, f1, f32, f33, f35, f34, f19, f24, f17, f23, f18, f29, f2, f15	f29	Same_srv_rate
NSL KDD'99 (R2L_warezclient)	13	f11, f19, f17, f14, f16, f13, f18, f9, f20, , f15, f7, f8, f21	0	28	f6, f5, f3, f12, f39, f28, f4, f10, f38, f40, f41, f36, f27, f1, f32, f33, f35, f37, f34, f31, f26, f25, f24, f23, f22, , f29, f2, f30	f12	Logged_in
NSL KDD'99 (R2L_warezmaster)	21	f39, f28, f11, f41, f27, f37, f19, f31, f26, f25, f14, f16, f13, f18, f9, f20, f30, f15, f7, f8, f21	0	20	f6, f5, f3, f12, f4, f10, , f38, f40, f36, f1, f32, f33, f35, f34, f24, f17, f23, f22, f29, f2	f29	Same_srv_rate

Table 7: Number of features do not Contribute in PROBES Attack

Feature Selection Approach	Features do not Contribute			Features Contribute			
	Number	Features	Value	Number	Features	Key Feature	Value
NSL KDD'99 (All)	2	f20, f21	0	39	f32, f35, f34, f37, f23, f27, f4, f40, f5, f2, f3, f30, f29, f36, f33, f31, f41, f28, f24, f38, f25, f1, f26, f39, f6, f12, f10, f7, f9, f22, f8, f19, f18, f11, f17, f16, f15, f14, f13,	-	-
NSL KDD'99 (PROBES)	14	f7, f20, f9, f22, f8, f19, f18, f11, f17, f16, f15, f14, f13, f21	0	27	f32, f35, f34, f37, f23, f27, f4, f40, f5, f2, f3, f30, f29, f36, f33, f31, f41, f28, f24, f38, f25, f1, f26, f39, f6, f12, f10	-	-
NSL KDD'99 (PROBES_nmap)	20	f27, f40, f41, f28, f12, f10, f7, f20, f9, f22, f8, f19, f18, f11, f17, f16, f15, f14, f13, f21	0	21	f32, f35, f34, f37, f23, f4, f5, f2, f3, f30, f29, f36, f33, f31, f24, f38, f25, f1, f26, f39, f6	-	-
NSL KDD'99 (PROBES_ipsweep)	13	f26, f10, f7, f20, f9, f22, f8, f19, f18, f11, f15, f14, f21	0	28	f32, f35, f34, f37, f23, f27, f4, f40, f5, f2, f3, f30, f29, f36, f33, f31, f41, f28, f24, f38, f25, f1, f39, f6, f12f17, f16, f13	-	-
NSL KDD'99 (PROBES_portsweep)	14	f7, f20, f9, f22, f8, f19, f18, f11, f17, f16, f15, f14, f13, f21	0	27	f32, f35, f34, f37, f23, f27, f4, f40, f5, f2, f3, f30, f29, f36, f33, f31, f41, f28, f24, f38, f25, f1, f26, f39, f6, f12, f10	-	-
NSL KDD'99 (PROBES_satan)	9	f7, f20, f9, f8, f19, f18, , f15, f14, f21	0	32	f32, f35, f34, f37, f23, f27, f4, f40, f5, f2, f3, f30, f29, f36, f33, f31, f41, f28, f24, f38, f25, f1, f26, f39, f6, f12, f10, f22, f11, f17, f16f13	-	-

5. Experimental Setup

We have tested the various machine learning methods on KDDCUP'99 dataset. In this experiment we have used a computing environment of core i7 processor, 2.6 GHz, 8 GB RAM, 1TB hard disk and windows 10 (64 bit) operating system. The various tree based classifiers have been used to classify different types of DOS, Probes, U2R and R2L attacks.

Table 8: Classification of Various DOS Attacks Using Different Classifiers

Sl. No.	Classifier	Classified Percentage	Unclassified Percentage
1	BFT	99.9935	0.0065
2	CDT	99.9956	0.0044
3	FPA	99.9913	0.0087
4	FT	99.9978	0.0022
5	HT	99.9673	0.0327
6	J48	99.9826	0.0174
7	J48C	99.9935	0.0065
8	J48G	99.9804	0.0196
9	LADT	99.9826	0.0174
10	LMT	99.9978	0.0022
11	NBT	99.9956	0.0044
12	RF	99.9913	0.0087
13	RT	99.9760	0.0240
14	REPT	99.9956	0.0044
15	SF	99.9826	0.0174
16	Min	99.9673	0.0022
17	Max	99.9978	0.0327
18	Avg	99.9882	0.0118

The result shows that FT and LMT has classified about 99.9978 percent. The performance of the tree based classifier has been observed on the KDD CUP'99 DOS attack type dataset. The classified result has been plotted in the below mentioned graph.

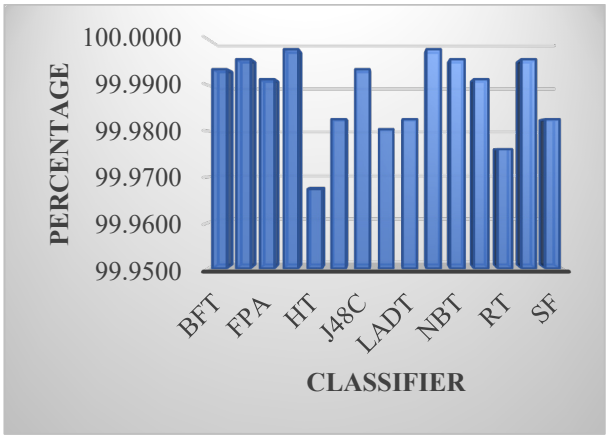

Figure 3: The Classification of Various Attack Type of DOS Attack

We have observed the result of the various classifiers applied on the rank-based feature selection methods. Information Gain (IG), Gain Ratio (GR) and Symmetrical Uncertainty (SU) feature selection methods have been applied to measure the performance and classification of various attack type. The accuracy and classification rate of few tree based classification algorithm is very high. The performance of the various algorithms and the classification percentage is reflected in the plotted graphs.

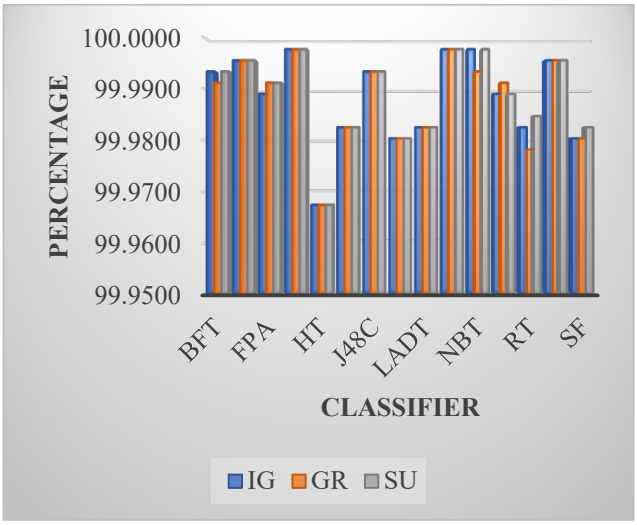

Figure 4: The Classification of Various Attack Type of DOS Attack Using Feature Selection

Further in the rank-based feature selection classification process we have applied a range of features to monitor the change in classification process. We have selected a number of features from the set of 41 features in the dataset and found there are 30 number of features participating or contributing in the classification process. We have applied various tree based classification algorithms to classify the NSL-KDDCUP'99 (DoS) Dataset and found the rank of the features are contributing in the classification process. There are 11 number of features whose rank value is null, which are not participating in the classification process. The performance of the J48 algorithms and the classification percentage is reflected in the plotted graphs.

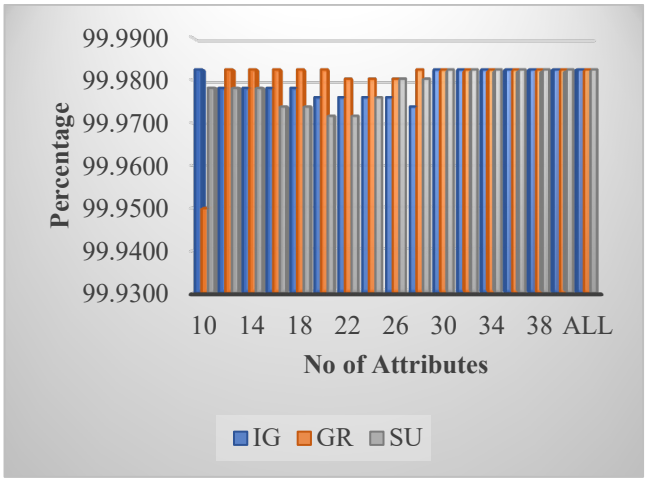

Figure 5: The Classification of Various Attack Type of DOS Attack Using Feature Selection and Attribute Selection

We have applied tree based feature selection classification technique on the NSL KDD'99 data set, The null valued redundant features are eliminated and the classification technique is applied to compute the percentage of accuracy of different attack types.

We have applied the J48, Random Forest and Functional Trees which are one the best decision tree based classification algorithm classifies the optimum accuracy of the different types of attack.

The below mentioned features are eliminated and the other non-null features are used to classify the different classes of intrusive data. The percentage of classification of different classes of attack are mentioned below.

The graph reflects the percentage of classification of different types of DOS attacks. The redundant null valued features are eliminated and the most suited tree-based algorithms are applied to find the optimum classified result.

The graph reflects the percentage of total classification of various DOS attacks. The redundant null valued features are eliminated and the most suited tree based algorithms are applied to find the optimum classified result.

In the classification process there are a number of key features which plays an important role to classify and determine the various attack types in the class of intrusive dataset. We have applied the same algorithms to determine the percentage of classification of various attack types using the key features.

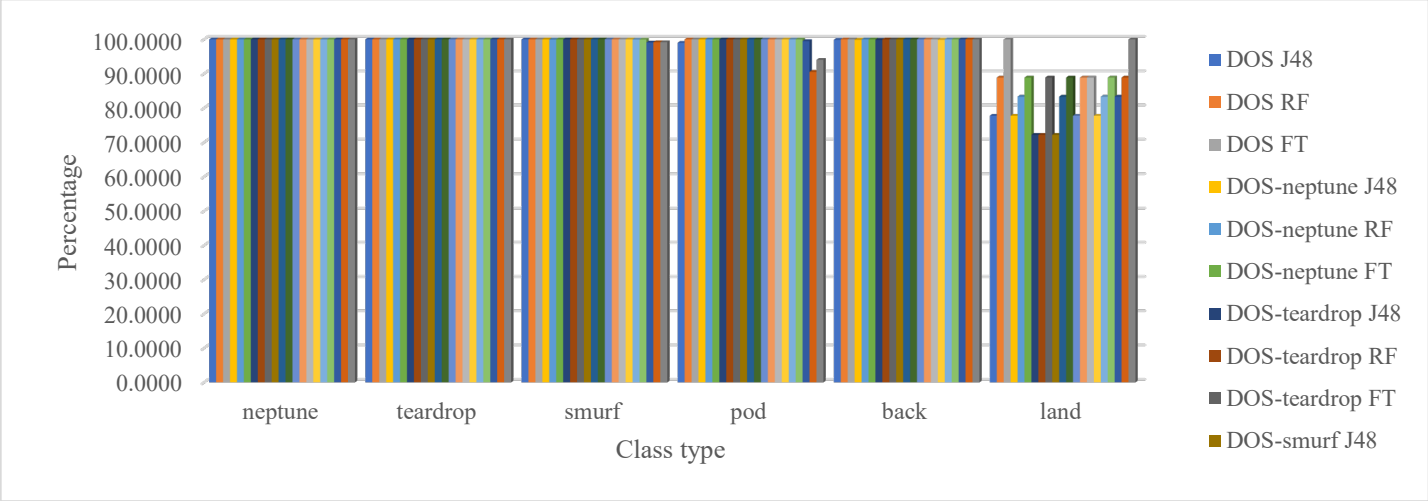

Figure 6: The Classification of Various Attack Class Types of DOS Attack by Removing Redundant Features

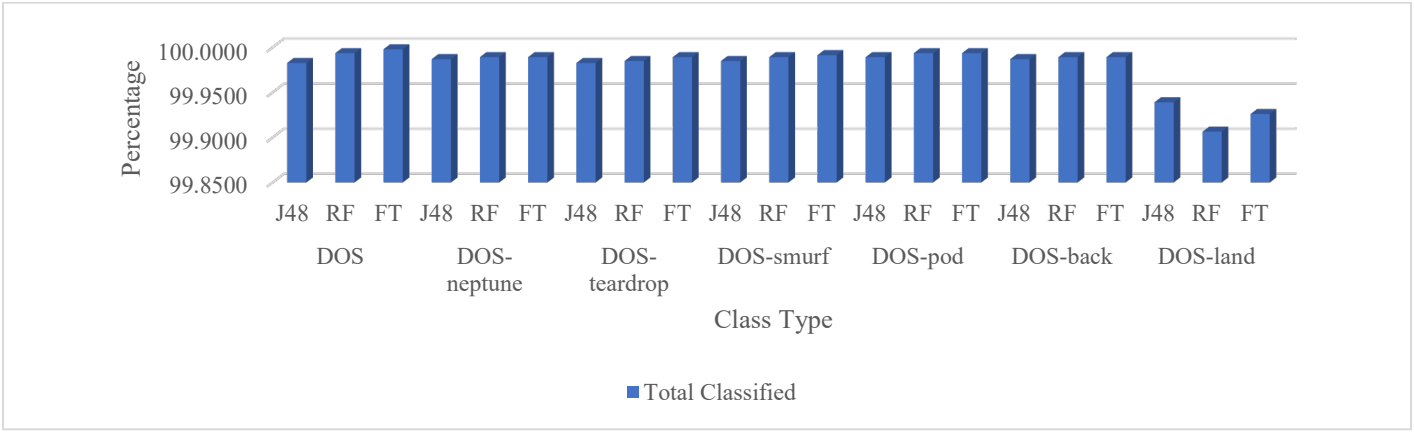

Figure 7: The Total Classification of Various Attack Class Ty pes of DOS Attack by Removing Redundant Features

Table 9: Classification of Various DOS Attacks Using Feature Reduction of Rank Based Feature Selection Classifiers

Sl. No	Class Type	Removed Features	Algorithm Used	Total No of Instances	Total Classified	Total Unclassified
1	NSL KDD'99 (DOS)	f11,f20, f22, f18, f19, f17 , f14, f9, f15, f16, f21	J48	45927	99.9826	0.0174
			RF	45927	99.9935	0.0065
			FT	45927	99.9978	0.0022
2	NSL KDD'99 (DOS-neptune)	f8, f10, f13, f7, f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	J48	41214	99.9869	0.0131
			RF	41214	99.9891	0.0109
			FT	41214	99.9891	0.0109
3	NSL KDD'99 (DOS-teardrop)	f39, f26, f41, f12, f10,	J48	892	99.9826	0.0174
			RF	892	99.9848	0.0152

		f13, f28, f31, f37, f7, f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	FT	892	99.9891	0.0109
4	NSL KDD'99 (DOS-smurf)	f30, f39, f25, f26, f8, f41, f12, f6, f10, f13, f28, f27, f31, f37, f7, f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	J48	2646	99.9848	0.0152
			RF	2646	99.9891	0.0109
			FT	2646	99.9913	0.0087
5	NSL KDD'99 (DOS-pod)	f30, f39, f25, f26, f41, f12, f6, f10, f13, f28, f27, f7, f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	J48	201	99.9891	0.0109
			RF	201	99.9935	0.0065
			FT	201	99.9935	0.0065
6	NSL KDD'99 (DOS-back)	f35, f8, f37, f7, f11, f20, f22, f18, f19,f17, f14, f9, f15, f16, f21	J48	956	99.9869	0.0131
			RF	956	99.9891	0.0109
			FT	956	99.9891	0.0109
7	NSL KDD'99 (DOS-land)	f5, f8, f41, f12, f6, f10, f13, f28, f11, f20, f22, f18, f19, f17, f14, f9, f15, f16, f21	J48	18	99.9390	0.0610
			RF	18	99.9064	0.0936
			FT	18	99.9260	0.0740

Table 10: Classification of Various DOS Attacks Based on Key Feature Selection of Rank Based Feature Selection Classifiers

Sl. No	Class Type	Selected Features	Key Feature	Algorithm Used	Total No of Instances	Total Classified	Total Unclassified
1	NSL KDD'99 (DOS)	f5, f4, f2, f3, f29, f36, f30, f24, f35, f34, f23, f33, f38, f39, f25, f26, f40, f8, f41, f12, f6, f10, f13, f32, f28, f27, f31, f37, f7, f1	_	J48	45927	99.9826	0.0174
				RF	45927	99.9935	0.0065
				FT	45927	99.9978	0.0022
2	NSL KDD'99 (DOS-neptune)	f5, f4, f2, f3, f29, f36, f30, f24, f35, f34, f23, f33, f38, f39, f25, f26, f40, f8, f41, f12, f6, f10, f13, f32, f28, f27, f31, f37, f7, f1	_	J48	41214	99.9869	0.0131
				RF	41214	99.9891	0.0109
				FT	41214	99.9891	0.0109
3	NSL KDD'99 (DOS-teardrop)	f5, f4, f2, f3, f1	f5	J48	892	99.9586	0.0414
				RF	892	99.9586	0.0414
				FT	892	99.9586	0.0414
4	NSL KDD'99 (DOS-smurf)	f4, f2, f3, f1, f29	f29	J48	2646	99.5275	0.4725
				RF	2646	99.5275	0.4725
				FT	2646	99.5275	0.4725
5	NSL KDD'99 (DOS-pod)	f4, f2, f3, f1, f29	f29	J48	201	99.5275	0.4725
				RF	201	99.5275	0.4725
				FT	201	99.5275	0.4725
6	NSL KDD'99 (DOS-back)	f4, f2, f3, f1, f12	f12	J48	956	99.5275	0.4725
				RF	956	99.5275	0.4725
				FT	956	99.5275	0.4725
7	NSL KDD'99 (DOS-land)	f4, f2, f3, f1, f7	f7	J48	18	99.5275	0.4725
				RF	18	99.5645	0.4355
				FT	18	99.5667	0.4333

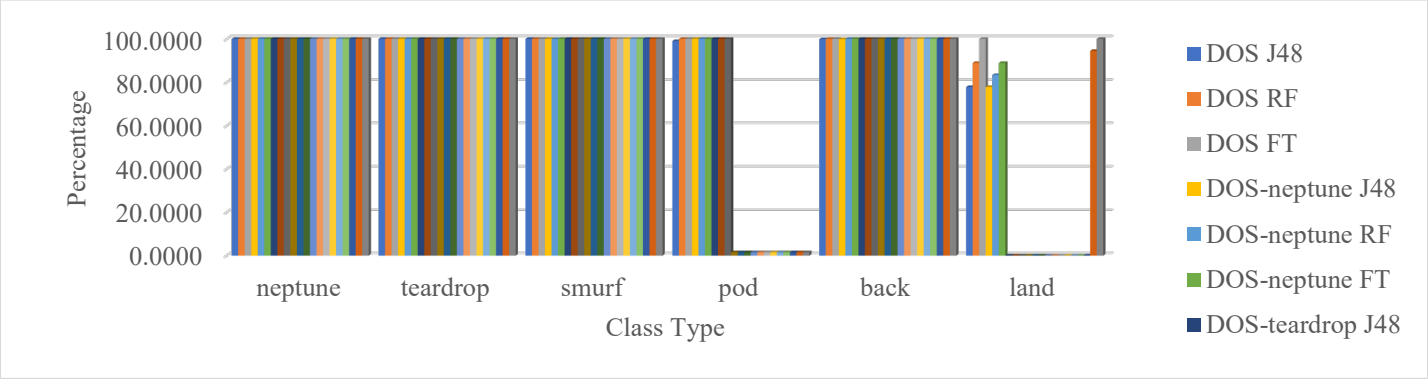

Figure 8: The Classification of Various Attack Class Types of DOS Attack by Selecting Key Features

The graph reflects the percentage of classification of various DOS attacks. The appropriate key features for DOS attack are applied and the redundant null valued features are eliminated then the most suited tree based algorithms are applied to find the optimum classified result.

The graph reflects the percentage of total classification of various DOS attacks. The appropriate key features for DOS attack are applied and the redundant null valued features are eliminated then the most suited tree based algorithms are applied to find the optimum classified result.

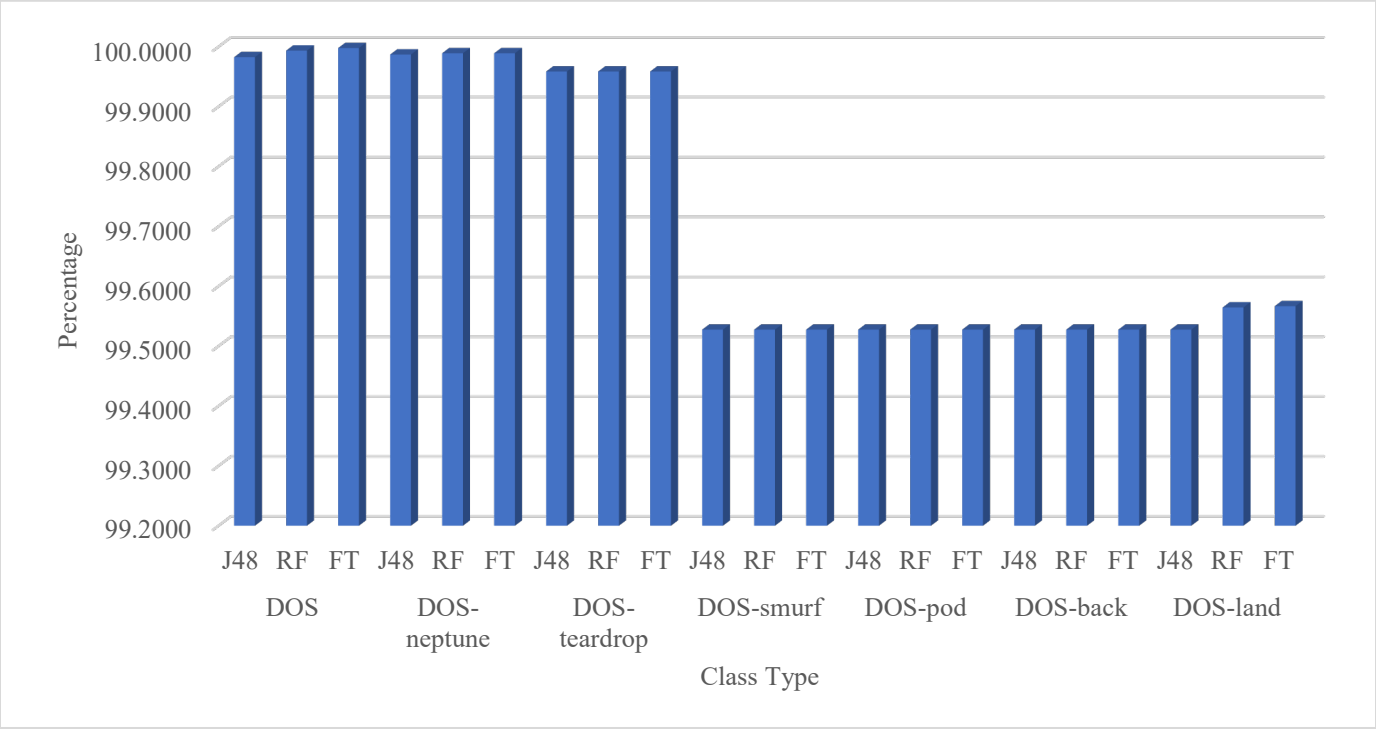

Figure 9: The Total Classification of Various Attack Class Types of DOS Attack by Selecting Key Features

Table 11: Classification of Various PROBES Attacks Using Feature Reduction of Rank Based Feature Selection Classifiers

Sl. No	Class Type	Removed Features	Algorithm Used	Total No of Instances	Total Classified	Total Unclassified
1	NSL KDD'99 (PROBES)	f7, f20, f9, f22, f8, f19, f18, f11, f17, f16, f15, f14, f13, f21	J48	11656	99.5281	0.4719
			RF	11656	99.6826	0.3174
			FT	11656	99.4852	0.5148
2	NSL KDD'99 (PROBES_ nmap)	f27, f40, f41, f28, f12,f10, f7, f20, f9, f22, f8, f19,f18, f11, f17, f16, f15, f14, f13, f21	J48	1493	99.5281	0.4719
			RF	1493	99.6482	0.3518
			FT	1493	99.5367	0.4633
3	NSL KDD'99 (PROBES_ ipsweep)	f26, f10, f7, f20, f9, f22, f8, f19, f18, f11, f15, f14, , f21	J48	3599	99.5281	0.4719
			RF	3599	99.6654	0.3346
			FT	3599	99.4595	0.5405
4	NSL KDD'99 (PROBES_ portsweep)	f7, f20, f9, f22, f8, f19, f18, f11, f17, f16, f15, f14, f13, f21	J48	2931	99.5281	0.4719
			RF	2931	99.6740	0.3260
			FT	2931	99.4852	0.5148
5	NSL KDD'99 (PROBES_ satan)	f7, f20, f9, f8, f19, f18, f15, f14, f21	J48	3633	99.5281	0.4719
			RF	3633	99.6740	0.3260
			FT	3633	99.4852	0.5148

The graph reflects the percentage of classification of different types of PROBES attacks. The redundant null valued features are eliminated and the most suited tree based algorithms are applied to find the optimum classified result.

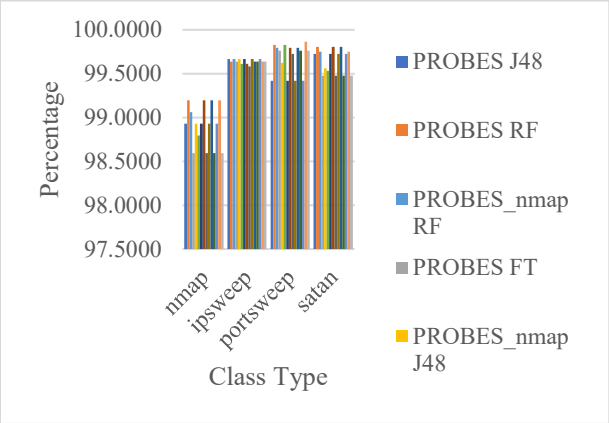

Figure 10: The Classification of Various Attack Class Types of PROBES Attack by Removing Redundant Features

The graph reflects the percentage of total classification of various PROBES attacks. The redundant null valued features are eliminated and the most suited tree based algorithms are applied to find the optimum classified result.

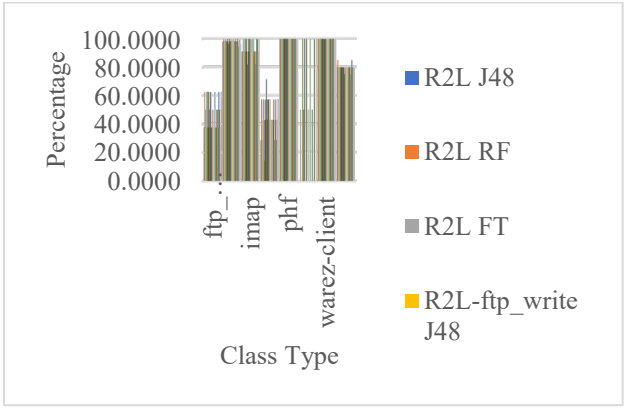

Figure 11: The Total Classification of Various Attack Class Types of PROBES Attack by Removing Redundant Features

Table 12: Classification of Various R2L Attacks Using Feature Reduction of Rank Based Feature Selection Classifiers

Sl. No	Class Type	Removed Features	Algorithm Used	Total No of Instances	Total Classified	Total Unclassified
1	NSL KDD'99 (R2L)	f29, f2, f20, f30, f15, f7, f8, f21	J48	995	98.0905	1.9095
			RF	995	98.7940	1.2060
			FT	995	98.4925	1.5075
2	NSL KDD'99 (R2L-ftp_ write)	f39, f28, f11, f38, f40, f41, f27, f26, f25, f14, , f18, f20, f30, f15, f7, f8, f21	J48	8	98.2915	1.7085
			RF	8	98.7940	1.2060
			FT	8	98.5930	1.4070
3	NSL KDD'99 (R2L_guess_ passwd)	f35, f19, f31, f17, f14, f16, f13, f18, f9, f20,f30, f15, f7, f8, f21	J48	53	98.1910	1.8090
			RF	53	98.8945	1.1055
			FT	53	98.4925	1.5075
4	NSL KDD'99 (R2L_ imap)	f11, f41, f27, f37, f19, f17, f14, f22, f18, f9, f20, f30, f15, f7, f8, f21	J48	11	98.0905	1.9095
			RF	11	98.7940	1.2060
			FT	11	98.5930	1.4070
5	NSL KDD'99 (R2L_ multihop)	f39, f28, f11, f38, f41, f27, f37, f26, f25, f9, f20, f30, f15, f7, f8, f21	J48	7	98.1910	1.8090
			RF	7	98.6935	1.3065
			FT	7	97.9899	2.0101
6	NSL KDD'99 (R2L_phf)	f39, f11, f38, f40, f41, f36, f27, f37, f26, f25, f17, f22, f16, f13, f18, f9, f20, f30, f15, f7, f8, f21	J48	4	98.1910	1.8090
			RF	4	98.6935	1.3065
			FT	4	98.1910	1.8090
7	NSL KDD'99 (R2L_spy)	f28, f10, f11, f40, f41, f36, f27, f37, f31, f26, f25, f14, f22, f16, f13, , f9, f20, f30, f7, f8, f21	J48	2	98.2915	1.7085
			RF	2	98.5930	1.4070
			FT	2	97.9899	2.0101
8	NSL KDD'99 (R2L_ warezclient)	f11, f19, f17, f14, f16, f13, f18, f9, f20, , f15, f7, f8, f21	J48	890	98.0905	1.9095
			RF	890	98.8945	1.1055
			FT	890	98.3920	1.6080
9	NSL KDD'99 (R2L_w arezmaster)	f39, f28, f11, f41, f27, f37, f19, f31, f26, f25, , f14, f16, f13, f18, f9, f20, f30, f15, f7, f8, f21	J48	20	98.1910	1.8090
			RF	20	98.6935	1.3065
			FT	20	98.5930	1.4070

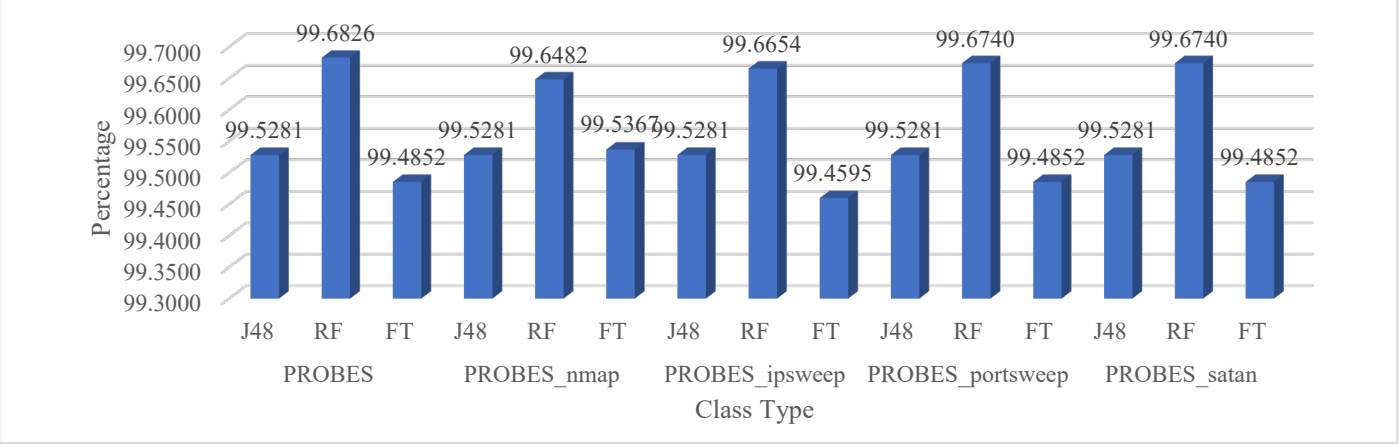

Figure 12: The Classification of Various Attack Class Types of R2L Attack by Removing Redundant Features

The graph reflects the percentage of classification of different types of R2L attacks. The redundant null valued features are eliminated and the most suited tree based algorithms are applied to find the optimum classified result.

The graph reflects the percentage of total classification of various PROBES attacks. The redundant null valued features are eliminated and the most suited tree based algorithms are applied to find the optimum classified result.

Table 13: Classification of Various R2L Attacks Based on Key Feature Selection of Rank Based Feature Selection Classifiers

Sl. No	Class Type	Selected Features	Key Feature	Algorithm Used	Total No of Instances	Total Classified	Total Unclassified
1	NSL KDD'99 (R2L)	f6, f5, f3, f12, f39, f28, f4, f10, f11, f38, f40, f41, f36, f27, f1, f32, f33, f35, f37, f34, f19, f31, f26, f25, f24, f17, f23, f14, f22, f16, f13, f18, f9		J48	995	98.0905	1.9095
				RF	995	98.7940	1.2060
				FT	995	98.4925	1.5075
2	NSL KDD'99 (R2L-ftp_write)	f3, f4, f1, f29, f2	f29	J48	8	98.0905	1.9095
				RF	8	98.4925	1.5075
				FT	8	96.8844	3.1156
3	NSL KDD'99 (R2L_guess_ passwd)	f3, f4, f1, f29, f2	f29	J48	53	98.0905	1.9095
				RF	53	98.4925	1.5075
				FT	53	96.8844	3.1156
4	NSL KDD'99 (R2L_imap)	f3, f4, f1, f29, f2	f29	J48	11	98.0905	1.9095
				RF	11	98.4925	1.5075
				FT	11	96.8844	3.1156
5	NSL KDD'99 (R2L_ multihop)	f3, f4, f1, f29, f2	f29	J48	7	98.0905	1.9095
				RF	7	98.4925	1.5075
				FT	7	96.8844	3.1156
6	NSL KDD'99 (R2L_phf)	f3, f4, f1, f29, f2	f29	J48	4	98.0905	1.9095
				RF	4	98.4925	1.5075
				FT	4	96.8844	3.1156
7	NSL KDD'99 (R2L_spy)	f3, f4, f1, f29, f2	f29	J48	2	98.0905	1.9095
				RF	2	98.4925	1.5075
				FT	2	96.8844	3.1156
8	NSL KDD'99 (R2L_ warezclient)	f3, f12, f4, f1, f2	f12	J48	890	98.0905	1.9095
				RF	890	98.2915	1.7085
				FT	890	98.0905	1.9095
9	NSL KDD'99 (R2L_ warezmaster)	f3, f4, f1, f29, f2	f29	J48	20	98.0905	1.9095
				RF	20	98.4925	1.5075
				FT	20	96.8844	3.1156

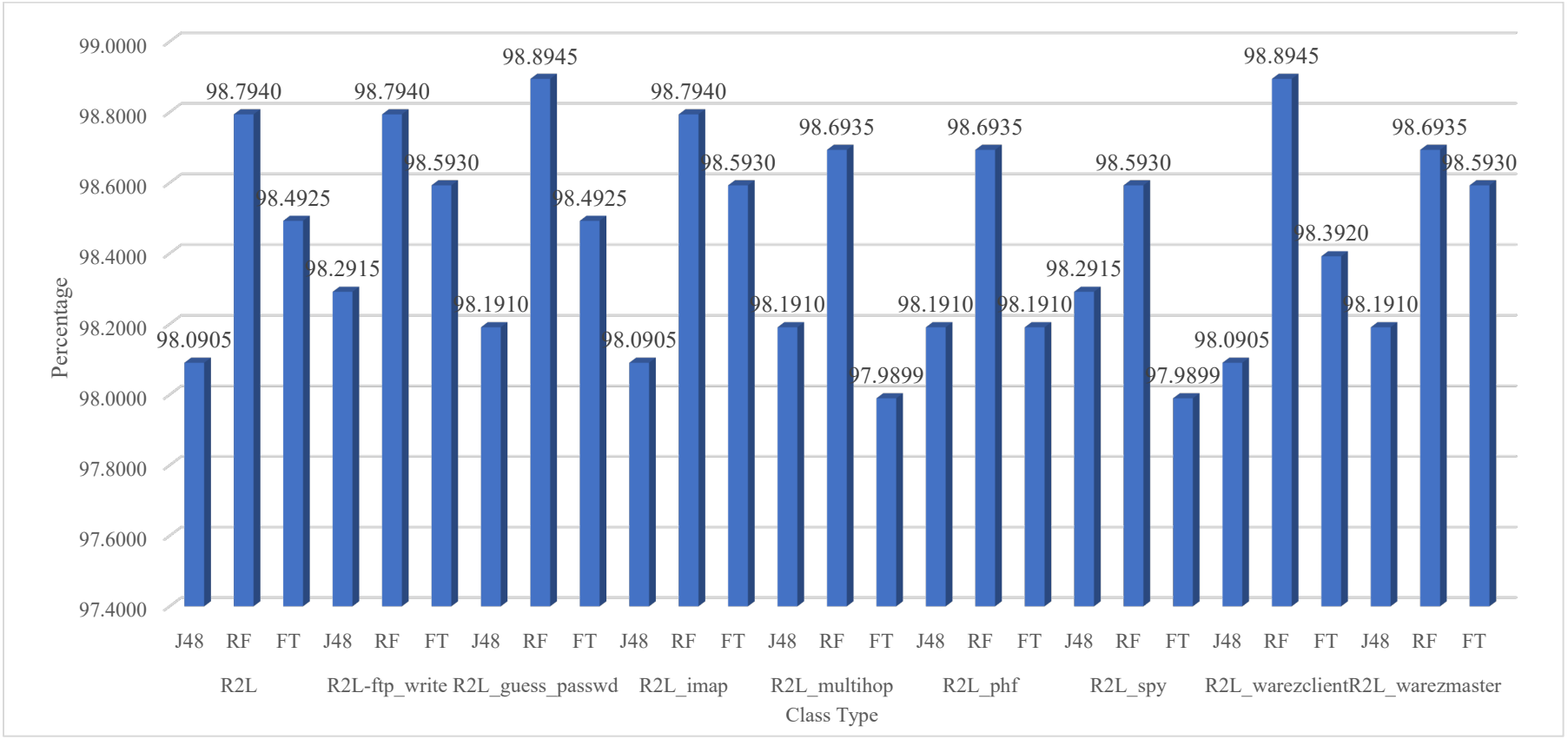

Figure 13: The Total Classification of Various Attack Class Types of R2L Attack by Removing Redundant Features

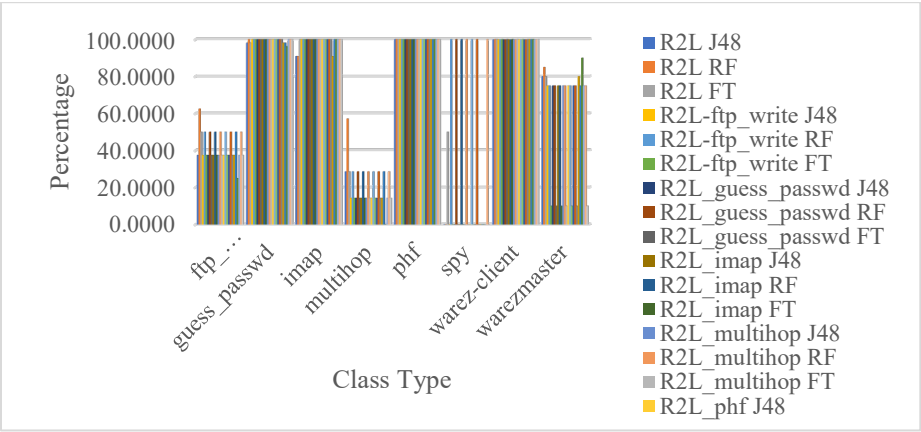

Figure 14: The Classification of Various Attack Class Types of R2L Attack by Selecting Key Features

The graph reflects the percentage of classification of various R2L attacks. The appropriate key features for R2L attack are applied and the redundant null valued features are eliminated then the most suited tree based algorithms are applied to find the optimum classified result.

The graph reflects the percentage of total classification of various R2L attacks. The appropriate key features for R2L attack are applied and the redundant null valued features are eliminated then the most suited tree based algorithms are applied to find the optimum classified result.

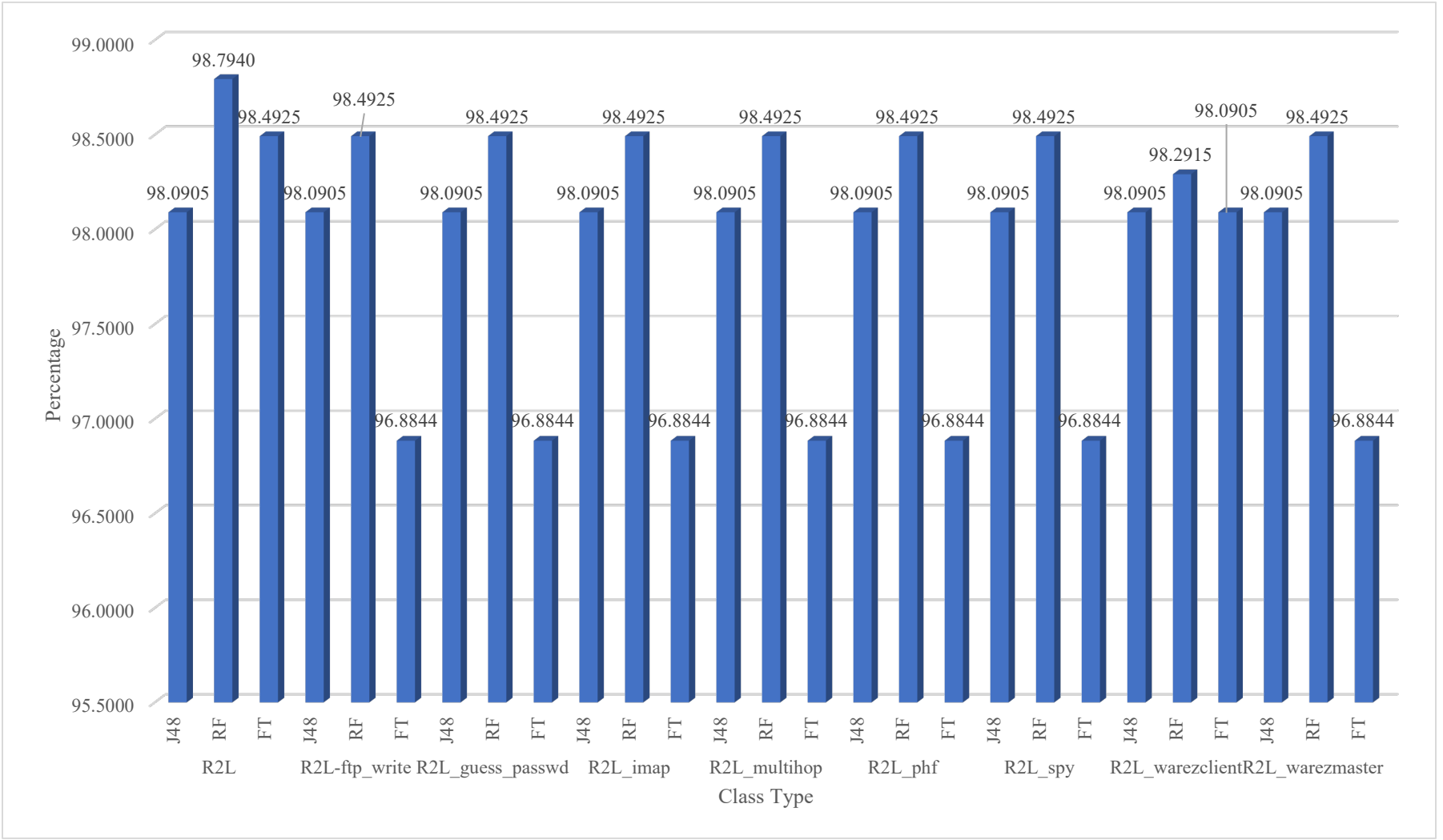

Figure 15 : The Total Classification of Various Attack Class Types of R2L Attack by Selecting Key Features

Table 14: Classification of Various U2R Attacks Using Feature Reduction of Rank Based Feature Selection Classifiers

Sl. No	Class Type	Removed Features	Algorithm Used	Total No of Instances	Total Classified	Total Unclassified
1	NSL KDD'99 (U2R)	f15, f14, f13, f41, f11, f9, f8, f7, f21, f19, f33, f30, f31, f37, f20, f38, f39, f29, f28, f27, f26, f40, f22, f23, f24, f25, f1	J48	52	80.7692	19.2308
			RF	52	84.6154	15.3846
			FT	52	84.6154	15.3846
2	NSL KDD'99 (U2R-buffer_ overflow)	f35, f18, f15, f11, f9, f8, f7, f21, f19, f31, f20, f38, f39, f26, f22,	J48	30	76.9231	23.0769
			RF	30	80.7692	19.2308
			FT	30	78.8462	21.1538
3	NSL KDD'99 (U2R_loadmodule)	f15, f11, f9, f8, f7, f21, f31, f20, f38, f39, f28, f27, f26, f40, f22, f25	J48	9	82.6923	17.3077
			RF	9	84.6154	15.3846
			FT	9	88.4615	11.5385
4	NSL KDD'99 (U2R_ perl)	f10, f15, f13, f41, f11, f9, f8, f7, f21, f19, f30, f31, f37, f20, f38,f39, f28, f27, f26, f22, f25	J48	3	78.8462	21.1538
			RF	3	78.8462	21.1538
			FT	3	86.5385	13.4615
5	NSL KDD'99 (U2R_ rootkit)	f18, f15, f8, f7, f21, f19, f30, f31, f20, f38, f28, f27, f26, f22, f25	J48	10	80.7692	19.2308
			RF	10	80.7692	19.2308
			FT	10	90.3846	9.6154

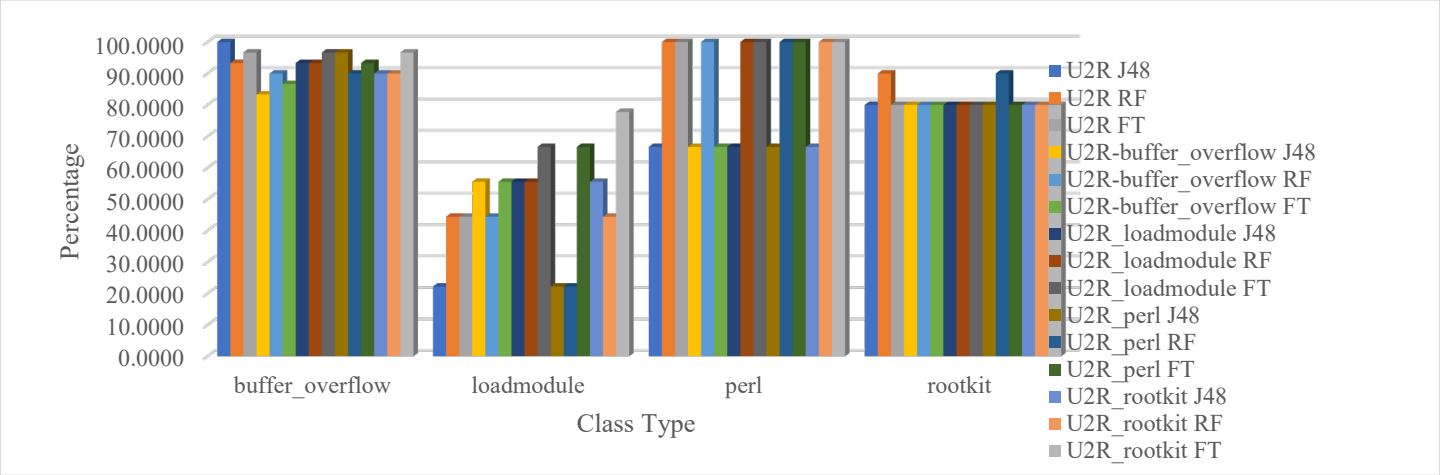

Figure 16: The Classification of Various Attack Class Types of U2R Attack by Removing Redundant Features

The graph reflects the percentage of classification of different types of U2R attacks. The redundant null valued features are eliminated and the most suited tree-based algorithms are applied to find the optimum classified result.

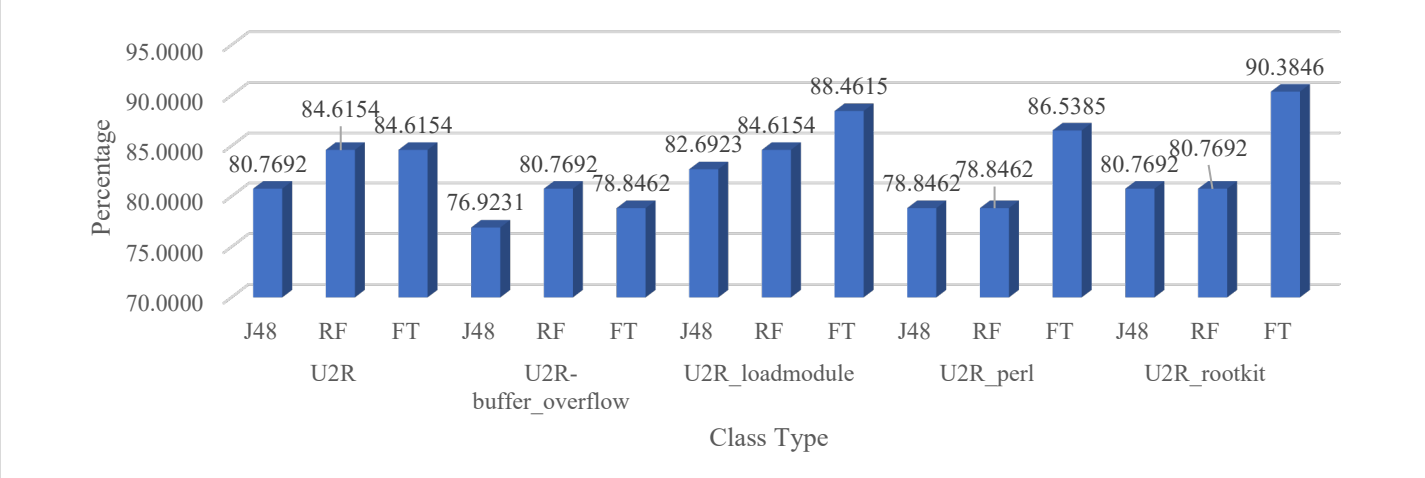

Figure 17: The Total Classification of Various Attack Class Types of U2R Attack by Removing Redundant Features

The graph reflects the percentage of total classification of various U2R attacks. The redundant null valued features are eliminated and the most suited tree based algorithms are applied to find the optimum classified result.

The graph reflects the percentage of classification of various U2R attacks. The appropriate key features for U2R attack are applied and the redundant null valued features are eliminated then the most suited tree based algorithms are applied to find the optimum classified result

Table 15: Classification of Various U2R Attacks Based on Key Feature Selection of Rank Based Feature Selection Classifiers

Sl. No	Class Type	Selected Features	Key Feature	Algorithm Used	Total No of Instances	Total Classified	Total Unclassified
1	NSL KDD'99 (U2R)	f12, f3, f2, f4, f29, f1	f12, f29	J48	52	63.4615	36.5385
				RF	52	65.3846	34.6154
				FT	52	59.6154	40.3846
2	NSL KDD'99 (U2R-buffer_overflow)	f12, f3, f2, f4, f1	f12	J48	30	67.3077	32.6923
				RF	30	57.6923	42.3077
				FT	30	61.5385	38.4615
3	NSL KDD'99 (U2R_loadmodule)	f12, f3, f2, f4, f1	f12	J48	9	67.3077	32.6923
				RF	9	57.6923	42.3077
				FT	9	61.5385	38.4615
4	NSL KDD'99 (U2R_perl)	f12, f3, f2, f4, f1	f12	J48	3	67.3077	32.6923
				RF	3	57.6923	42.3077
				FT	3	61.5385	38.4615
5	NSL KDD'99 (U2R_rootkit)	f3, f2, f4, f29, f1	f29	J48	10	53.8462	46.1538
				RF	10	65.3846	34.6154
				FT	10	57.6923	42.3077

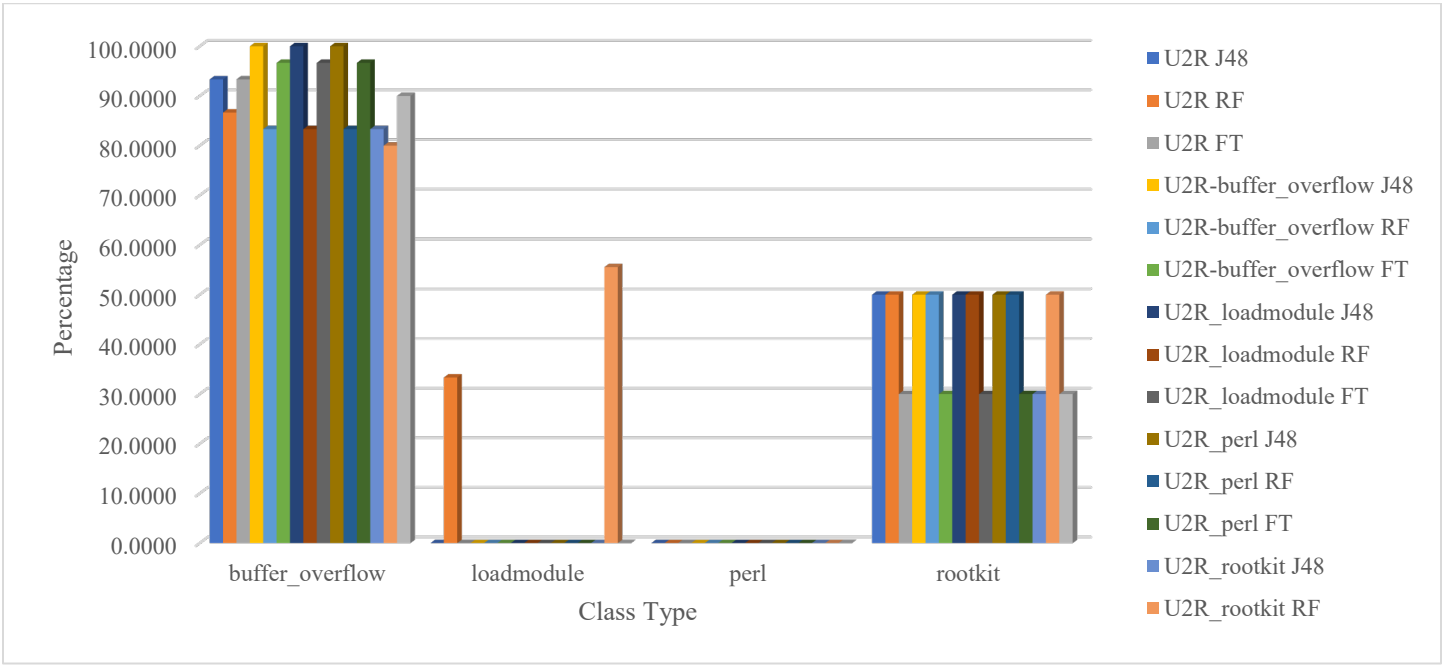

Figure 18: The Classification of Various Attack Class Types of U2R Attack by Selecting Key Features

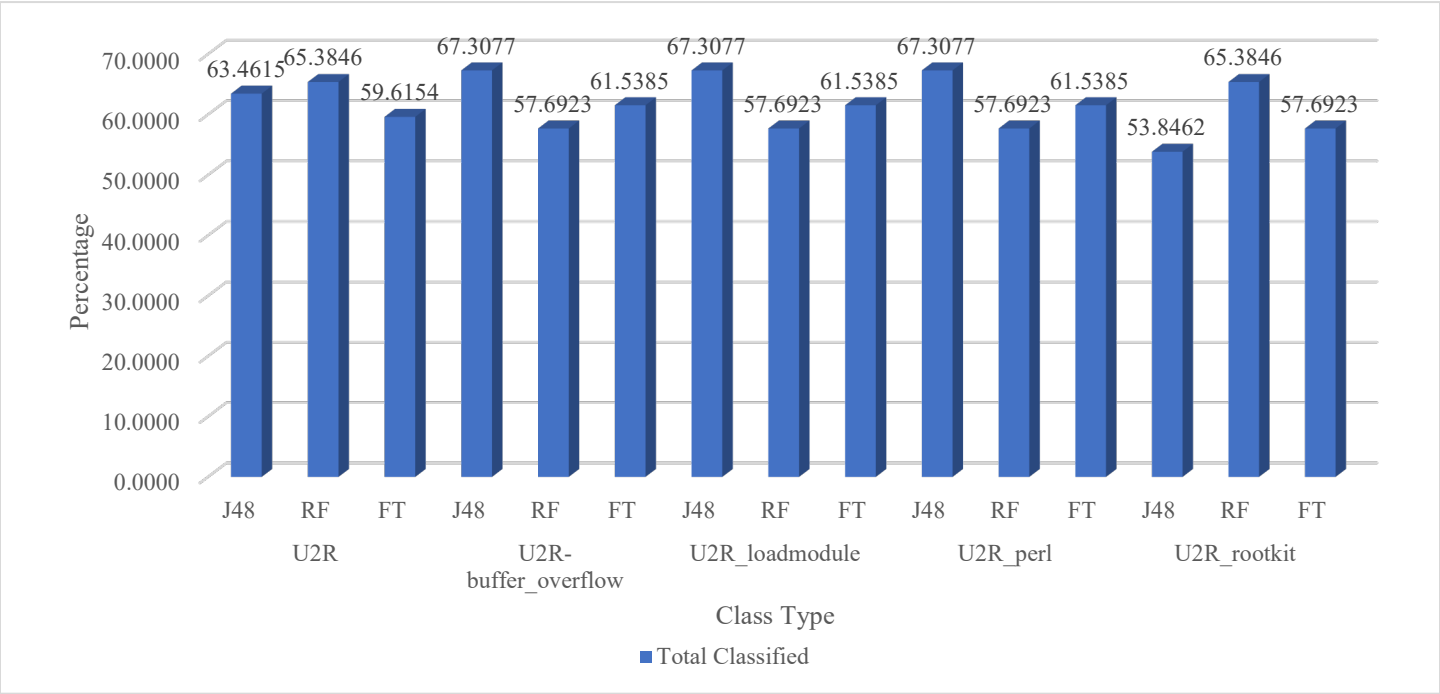

Figure 19: The Total Classification of Various Attack Class Types of U2R Attack by Selecting Key Features

The graph reflects the percentage of total classification of various U2R attacks. The appropriate key features for U2R attack are applied and the redundant null valued features are eliminated then the most suited tree based algorithms are applied to find the optimum classified result.

6. Conclusion

The classification of various attack types is based on the percentage of total classified. The null rank valued features are eliminated and the selected features are applied to find the optimum classification result. The various attack classes are selected and the attack types are classified on the basis of classification result. Various tree-based classification algorithms are used and the results are optimized based on their classification percentage. The optimized result is compared with the result computed by selected key feature. The computed result is merely dependent and influential by the selected key feature. The null valued features are eliminated and the contributing features are selected. Finally, the key feature is selected from the list of contributing features and the result is optimized.

Conflict of Interest

The authors declare no conflict of interest.

References

[1] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, M. Rajarajan, "A survey of intrusion detection techniques in cloud," Network Computer Application, **36**(1), 42–57, 2013. Doi: https://doi.org/10.1016/j.jnca.2012.05.003

[2] Z. Muda, W. Yassin, M. N. Sulaiman, N. I. Udzir, "Intrusion detection based on K-means clustering and OneR classification," International Conference Information Assurance Security IAS 2011, 192–197, 2011. DOI: 10.1109/ISIAS.2011.6122818

[3] W. Koff, P. Gustafson, "CSC leading edge forum data revolution," CSC leading edge forum, 68, 2011.

[4] S. V. Thakare, D. V. Gore, "Comparative study of CIA," International Conference Communication System Network Technology, 713–718, 2014.

[5] U. Fiore, F. Palmieri, A. Castiglione, A. De Santis, "Network anomaly detection with the restricted Boltzmann machine," Neurocomputing, 122, 13–23, 2013.

[6] Z. Muda, W. Yassin, M. N. Sulaiman, N. I. Udzir, "Intrusion detection based on K-means clustering and Naïve Bayes classification," 7th Int. Conf. IT Asia Intrusion, 1–6, 2011.

[7] G. Folino, P. Sabatino, "Ensemble based collaborative and distributed intrusion detection systems: a survey," Netw. Comput. Appl., 66, 1–16, 2016. Doi: https://doi.org/10.1016/j.jnca.2016.03.011

[8] S. Fakhraei, H. Soltanian-Zadeh, F. Fotouhi, "Bias and stability of single variable classifiers for feature ranking and selection," Expert Syst. Appl., **41**(15), 6945–6958, 2014. Doi: https://doi.org/10.1016/j.eswa.2014.05.007

[9] Y. Freund, R. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," Comput. Learn. theory, 55, 119–139, 1995.

[10] S. Alelyani, J. Tang, H. Liu, "Feature selection for clustering : a review," Data Clust. Algorithms Appl., 1–37, 2013.

[11] F. Amiri, M. Rezaei Yousefi, C. Lucas, A. Shakery, N. Yazdani, "Mutual information-based feature selection for intrusion detection systems," Netw. Comput. Appl., **34**(4), 1184–1199, 2011. Doi: https://doi.org/10.1016/j.jnca.2011.01.002

[12] S. Solorio-Fernandez, J. A. Carrasco-Ochoa, J. F. Martinez-Trinidad, "A new hybrid filter wrapper feature selection method for clustering based on ranking," Neurocomputing, 214, 866–880, 2016.

[13] M. A. Hall, "Correlation-based feature subset selection for machine learning," Hamilton, New Zeland, 1999.

[14] N. Cleetus, "Genetic algorithm with different feature selection method for intrusion detection," First Int. Conf. Comput. Syst. Commun., 220–225, 2014.

[15] D. E. Denning, "Intrusion-detection model," IEEE Trans. Softw. Eng., 2, 222–232, 1987.

[16] M. V. Mahoney, P. K. Chan, "PHAD: packet header anomaly detection for identifying hostile network traffic," Florida Technol. Tech. Rep. CS-2001, 2001.

[17] S. B. Shamsuddin, M. E. Woodward, "Modeling protocol based packet header anomaly detector for network and host intrusion detection systems," Proc. 6th Int. Conf. Cryptol. Netw. Secur., 209–227, 2007.

[18] Md. Nasim Adnan, Md. Zahidul Islam, "Forest PA: constructing a decision forest by penalized attributes used in previous trees," Expert Systems With Applications, 89, 389-403 2017.

[19] J. Gama, "Functional trees," Machine Learning, **55**(3), 219-250.

[20] G. Hulten, Laurie Spencer, Pedro Domingos, “Mining time-changing data streams,” in ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 97-106, 2001.
[21] R. Quinlan, C4.5: programs for machine learning, Morgan Kaufmann Publishers, San Mateo, CA.
[22] J. M. Perez, Javier Muguerza, Olatz Arbelaitz, Ibai Gurrutxaga, Jose I. Martin, “Combining multiple class distribution modifies subsamples in a single tree,” Pattern recognition Letters, **28**(4), 414-422, 2007. Doi: https://doi.org/10.1016/j.patrec.2006.08.013
[23] G. Webb, “Decision tree grafting from the all-tests-but-one partition,” in San Francisco, CA, 1999.
[24] G. Holmes, Bernhard Pfahringer, Richard Kirkby, Eibe Frank, Mark Hall, “Multiclass alternating decision trees,” in ECML, 161-172, 2001.
[25] N. Landwehr, Mark Hall, Eibe Frank, “Logistic model trees,” Machine Learning, **95**(1-2), 161-205, 2005.
[26] R. Kohavi, “Scaling up the accuracy of Naïve-Bayes classifiers,” A Decision-Tree Hybrid, in Second International Conference on Knowledge Discovery and Data Mining, 202-207, 1996.
[27] L. Breman, “Random forests,” Machine Learning, **45**(1), 5-32.
[28] I. Giggins, “Knowledge discovery through sysfor – systematically developed forest of multiple decision trees,” in Australasian Data Mining Conference (AusDM 11), Ballarat, Australia, 195-204, 2011.
[29] M. Nanda, M. Patra, "Network intrusion detection and monitoring in cloud based systems," in 2019 International Conference on Applied Machine Learning (ICAML), Bhubaneswar, India, 197-204, 2019. doi: 10.1109/ICAML48257.2019.00045,
[30] M. K. Nanda, M. R. Patra, Chapter 17, Intrusion detection and classification using decision tree-based feature selection classifiers, Springer Science and Business Media LLC, 2021.
[31] H.S. Hota, Dinesh K. Sharma, A.K. Shrivas, "Development of an efficient classifier using proposed sensitivity-based feature selection technique for intrusion detection system", International Journal of Information and Computer Security, 2018.
[32] M.H. Kamarudin, Carsten Maple, Tim Watson, Nader Sohrabi Safa, "A new unified intrusion anomaly detection in identifying unseen web attacks", Security and Communication Networks, 2017.