INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2021, VOL. 67, NO. 2, PP. 267-275
Manuscript received June 12, 2020; revised April, 2021 DOI: 10.24425/ijet.2021.135975

Optimal Ensemble Learning Based on Distinctive Feature Selection by Univariate ANOVA-F Statistics for IDS

Shaikh Shakeela, N Sai Shankar, P Mohan Reddy, T Kavya Tulasi, and M Mahesh Koneru

***Abstract*—Cyber-attacks are increasing day by day. The generation of data by the population of the world is immensely escalated. The advancements in technology, are intern leading to more chances of vulnerabilities to individual's personal data. Across the world it became a very big challenge to bring down the threats to data security. These threats are not only targeting the user data and also destroying the whole network infrastructure in the local or global level, the attacks could be hardware or software. Central objective of this paper is to design an intrusion detection system using ensemble learning specifically Decision Trees with distinctive feature selection univariate ANOVA-F test. Decision Trees has been the most popular among ensemble learning methods and it also outperforms among the other classification algorithm in various aspects. With the essence of different feature selection techniques, the performance found to be increased more, and the detection outcome will be less prone to false classification. Analysis of Variance (ANOVA) with F-statistics computations could be a reasonable criterion to choose distinctives features in the given network traffic data. The mentioned technique is applied and tested on NSL KDD network dataset. Various performance measures like accuracy, precision, F-score and Cross Validation curve have drawn to justify the ability of the method.**

***Keywords*—ANOVA-F test, Cross Validation, Decision Trees, Features, NSL-KDD, Dataset**

I. Introduction

INTRUSION detection field has gained more pace in recent times because of the huge increase in cyber-attacks. The researchers are coming with different solutions to cop up with various types of vulnerable activities in the network. Data communication over the world-wide network traffic involves plethora of technicalities like device configurations, network protocols, security frameworks, network interoperability, hybrid or heterogeneous networks and many more. Majorly world population using 3G, LTE and WiFi for internet. As the networks massively used and also the presence of loopholes in the network security, enables intruders to break into the any secured network. Mobility and flexible access in WiFi which used to consider as the quality of the network but that itself became questionable security issues. The wireless LAN standards 802.11 now a days common among every organization for internet facility and since the origin till now this standard has seen lot of phases with respect to security issues. In the beginning the standard incorporated Wired Equivalent Protection (WEP) protocol to secure a connection but due to weak encryption and strategies it has jeopardized the user data and the network. Confidentiality and protection should be the prime importance of the network which can be acquired with the enhancements and the robustness present in later versions of WLAN security protocols and it has drastically reduced the threats. But still malicious activities of attackers are pervading the whole network infrastructure. To find the solution for such problems the primary thing is to know the whole architecture of the computer data networks and various other aspects [1]. Potential threats can be of different types, basically there are two types of attacks one is passive, and another is active. In passive attacks system resources will not be affected, instead intruder tries to monitor the network for data traffic. And the network traffic information may be used by the attacker, against the legitimate system. Some of the examples of this types of attacks are Scanning, Encryption and Tapping. But in active attacks intruder potentially exploit the system and also its resources. Session replays, DOS (Denial of Service) and masquerade are some of the examples. Man-in-the-middle, Spoofing, Eavesdropping, Spam are commonly seen with computer networks. There are numerous ways of attacks, even some are unnamed, untraceable and people with ill intensions are trying new methods to harm more. In order to tackle with these huge variety of attacks requires massive research and foresightedness.

To develop a highly effective intrusion detection system, one need to analyze and study the behavior of various attacks and other relevant aspects. In existing research proposed by prominent people showed various approaches using different strategies. However, these approaches are specific for particular type of attack or malicious activities. People from around the world are working to develop robust and substantial mechanisms. As the data processed at various stages of computer system and networks so there are very high chances of finding one among those loose ends and exploit that. With the availability of datasets in abundance and high system processing power researchers are turning towards Machine Learning (ML) and Data Science areas. One such kind of research is related to malicious JavaScript code detection which is commonly used by attacker to trespass in user system [2]. This idea deals with the malign activity by using modern deep learning framework for the IDS design. Especially it is concentrating on using the concept of sparse random projection and logistic regression algorithm. And also, the stacked denoising auto encoder has been used to extract the feature in JavaScripts. Another most popular algorithm in ML is Recurrent Neural Network (RNN) that can be used to track such attacks.

The Authors are with of ECM, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India, (e-mail: shaikh.shakeela37@gmail.com, shankarchowdary402@gmail.com, mohanreddi98@gmail.com, kavyatulasi.6@gmail.com, maheshmanepalli8@gmail.com).

Researchers have tried this ML model on NSL-KDD dataset to detect intrusions [3]. RNN basically trace the changes in the sequence of samples provided as the datasets for the training and does the detection, RNNs are specially designed for time series data analysis and prediction applications. In organizations Network Intrusion Detection tools are very essential, in spite of network firewall and other preventive measure hackers breach into the organizations networks so Network Intrusion Detection (NIDS) Tools are highly recommended. NIDS categorized into two, first one is signature based and the second anomaly detection NIDS (ADNIDS). One of the techniques involves flexible and effective NIDS using Self Taught Learning (STL) which is also a deep learning technique [4]. The network traffic plays a major role to identify the type of intrusion in the network so the benchmark dataset which is most popular in intrusion detection system analysis is KDD (Knowledge discovery dataset) datasets. There has been wide research on this dataset, and it has various versions which includes the records of network traffic and data protocols related information [5]. Most of the algorithms which are commonly used in machine learning for intrusion detection systems has shown good results, apart from the basic artificial neural networks there are other methods which are also used for intrusion detection which are back propagation artificial neural networks, multiclass support vector machines and decision trees [6]. Most of the times this machine learning algorithms are supervised learning techniques which are suitable for intrusion detection. One of the researches has used unsupervised learning technique for detecting the attacks, this uses Boltzmann machines which is a deep learning model. This technique includes the use of multilayer restricted Boltzmann machine (RBM) and performs the training on each layer of this RBM in an unsupervised way [7]. It also adds back propagation neural network, followed by the main part of the model and the later part performs supervised learning on the data. Some of the researchers have used various optimization techniques for intrusion detection tasks the approach deals with the dataset with the help of multi objective Genetic algorithm techniques [8]. Genetic algorithm is the most popular optimization technique which has wide area of usage in various applications. When it is used with the benchmark dataset for the detection purpose, it has shown substantial outcome. However, researchers are showing more interest towards machine learning algorithms at present. The popularity and the performance of the ML models especially in the area of classification application is more. So now ML models are used with the aid of various other concepts. Earlier in the discussion some of the hybrid models have been presented. In the field of Machine learning other than the neural networks models there one more algorithm which is very in popular for classification problem is Naïve Bays (NB) classifiers. Research literature shows the traditional NB classifiers also has been a good tool for the intrusion detection. But the variations in NB classifier i.e. HNB binary Classifier has better performance in terms of accuracy [9]. One of the lucrative ways of classifying the intrusions is the use of soft computing techniques [10]. It basically detects the type of attack by finding the unusual behavior of computer network by incorporating several approaches like fuzzy inference approach, neuro fuzzy networks and subtracting clustering.

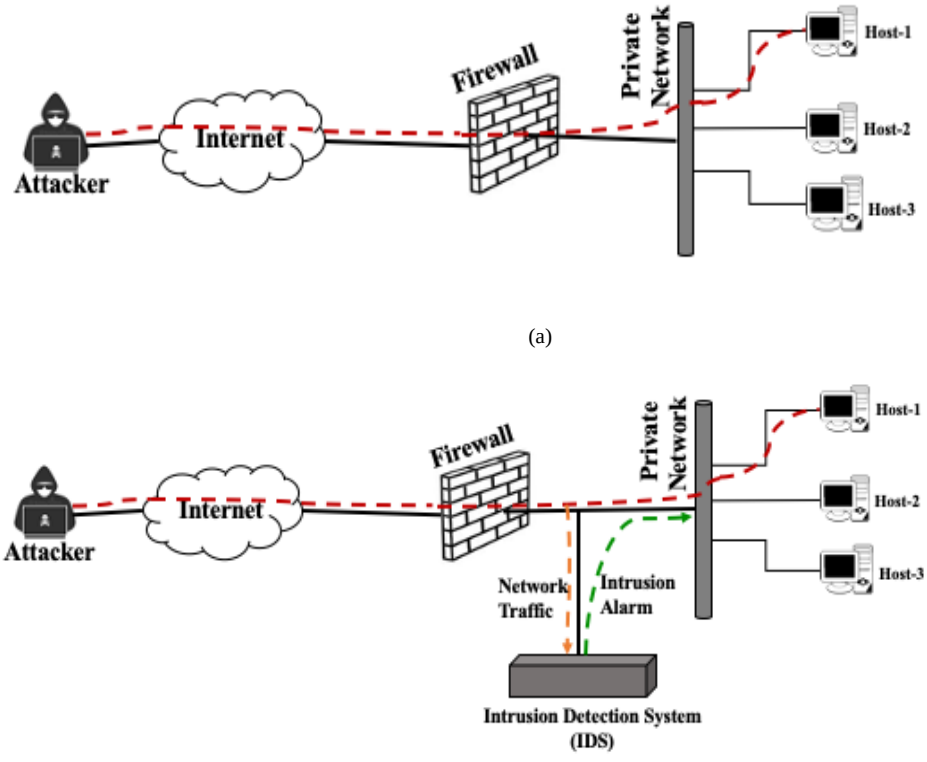

Fig. 1. (a) Network without IDS (b) Network with IDS

The proposed work based on the Decision Tree based classification of various attacks. Attacks in data and computer networks have vast information which will be taken as the features of the particular attack. These features are nothing but the behavior of the data networks and protocol. When there will be an unusual and suspicious environment occurs, these features will be the reason behind this. The attackers play with protocols and take the advantage to destroy the normal data transmission to or from the host system or network. Features are most important factor to decide whether the network in danger for data theft. In order to perform intrusion detection this features has to be applied to the Decision Trees ML model. Prior to feeding features to the model, there is a need of analyzing the feature. These features will intern reveals some other information when these undergoes for various statistical analysis. One such statistical analysis technique is ANOVA and F-Test. Preceding parts of the paper concentrate on the discussion on proposed method and benchmark dataset. The arrangement of representation of the paper is divided into VIII parts. The part I comprises of the introduction and the literature of Intrusion detection systems. Part II sheds light on how the basic intrusion detection system works and it explains what various building blocks of the system are. Part III and Part IV focuses on the detailing about the benchmark dataset NSL-KDD and ANOVA F-Test respectively. The most important part of the proposed work is the model used in the method which is Decision Trees. Part V gives the significance of Decision Trees in IDS. The methodology by combining all the concepts will be included in Part VI. And finally, Part VII concentrates on the result discussion and the impact of the proposed work on the Intrusion detection. And finally, the paper concludes in part VIII.

II. BASIC INTRUSION DETECTION SYSTEM

The intension behind making IDS is to tackle attacks in the network. The intrusion could be from inside or outside the network. Figure 1. shows the basic structure of IDS. Here two types of design can be seen. Figure 1. (a) shows that the network with no IDS, this makes the whole data communication at risk. Internet in the figure comprises of various networking resources and devices like servers, gateways and routers etc. In-spite the presence of firewall (typically used to protect the network) attackers are able to reach through the end user or any private network. Breaking the firewall now a days became a toy game for intruders. So, without the use of IDS the data networks and user nodes will not be safe. Figure 1. (b) shows the network with IDS. When the network employs the IDS, it acts like a sensor and it supervises the network traffic.

IDS try to find any abnormal behavior in the data traffic and sends indications to host. These indications could be of the protocols specifically designed to intimate the node about the breach of the security in its vicinity. Moreover, IDS is responsible to take care of the security issues which may occur in a network and sends intrusion alarms to the end system to safeguard it for present and future attacks. IDS structure involves the deployment of models or any framework. These are especially designed with reference to the data which gives the information about the attacks occurred over the time in history up to present. And some algorithms are capable of finding the anomaly in the network traffic behaviour so that the presence of any malicious activity can be intercepted.

TABLE I
KDD DATASET ATTRIBUTES

S.No.	Attribute Name	S.No.	Attribute Name	S.No.	Attribute Name	S.No.	Attribute Name
1	duration	13	num_compro	25	rerror_rate	37	dst_host_srv_diff_host_rate
2	protocol_type	14	root_shell	26	same_srv_rate	38	dst_host_serror_rate
3	Service	15	su_attempted	27	diff_srv_rate	39	dst_host_srv_serror_rate
4	src_bytes	16	num_root	28	srv_count	40	dst_host_rerror_rate
5	dst_bytes	17	num_file_creations	29	srv_serror_rate	41	dst_host_srv_rerror_rate
6	flag	18	num_shells	30	srv_rerror_rate	42	class
7	land wrong_fragment	19	num_access_files	31	srv_diff_h_rate		
8	urgent	20	num_outbound_cmds	32	dst_host_count		
9	hot	21	is_hot_login	33	dst_host_srv_count		
10	num_failed_logins	22	is_guest_login	34	dst_host_same_srv_rate		
11	logged_in	23	count	35	dst_host_diff_srv_rate		
12	num_compromised	24	serror_rate	36	dst_host_same_src_port_rate		

III. NSL KDD BENCHMARK DATASET

When the IDS system is designed, the benchmark data set plays a vital role. This data set should contain the information about what should be the behavior of the various typical network protocol instances and properties in normal and as well as in case of any malign activity. Such data will become the key element to design a robust and strong IDS model. As much as the data set provide the information the detection becomes more accurate. One such very popular dataset is NSL-KDD dataset, and in the proposed paper this has been taken as benchmark dataset to implement the system [11]. Basically, the system will be designed by learning, from the data sets. NSL-KDD is the better version than its parent dataset called KDD'99. The KDD (Knowledge Discovery in Dataset) Cup was tools competition by International Knowledge Discovery and Data Mining in 1999. The aim of the competition was to collect records of data traffic and to build a predictive model which can differentiate between the normal network activity or the act of intrusion. Huge amount of data has been collected during the competition and bundled it to create a data set, which was called KDD'99. The NSL-KDD is the refined version of KDD'99 and it was developed by University of New Brunswick. The study of NSL-KDD data set says it has 42 attributes. These attributes can be considered as the features to define a particular intrusion type. The redundant and duplicate instances have been taken off from the KDD'99 to create newer version i.e. NSL-KDD. Duplications and redundancy in the instances lead to bias in the detection outputs. Table I. gives the description of NSL-KDD data set attributes.

This data set consists of 4 classes of attacks: Probe, Denial of Services (DOS), Remote to local and User to root (U2R). Categorization of various attacks depend on their behavior. Each attack will possess a peculiar behavior on the network. Figure 2. Shows the various classes of attacks and their examples.

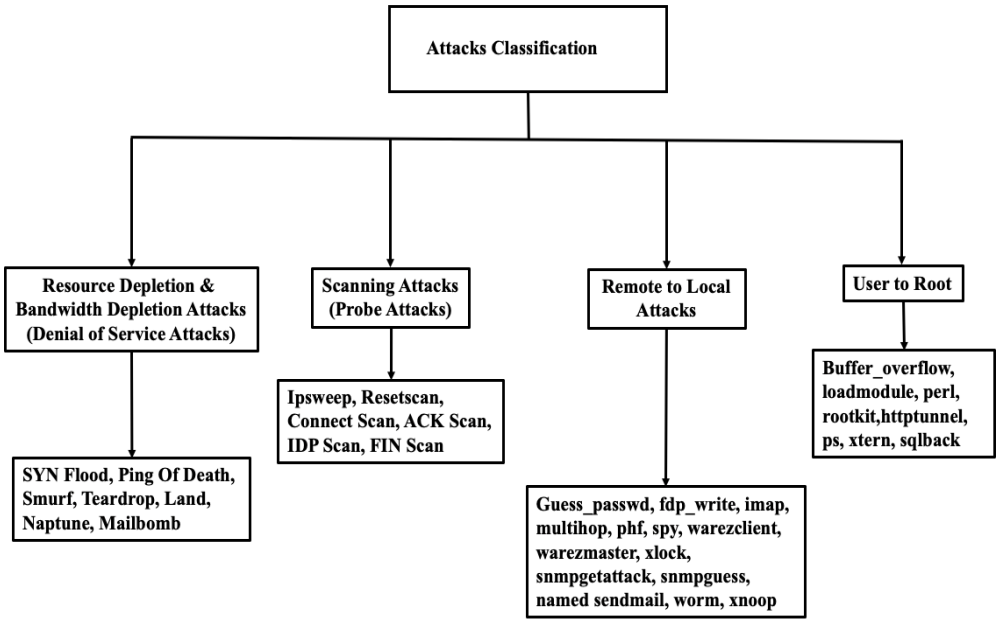

Fig. 2. Classification of Attacks

IV. UNIVARIATE ANOVA F-TEST

When data sets are considered for designing of the model, performance of the design depends on how well the model learns from the features present in the data set. The type of information for the learning task, totally revealed by the distinct features. However, the incorporation of all the feature or instances available in data set will lead to curse of dimensionality. Handling many features results in overfitting and increases complexity. To deal with these problems the features must be treated effectively. Dimensionality reduction is the solution for earlier mentioned issues. Its concept stresses on the point to reduce the number of featured which are redundant or similar. So, finding the subset of significant features will be a big challenge. In literature there have been wide range of feature selection algorithms, which have been proposed, such as PCA (Principle Component) and LDA (Linear Discriminant Analysis). One of such technique is ANOVA (Analysis of Variance) F Test, which is a statistical analysis performed on benchmark dataset to know feature characteristics [12]. The idea behind Univariate ANOVA F test is to select the features by the consideration of distribution of dataset and it selects the feature which are mostly related to the target attack. Categorical independence or dependency of the variables will be calculated by setting statistical tests. ANOVA results F statistics in order to find the ratio of the variation in means of categories to individual variations. The value of F-test directly has no impact, rather it is used as a statistical parameter for the measurement of significance in results. If it finds the variance of the features are equal or if not satisfying the statistical significance (p-

value<0.05 or 0.01) then the features will be excluded from the dataset [13]. If the dataset D containing n rows is applied for the feature selection process and undergo for ANOVA F test. Each row comprises of categorical variables, which are k continuous values. Let y_{ij} having j^{th} row in category i and $\bar{y}_i$ is mean of i. The overall mean is represented by $\bar{y}$. Let n_i be number of all the values present in the group. Then the F test can be performed by measuring the variations found among the means of each group and the variation of data values as overall data set. F statistics consists of S_a and S_e. First, S_a is the sum of squared error considering all sample mean compared to the mean weighted by the size of each group. So, S_a for the data set D can be given as:

$$S_a(D) = \sum_{i=1}^{k} n_i \, (\bar{y}_i - \bar{y})^2 \quad (1)$$

The second S_e, is the sum of squared error of all the values compared to the overall mean. And given as:

$$S_e(D) = \sum_{i=1}^{k} \sum_{j=1}^{n_i} \left(y_{ij} - \bar{y}_i\right)^2 \quad (2)$$

Then the F test is calculated as:

$$F(D) = \frac{S_e(D)/(k-1)}{S_a(D)/(n-k)} \quad (3)$$

This is how the ANOVA F test can be performed over the data to find the more significant features. If F statistic is provided, it is helpful to determine the probability p value which could be an evidence to remove the null hypothesis with respect to features.

V. DECISION TREES

Decision Tree is the most popular and efficient ensemble learning method. Among many machine learning algorithms Decision Tree algorithm best suited for classification of data, especially it performs well, when the data consists of network traffic information. Apart from classification this algorithm finds its use in decision theory, pattern recognition and statistics. It's a data exploration method to discover heuristics in datasets. It conquers the task by recursive and iterative partitioning of data based on underlying hierarchical and sequential structuring rules. The data exploration in decision tress can be done by many ways such as, Description, Classification and Generalization of data and its features variables [14]. In literature many variations to this concept has been suggested which are ID3, C4.5 CART (Classification and Regression Trees), CHAIDS and MARS. Decision tress classifies the data by forming a tree structures which consists of hierarchy of tree nodes and leaves. Here the nodes are the elements which takes part in decision making and recursively division of data can be performed. The algorithm goes in a step by step manner to classify the data and at each node an attribute will be considered. First the data is passed to the structure to make the Decision tree learn from the data and make the model ready for the testing. To enable this kind of methodology decision trees will take some statistical paraments into consideration those are Entropy, Information gain, Gini Index and Gain Ratio. These parameters are the deciding factor for feature selection at each node in the tree. The entropy in data is a measure of randomness in data values, if the randomness is high, it is difficult to divide the data at various stages of the tree structure. Entropy of the data D at a certain state s for only one feature can be given as,

$$E(s) = \sum_{x=1}^{k} -p_x \log_2 p_x \quad (4)$$

Where the p_x is probability of an event x of the state s. And for multiple features the entropy expression presented as:

$$E(T, X) = \sum_{c \in X} P(c) E(c) \quad (5)$$

Here, T is the current state and X represents the selected feature. One of the important parameters which could be considered while splitting the data sets for the designing the tree is Information Gain. This parameter is useful to measure how nicely the feature will be parted for training the data with respect to the targets. For the better learning of the algorithm information gain should be as high as possible. It calculates the variation between entropy before splitting and average entropy found after the split of the data. And the information gain can be given as:

$$Information\ Gain\ (T, X) = E(T) - E(T, X) \quad (6)$$

In a more intuitive way above expression can be presented as:

$$Information\ Gain = E(before) - \sum_{l=1}^{k} E(l, after) \quad (7)$$

Here in equation (7), "$before$" can be the data before parting and k could be the number of sub datasets obtained by splitting. $(l, after)$ can be the data l after splitting. Another parameter often taken into account while designing the decision tree is Gini Index. It works as evaluating criterion during the data splitting. It is useful for only binary splitting. Which is calculated as:

$$Gini\ Index = \sum_{x=1}^{c} (p_x)^2 \quad (8)$$

And the last parameter which should be calculated is Gain Ratio. It gives the ratio of Information Gain and partitioned data (w) Information. Represented as:

$$Gain\ Ratio = \frac{\left(E(before) - \sum_{l=1}^{k} E(l, after)\right)}{\sum_{l=1}^{k} w_l \log_2 w_l} \quad (9)$$

The above-mentioned various parameters will be considered for the design of the algorithm. Particular parameter will be selected according to the dataset and to be applied for the desired final model.

VI. METHODOLOGY

The proposed work based on the idea of feature selection by determining the statistical significance within the features using ANOVA F-Test. Then dataset will be applied to decision tree for the classification and detection of intrusion with reference to the mentioned malicious activities in the NSL-KDD dataset. The proposed method involves various steps in order to detect the type of attack by considering the present network traffic information. The software platform used to design the system is Python in Anaconda Distribution. The whole code is written and tested in Jupyter Notebook, which is an open source computing environment available in Anaconda Distribution. The figure 3

shows the steps to develop the complete system. The major important steps can be seen in the proposed IDS workflow are Data Preprocessing, Feature Selection, Model Building, Prediction and Model Evaluation. Foremost, data preprocessing step plays a vital role in the model development phase, which acts on the data to get the information about categorical features of various attacks and incorporates label encoding and One-hot encoding to convert the categorical names into numerical values. And also, the whole dataset after encoding split into 4 separate datasets of DoS, Probe, R2L and U2R. Often the values of features may of large values, which will create problems and prevents the model to learn the complete data set properly. Feature scaling is the method to tackle the earlier discussed situation. Features are scaled to a limited range of values. When an algorithm developed for certain applications it should be firstbuild and then tested for the desirable outcome. For two different tasks, dataset must be split into 2, one set meant for the learning purpose of the model, called as Training dataset and another should be used to test the model for knowing the performance, called as Testing dataset. Now another important part of the proposed idea is carried out, which does the Feature selection using ANOVA F-Statistics. 4 datasets features to be selected as per the statistical analysis and employed for model building using decision trees. The decision tree recursively divides the datasets based on the criteria (Information gain, Gini Index etc.). And finally, when the decision tree finishes with the learning from the features, the model is ready for the testing or in other words ready to detect the type of attack. The testing process reveals the true performance of the system and exposes the flaws by analyzing the confusion matrix and the evaluation metric drawn from it.

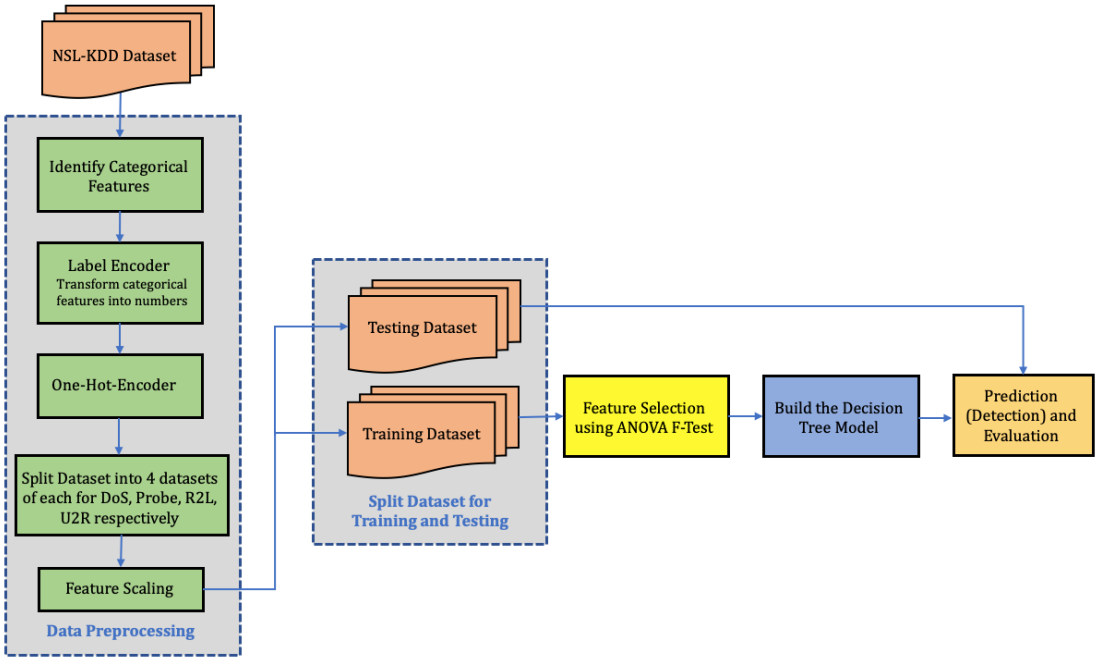

Fig. 3. Methodology- Workflow of IDS using ANOVA F-Test and Decision Trees

Basically, in classification applications the major evaluation metrics considered are accuracy, precision, recall and F1-Score. These metrics will be computed with the help of values which tells the rate of classifications. Such rates are as True positive, True Negative, False Positive and False Negative [15]. True positive (TP) gives the fact that actual instance is positive and predicted instance is also detected as positive. True Negative (TN) states the fact that actual instance is negative and predicted instance also negative. False Positive (FP) is when the actual instance is negative but results as positive. And False Negative (FN) represents that when the actual instance is positive and predicted as negative. These underlying metrics are helpful to determine the aforesaid performance metrics which are given as:

Accuracy measures the percentage of correct classification in total number of instances. And can be calculated as:

$$Accuracy = (\mathrm{TP} + \mathrm{TN})/(\mathrm{TN} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}) \quad (10)$$

Precision measures the number of correct detections with the number of incorrect instances. Calculated as:

$$Precision = \mathrm{TP}/(\mathrm{TP} + \mathrm{FP}) \quad (11)$$

Recall measures the number of correct predictions with the number of missed instances, given as:

$$Recall = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN}) \quad (12)$$

F1-Score measures the average of precision and recall. The harmonic mean of mentioned two metrics can be found as:

$$F1 - Score = \frac{2*(Recall*Precision)}{(Recall+Precision)} \quad (13)$$

With the assessment of these metrics the model is made ready for the real-time use. If the evaluation parameters give low range of values, then the parameters of the decision tree algorithms should be modified and again evaluated.

VII. RESULT AND DISCUSSIONS

Following table of values and graphs in this discussion represents the evidence to the performance of the proposed methodology. When the algorithm evaluated over the features for various performance measures, gives reasonably high scores. The algorithm has been developed by using scikit learn python library which is very popular to implement machine learning algorithms. Decision trees algorithm has various versions from its initially developed model to present version. Current scikit learn library has adopted optimized CART algorithm [16]. CART creates binary trees out of features and threshold values that produces the largest information gain for each node. Table II. Presents the percentages obtained after the evaluation of the proposed technique. Mentioned values for each kind of attack are acquired after the reduction in number of features. Out of 123 categorical and non-categorical features derived from the dataset 13 has selected by proposed feature selection technique. SelectPercentile is a class in the sk_learn feature selection package where the score function and percentile should be passed as parameter. In present work f_classif is chosen to be the first parameter which signifies the use of ANOVA F Test and the second parameter is for the percentage of features to be selected based on the feature selection technique used. Figure 4. Shows the Cross Validation score for the proposed model. Here from the figure, one can understand that the accuracies obtained for various values passed to the percentile parameter. When the value provided to be 100, it considers all the 123 features available in the dataset. Graph shows for each type of attack and it reveals that what will be the accuracy if the consideration of number of features varies. It has been seen that from percentile of 1 to 100 features, for value of 10 model is giving best results, for all four attacks. At the value of 10 the selected features by the ANOVA F Test is 13. Table II. Shows the best performance metric values when the features are 13. In case of DoS attacks accuracy and recall scores are high and gives the percentages as 99.64 and 99.67 respectively. Even the precision and F measure score close to 100 percent. The other two attacks Probe and R2L also yielding the closer values like DoS performance measures. But in case of U2R the values are little lagging behind than other rest of the attacks, apart from accuracy metric values, precision results in 86.74, Recall is 87.88 and 86.78. If the overall performance to be the interest then by considering accuracy, precision, recall and F measure altogether DoS attacks can be detected efficiently with the present techniques. Moreover, the proposed method performs very effectively to classify and detect attack from the underlying database NSL KDD. Also, in real time scenario if malicious activity observed and felt, passing the correct network traffic information as features to model will detect the presence of attack.

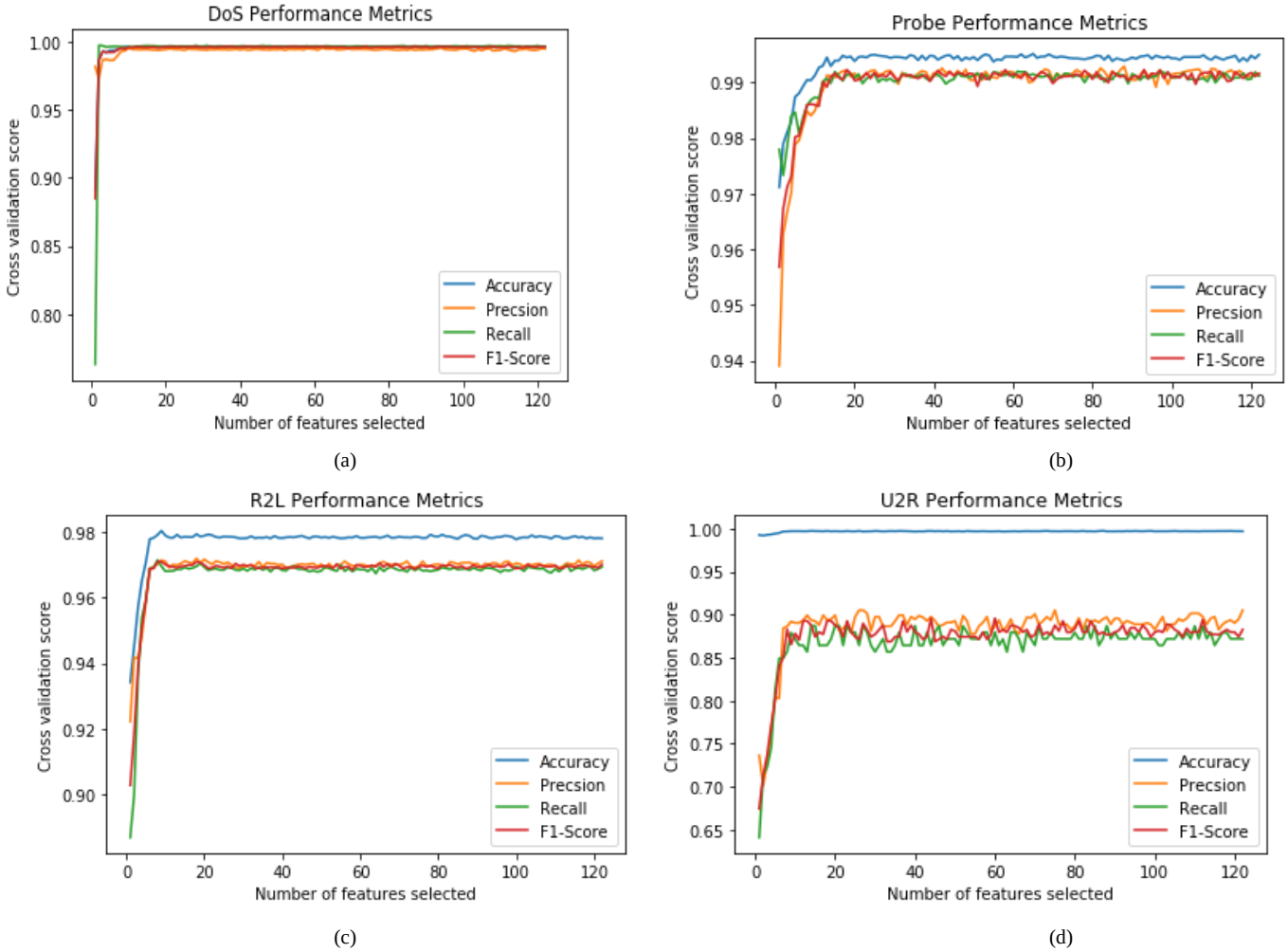

Fig. 4. Cross Validation Scores for Various Performance Metrics (a) Performance Metrics for DoS attacks (b) Performance Metrics for Probe attacks (c) Performance Metrics for R2L attacks (d) Performance Metrics for U2R attacks.

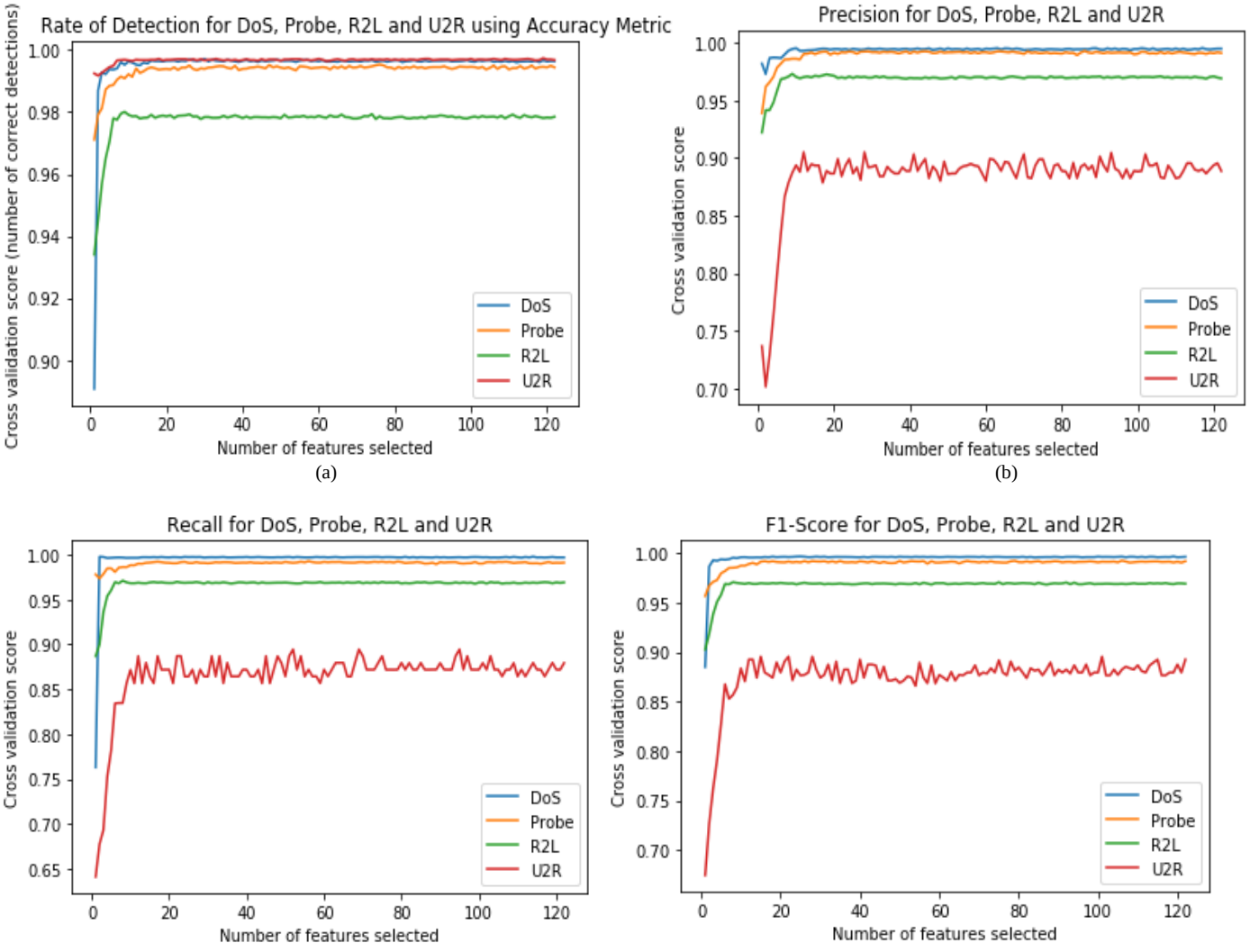

Fig. 5. Cross Validation Scores for (a) Rate of detection for DoS, Probe, R2L and U2R (b) Precision Score for all 4 attacks (c) Recall Score for all 4 attacks (d) F1-Score for all 4 attacks.

TABLE II
EVALUATED VALUES OF PERFORMANCE METRICS FOR OPTIMAL NUMEBER OF FEATURE VARIOUS ATTACKS

S.No.	Type of Attack	Accuracy	Precision	Recall	F-measure
1	DOS	99.64	99.53	99.67	99.58
2	Probe	99.56	99.34	99.26	99.32
3	R2L	97.92	97.15	96.95	97.05
4	U2R	99.64	86.74	87.88	86.78

Figure. 4. Shows various performance metrics for each of the attack when the number of selected features varies. From the graphs its evident that how well the proposed method is performing in all the evaluated metrics. And for each performance metric the comparative representation for all the attacks can be seen in fig. 5 and it has shown that which type attack is best detected by this method based on best features.

VIII. CONCLUSION

This paper evaluates the method developed to detect malicious attacks in any data communication network. However, many techniques proposed by researchers, performing well, but this present work gives a simple yet efficient technique to track the cyber-attacks. The design of the algorithm and the preparation of the dataset makes the detection task more accurate than

earlier developed methods. Majorly the design of the system concentrates on learning the features. It is not necessary to consider whole set of features. If the complete feature set used, then it deliberately increases the chances of false detection and consumes more time to process the results. Some features are redundant in case of some kind of attacks so ANOVA F Test efficient to find the essential feature information from the dataset and Decision Trees profoundly learns from the selected features and brings the outcome. The design works perfectly if the mentioned classes of attacks relevant features available in the real time monitoring scenario but if unfamiliar cyber-attack reaches the computer system or network it will not be possible to track. So consistent research is required to study the novel patterns in the network traffic relevant information. And there is a need to develop the mechanism to handle the great number of features and finding the significance in it. As the artificial intelligence and machine learning gaining popularity people are working to develop optimal ways to cop up with earlier mentioned problem. For the tasks like classification and detection, similar to Decision tree other algorithm are also be found to be useful those are Random Forest, Boosted Trees and XGBoost etc. Future works for the extension of this proposed work could be concentrating on applying and developing other feature engineering methods and classification algorithms.

REFERENCES

[1] Ektefa, M. Mohammadreza, S. Sara and A. Fatimah, “Intrusion detection using data mining techniques,” 200 - 203. 10.1109/INFRKM.2010.5466919.

[2] Y. Wang, W. Cai and P. Wei, “A Deep Learning Approach For Detecting Malicious Javascript Code,” Wiley Online Library . February 2016.

[3] C. Yin , Y. Zhu, J. Fei and H. Xinzheng, “A Deep Learning Approach For Intrusion Detection Using Recurrent Neural Networks,” IEEE Access. November 7, 2017.

[4] Q. Niyaz, W. Sun, Y Javaid and A. Mansoor, “A Deep Learning Approach For Network Intrusion Detection system,” In Eai Endorsed Transactions on Security and Safety, Vol. 16, Issue 9, 2016.

[5] M. Preeti, V. Vijay, T. Uday and S. P. Emmanuel, “A Detailed Investigation And Analysis Of Using Machine Learning Techniques For Intrusion Detection,” IEEE Communications Surveys & Tutorials, Volume: 21, Issue:1, First quarter 2019.

[6] Y. Li, M. Rong And R. Jiao, “A Hybrid Malicious Code Detection Method Based On Deep Learning,” International Journal of Software Engineering and Its Applications 9(5):205-216, May 2015.

[7] Gulshan and Krishan, “A Multi-Objective Genetic Algorithm Based Approach For Effective Intrusion Detection Using Neural Networks,” Springer. 2015.

[8] K. Levent and D. C. Alan, “Network Intrusion Detection Using A Hidden Naïve Bayes Binary Classifier,” 2015 17th Uksim-Amss International Conference on Modelling and Simulation (Uksim).

[9] A. Nadjaran, K. Mohsen, “A New Approach To Intrusion Detection Based On An Evolutionary Soft Computing Model Using Neuro-Fuzzy Classifiers,” July 2007, Computer Communications 30(10):2201-2212.

[10] D. Amin and R Mahmood, “Feature Selection Based On Genetic Algorithm And Support Vector Machine For Intrusion Detection System,” The Second International Conference On Informatics Engineering & Information Science (Icieis2013).

[11] A. Preeti and S. Sudhir, “Analysis of KDD Dataset Attributes - Class wise for Intrusion Detection,” Procedia Computer Science, Volume 57, 2015, 842-851,

[12] D. M. Doan, D. H. Jeong and S. Ji, “Designing a Feature Selection Technique for Analyzing Mixed Data,” 2020 10th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 2020, pp. 0046-0052, doi: 10.1109/CCWC47524.2020.9031193.

[13] Campbell and Zachary, “Differentially Private ANOVA Testing,” 2018 1st International Conference on Data Intelligence and Security (ICDIS) (2018): 281-285.

[14] S. K. Murthy, “Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey. Data Mining and Knowledge Discovery 2, 345–389 (1998).

[15] S. Dhaliwal, A. Nahid and R. Abbas, “Effective Intrusion Detection System Using XGBoost. Information 2018, 9, 149.

[16] Pedregosa et al., “Scikit-learn: Machine Learning in Python,” JMLR 12, pp. 2825-2830, 2011.