International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 04 Issue: 12 | Dec -2020 ISSN: 2582-3930

Evaluation of Different Feature Selection and Classification Methods for Intrusion Detection

Lynelle Fernandes
Department of Information Technology, S.F.I.T.
University of Mumbai
Mumbai,India.

Aneesha D'souza
Department of Information Technology, S.F.I.T.
University of Mumbai
Mumbai, India.

Voleta Noronha
Department of Information Technology, S.F.I.T.
University of Mumbai
Mumbai, India.

Alvina Alphonso
Department of Information Technology, S.F.I.T.
University of Mumbai
Mumbai, India.

Abstract*— In today's digital age, the protection of digital data storage has become quite an important phenomenon as this data can be accessed very easily. Over the past few years, there has been a drastic increase in cybercrime activities. Hackers use novel attacks to obtain sensitive information, due to this our data can be misused. Various preventive measures are used to protect this data, one of which is installing an Intrusion Detection System. Hence, an Intrusion detection system (IDS) is adopted to handle various network security threats. In this paper, we have performed varied methods of feature selection and classification techniques that can be used for an Intrusion detection system for detecting unknown or modified attacks. To evaluate the effectiveness of the IDS, the dataset named CICIDS2017, consisting of the latest threats is used. Relevant features are selected first using the different feature selection algorithms such as L2 regularization, correlation matrix, ExtraTree classifier, and chi-square. The selected features are tested using the inbuilt python classifiers like Naïve Bayes, Support Vector Machine (SVM) and k-nearest neighbours. Classification algorithm used are then compared. The results with the models are then noted. Results seen are above 90% for the performance metrics considered.

Keywords*— Intrusion Detection Systems (IDS), Classifier algorithm, CICIDS2017 dataset, Network Security, Feature selection, SVM.

I. INTRODUCTION

Due to the increase of cyber-criminal activities in recent years the protection of data stored on a digital platform has become quite an important phenomenon. Hackers use different types of attack in order to obtain a user's sensitive information. In the 2018 Internet Crime Report, the FBI's IC reports that the organization receives an average of 300,000 cybercrime-related complaints per year; that's an average of 900 complaints per day. Senior fraud scams are seen to be increasingly common and result in significant losses each year. Network security is a wide term to define. In its broader sense, we can say that it means to protect the confidential information or data which is stored on the network. Many organizations want to detect the intrusion in the network before they can be under attacked or to experience the loss of confidential data [7]. An intrusion detection system is the solution to this problem, as it is used in order to monitor the network in order to detect any malicious activity over the network and issues alerts when such an activity is discovered.

Intrusion detection systems are mainly used to monitor the network for any malicious activity. An intrusion detection system is installed in the network to monitor the network traffic on the subnet and attempts on matching the traffic on each subnet with the known attacks when such attack is detected alert is sent to the network administrator.

The proposed methodology in this paper mainly aims to focus on the following:

- To select dataset (CICIDS 2017) with selected attacks to be trained and tested.
- To use different feature selection techniques in order to select the best features for classification.
- To using different classification algorithms for detecting and classifying attack with the best results.

II. RELATED WORKS

A. A detailed analysis of CICIDS2017 dataset fordesigning Intrusion Detection Systems.

Various shortcomings of the dataset (CICIDS2017) have been studied and outlined. Moreover, a major issue of class imbalance has been reduced by class relabeling.[2]

B. A cross-comparison of feature selection algorithms on multiple cyber security data-sets.

When Decision Tree was run with ExtraTreeClassifier/SelectFromModel, the feature count was reduced to 26 and the combination managed to correctly label 82929 malicious network traffic and incorrectly classified 341 attacks in an execution time of 83.03 seconds whilst retaining an accuracy rating of 100%.[5]

C. A cross-comparison of feature selection algorithms on multiple cyber security data-sets.

LASSO regularization is a very effective methods of preventing overtraining in applications using multi-layer perceptron's. The only downside might be that finding the optimal regularization parameter requires some trial and error. The greatest benefit of using LASSO as a feature selection tool would be that it is very fast in comparison to most other methods. Judging by the current results, the features found are at least somewhat close to the truth in terms of importance. If nothing else, the method has proven to be very efficient in filtering out less relevant input variables.[6]

D. Selecting a Template Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization.

This paper discusses about the issues related with the generally used intrusion detection datasets such as KDD99, DARPA98, etc. A new IDS dataset that includes seven common updates family of attacks that met real worlds criteria is created and is publicly available at an (http://www.unb.ca/cic/datasets/IDS2017.html).[1]

III. Dataset Description

The dataset was obtained from the website unb.ca which is publicly available. The dataset download consists of 2 zip files. There are attack specific '.csv' files. We have considered 4 '.csv' files in which the attacks mentioned are DDoS/DoS, PosrtScan, Infiltration attack and Web Attacks. The dataset contains 79 features such as destination port, source port, flow duration, total forward packets, total backwards packets, min and max packet lengths, label, etc.

The dataset has 225746 tuples and 79 attributes. 1 out of the 79 attributes is the target attribute viz. label. The dataset is split into training data and testing data of 70% and 30%, respectively.:

Table I. Feature list of CICIDS2017 Dataset

Feature No.	Feature Name	Feature No.	Feature Name
1	Destination Port	**41**	Packet Length Mean
2	Flow Duration	**42**	Packet Length Std
3	Total Fwd packets	**43**	Packet Length Variance
4	Total Backward Packets	**44**	FIN Flag Count
5	Total Length of Fwd Packets	**45**	SYN Flag Count
6	Total Length of Bwd Packets	**46**	RST Flag Count
7	Fwd Packet Length Max	**47**	PSH Flag Count
8	Fwd Packet Length Min	**48**	ACK Flag Count
9	Fwd Packet Length Mean	**49**	URG Flag Count
10	Fwd Packet Length Std	**50**	CWE Flag Count
11	Bwd Packet Length Max	**51**	ECE Flag Count
12	Bwd Packet Length Min	**52**	Down/Up Ratio
13	Bwd Packet Length Mean	**53**	Average Packet Size
14	Bwd Packet Length Std	**54**	AvgFwd Segment Size
15	Flow Bytes/s	**55**	AvgBwd Segment Size
16	Flow Packets/s	**56**	Fwd Header Length
17	Flow IAT Mean	**57**	FwdAvg Bytes/Bulk
18	Flow IAT Std	**58**	FwdAvg Packets/Bulk
19	Flow IAT Max	**59**	FwdAvg Bulk Rate
20	Flow IAT Min	**60**	BwdAvg Bytes/Bulk
21	Fwd IAT Total	**61**	BwdAvg Packets/Bulk
22	Fwd IAT Mean	**62**	BwdAvg Bulk Rate
23	Fwd IAT Std	**63**	SubflowFwd Packets
24	Fwd IAT Max	**64**	SubflowFwd Bytes
25	Fwd IAT Min	**65**	SubflowBwd Packets
26	Bwd IAT Total	**66**	SubflowBwd Bytes
27	Bwd IAT Mean	**67**	Init_Win_bytes_forward
28	Bwd IAT Std	**68**	Init_Win_bytes_backward
29	Bwd IAT Max	**69**	act_data_pkt_fwd
30	Bwd IAT Min	**70**	min_seg_size_forward
31	Fwd PSH Flags	**71**	Active Mean
32	Bwd PSH Flags	**72**	Active Std
33	Fwd URG Flags	**73**	Active Max
34	Bwd URG Flags	**74**	Active Min
35	Fwd Header Len	**75**	Idle Mean
36	Bwd Header Length	**76**	Idle Std
37	Fwd Packets/s	**77**	Idle Max
38	Bwd Packets/s	**78**	Idle Min
39	Min Packet Length	**79**	Label
40	Max Packet Length		

There were 4 '.csv' files used to detect 4 different attacks.

A. DoS/DDoS attack:

This attack, just as the name suggests occurs when the attackers wants to disturb the access or deny the access to certain resource or contents available on the specific destination.

B. PortScan attack:

This attack mainly aims to avail an active port so as to exploit the vulnerability and creates a gateway to various other types of attack.

C. Web Attacks:

This attack occurs when an attacker aims to procure and/or modify the confidential data of a legitimate user over a web application. Examples: Brute Force, Cross Site Scripting (XSS), SQL injection (SQLi).

D. BotNet attack:

This type of attack uses a malware that is deployed in a system to control the system. That system behaves as a master to control the other infected systems also called as 'Zombie computers' to perform the attack. It is an attack in which a inter connected system devices are used to perform unethical activities.

The dataset used has in all 79 features and 225746 rows of data. This data is not suitable to be used in the models directly and need to be cleaned. Null and infinite values have to be normalized and scaled, accordingly. Normalizer() and MinMaxScaler() functions are used to do so. The use of 'Hot Encoding" makes it possible to convert the labels from string to suitable data type. This is necessary so as to be able to work with the dataset to avail better results. Once the dataset is cleaned and pre-processed it is ready to be trained and tested.

IV. Methodology Used

A. Feature Selection Techniques

1) L2 Regularization:

The regularization methods L1 and L2 regularization are also called, lasso and ridge regression. L2 regularization (called ridge regression for linear regression) is different as it adds the L2 norm penalty ($\alpha \sum_{i=1}^{n} w_i^2$) to the loss function. Since the coefficients are squared in the penalty expression, it has a different effect from L1-norm, namely it forces the coefficient values to be spread out more equally.[11]

On using this technique for feature selection, the number of features were reduced from 78 predictor features to the selected 27 features. The number of features whose coefficients shrunk to zero were 10

2) Correlation Matrix:

A correlation matrix is used to display the correlation of coefficients between variables. It is generally viewed using heatmaps. It is used to analyses the data which is summarized and helps in advanced diagnostic and analysis. a table showing correlation coefficients between variables. It is important to make correct decisions regarding choice of correlation statistic, coding of the variables, treatment of missing data, and presentation while using correlation matrix. This helps in deciding which feature are closely related to the label and provides a higher accuracy model.

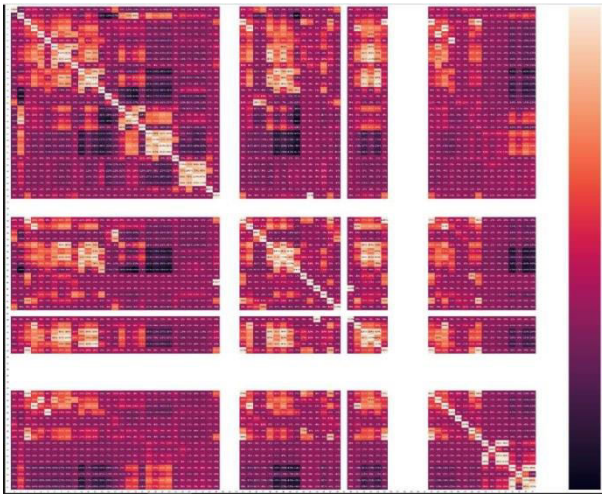

Fig. 1. Heatmap for Correlation matrix.

Fig. 1. Displays the correlation between the coefficients or features of the dataset. Features and their correlation are determined using the colors in the heatmap. Features highly correlated to the label are selected. 46 features are selected using this technique.

3) ExtraTree Classifier

Extremely Randomized Trees Classifier (Extra Trees Classifier) is similar to Random Forest Classifier. It is an ensemble learning technique where the results of multiple de-correlated decision trees are aggregated and collected in a "forest" to output its classification result. The original training sample is used to generate each Decision Tree in the ExtraTrees Forest. Then, each tree is provided with a random sample of k features, at each test node, from the feature-set from which the best feature must be selected from the decision tree to split the data based on some mathematical criteria (generally used Gini Index). This leads to the creation of multiple de-correlated decision trees from random sampling.

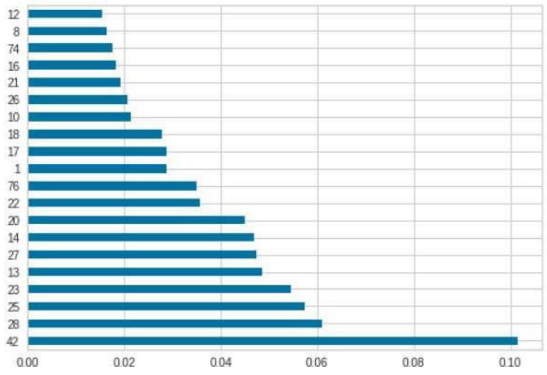

Fig. 2. Bar-graph for ExtraTree Classifier.

This feature selection technique selects the top 20 best features to be used for classification. A bar-graph was plotted based on the importance of the feature selected, as seen in Fig. 2.

4) Chi Square

Categorical features are selected using Chi-square. We calculate Chi-square is calculated between each feature and the target and is then used to select the desired number of features with best Chi-square scores. It helps to determines if the association between the two categorical variables in the sample would reflect in their real association in the population. It is assumed that, higher the Chi-Square value the feature is more dependent on the response and it can be selected for model training.

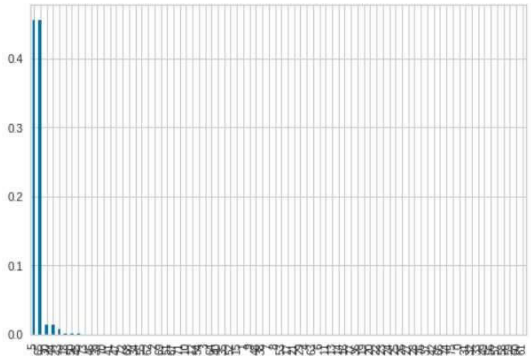

Fig. 3. Bar-graph for Chi square.

This technique helps to drop the features from 78 to 68. The NaN values obtained after Chi-square are dropped.

B. Classification Techniques

1) Naïve Bayes

A Naive Bayes classifier is a probabilistic machine learning model that can be used for classification-based problems. The classifier is based on the Bayes theorem.

Bayes' Theorem is denoted as:

$$P(h|d) = (P(d|h) * P(h)) / P(d)$$

where,
P(h|d) denotes the probability of hypothesis h given the data d. This is called the posterior probability.
P(d|h) denotes the probability of data d given that the hypothesis h was true.
P(h) denotes thee probability of hypothesis h being true. This is called the prior probability of h.
P(d) denotes the probability of the data.

Naive Bayes are fast and easy to implement however, the biggest disadvantage is that the predictors need to be independent. In real-life scenarios, the predictors are mostly dependent, this reducesthe performance of the classifier. Naive Bayes is considered as a powerful algorithm for predictive modelling. Naive Bayes is a commonly used classification algorithm for binary (two-class) and multi-class classification problems. Gaussian Naïve Bayes is used as we only need to estimate the mean and the standard deviation from your training data.

2) k-nearest Neighbours

The k-nearest neighbors (KNN) algorithm is a very simple, easy-to-implement supervised machine learning algorithm. It can be used to solve both classification as well as regression problems. The KNN algorithm makes an assumption that everything similar in nature exist in close proximity or are close to each other. The start of the 1970's has witnessed this algorithm being utilized for pattern recognition and statistical estimation, as a non-parametric technique.

K-Nearest Neighbours is one of the most basic yet one of the most essential classification algorithms used in Machine Learning. It can give highly competitive results. It is most commonly used for its ease of interpretation and low calculation time. The choice of the parameter K, in this algorithm is very crucial as the training error rate and the validation error rate are two parameters we need to access on different K-value.

It is widely used in real-life scenarios as it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data (as compared to other algorithms such as GMM, which assume a normal distribution of the given data).

3) Support Vector Machine (SVM)

Support Vector Machine (SVM) is a type of supervised machine learning algorithm that can be used for both classification as well as regression problems. It uses the technique called the kernel trick in order to transform data and find an optimal boundary, based on the transformation performed. It helps to separate data based on the labels defined. An SVM finds the best hyperplane that separates all data points of one class from the other class, for accurate classification. Larger margin between the two classes determines the best hyperplane for SVM. Margin is nothing but, the maximal width of the slab parallel to the hyperplane that has no interior data points.

Linear Kernel is used when the data can be Linearly separable, i.e. it can be separated using a single line. It is one of the most common kernels to be used. It is mostly used when there are a Large number of Features in a particular Data Set. Training a SVM with a Linear Kernel is much faster than with any other Kernel method. For linear kernel method, only the optimisation of the C Regularisation parameter is required while, when training with other kernel methods, there is a need to optimise the γ parameter which usually take more time.

Non-linear SVM means that the boundary isn't necessarily a straight line. The benefit of using this method is that it can capture more complex relationships between the datapoints without having to perform difficult transformations on your own. However, the training time is much longer as it's much more computationally intensive.

V. RESULTS

This paper uses the DDoS/DoS '.csv' file to perform the various feature selection techniques and classify them using the 2 classifiers mentioned. We are calculated the performance metrics of the correctly detected attacks.

A. *Performance measures used:*

Here, we calculate the attack occurred using these parameters of the confusion metrics TP, TN, FP, FN
where,
TP is True Positives, *TN* is True Negatives,
FP is False Positives, *FN* is False Negatives.

1) *Accuracy:*

One of the most commonly used performance metrics used in classification models. It has a general formula of:

$$Accuracy = \frac{Numbere\ of\ Correct\ Predictions}{Number\ of\ Total\ Predictions}$$

In our case of binary classification, accuracy is calculated in terms of positives and negatives as follows:

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

2) *Precision:*

Precision is defined as the ratio of correctly predicted positive observations (TP) to the total predicted positive observations (True and False values in Positives).

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

3) *Recall:*

It is also called as Sensitivity. It is defined as the ratio of correctly predicted positive observations to the all observations in actual class (here we consider it is Attack).

$$Recall(Sensitivity) = \frac{TP}{TP+FN}$$

4) *F1 Score:*

F1 Score can be defined using precision and recall parameters as, the weighted average of Precision and Recall. Therefore, it takes into account both, false positives and false negatives. F1 is usually more useful than accuracy, when you have an uneven class distribution.

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

B. *Naïve Bayes:*

TABLE II. PERFORMANCE METRICS WITH NAÏVE BAYES

Sr. No.	Feature Selection Technique Used	Performance Metrics			
		Accuracy	Precision	Recall	F1 Score
1.	L2 Regularization	0.939	0.987	0.873	0.927
2.	Correlation Matrix	0.947	0.990	0.889	0.936
3.	ExtraTree Classifier	0.902	0.925	0.844	0.883
4.	Chi Square	0.939	0.987	0.873	0.927

From Table II, we see that we get best accuracy and F1 score from using correlation matrix along with Naïve Bayes classifier. It can be inferred that with 46 predictor features we get the best accuracy of detecting attacks.

C. *k-nearest Neighbours:*

TABLE III. PERFORMANCE METRICS WITH K-NEAREST NEIGHBORS

Sr. No.	Feature Selection Technique Used	Performance Metrics			
		Accuracy	Precision	Recall	F1 Score
1.	L2 Regularization	0.999	0.999	0.999	0.999
2.	Correlation Matrix	0.999	1.00	0.999	0.999
3.	ExtraTree Classifier	0.998	0.999	0.999	0.999
4.	Chi Square	0.999	0.999	0.999	0.999

From Table III, we see that the best accuracy and F1 score is 0.999 with k-nearest Neighbors Classifier. It can be inferred that with any of these feature selection methods can be used. However, overfitting must be checked so as to infer the results correctly. Correlation matrix gives a precision rate of 1.00 and therefore, suggests it's the better feature selection method to be used.

D. *SVM:*

TABLE IV. PERFORMANCE METRICS WITH SVM

Sr. No.	Feature Selection Technique Used	Performance Metrics			
		Accuracy	Precision	Recall	F1 Score
1.	L2 Regularization	0.978	0.967	0.996	0.984
2.	Correlation Matrix	0.980	0.971	0.995	0.977
3.	ExtraTree Classifier	0.977	0.964	0.996	0.973
4.	Chi Square	0.978	0.967	0.996	0.974

From Table IV, we see that the best accuracy of 0.98 is obtained with Correlation matrix whereas, the best F1 score is obtained with L2 regularization. It can be inferred that with any of these two feature selection methods thee classifier gives an accurate detection rate. Both of the values are comparable.

VI. CONCLUSION AND FUTURE SCOPE

In this paper, various techniques were implemented for feature selection on CICIDS2017 dataset. The techniques were assessed based on the accuracy, precision, recall and f1-score of the classifiers used. The Correlation matrix feature selection technique gives the best accuracy of 94% using Naïve Bayes classifier and an accuracy of 99% using k-nearest Neighbours classifier. The accuracy of 98% is obtained using SVM. Through the feature selection technique, the dataset of 79 features was reduced to 46 features. Overfitting can be an issue due to which the k-nearest Neighbour classifier gave accuracies and f1 scores of 99%. The paper concludes the use

of Correlation matrix along with SVM classifier to get optimum and accurate detection of attack.

The training was done only on a selected dataset (DDoS/DoS) with a particular attack type and in order to increase the accuracy it needs to be trained on the dataset consisting of many other attack types. Here we have only considered the use of dataset to determine if the attack has occurred or not. The use of real-time data can be used to check real time accuracy and other performance metrics. The system is to be integrated into an application so that the interface is more accessible to the user. Live traffic on the network using a packet tracer must be used for real-time prediction of attacks.
Advanced machine learning models can we used to build the model or software and deploy them on the respective systems.

REFERENCES

[1] Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, pp. 108–116, Jan. 2018.

[2] R. Panigrahi and S. Borah, "A detailed analysis of CICIDS2017 dataset for designing Intrusion Detection Systems.," *International Journal of Engineering & Technology*, vol. 7, pp. 479–482, 2018.

[3] S. S. Panwar, Y. P. Raiwani, and L. S. Panwar, "Evaluation of Network Intrusion Detection with Features Selection and Machine Learning Algorithms on CICIDS-2017 Dataset," *SSRN Electronic Journal*, 2019.

[4] I. S. Thaseen and C. A. Kumar, "Intrusion detection model using fusion of chi-square feature selection and multi class SVM," *Journal of King Saud University - Computer and Information Sciences*, vol. 29, no. 4, pp. 462–472, 2017.

[5] A. Powell, D. Bates, C. Van Wyk, and A. D. de Abreu, "A cross-comparison of feature selection algorithms on multiple cyber security data-sets.," *Proceedings of the South African Forum for Artificial Intelligence Research* , vol. 2540, pp. 196–207, 2019.

[6] Erik Torstensson, "Using LASSO regularization as a feature selection tool." Bachelor thesis supervised by Mattias Ohlsson, Department of Astronomy and Theoretical Physics, Lund University, 2017. Accessed on: December 16, 2019. [Online]. Available: https://lup.lub.lu.se/student-papers/search/publication/8914341

[7] A. Iqbal and S. Aftab, "A Feed-Forward and Pattern Recognition ANN Model for Network Intrusion Detection," *International Journal of Computer Network and Information Security*, vol. 11, no. 4, pp. 19–25, Apr. 2019.

[8] G. Karatas and O. K. Sahingoz, "Neural network based intrusion detection systems with different training functions," *2018 6th International Symposium on Digital Forensic and Security (ISDFS)*, pp. 1–6, 2018.

[9] L. P. Dias, J. J. F. Cerqueira, K. D. R. Assis, and R. C. Almeida, "Using artificial neural network in intrusion detection systems to computer networks," *2017 9th Computer Science and Electronic Engineering (CEEC)*, pp. 145–150, Sep. 2017.

[10] A. Yulianto, P. Sukarno, and N. A. Suwastika, "Improving AdaBoost-based Intrusion Detection System (IDS) Performance on CIC IDS 2017 Dataset," *Journal of Physics: Conference Series*, vol. 1192, p. 012018, Mar. 2019.

[11] D. Agarwal, Adam, Paul, Brian, Deniel, S. S., Ebrahimi, and L. Asadzadeh, *Diving into data*, 12-Nov-2014. [Online]. Available: https://blog.datadive.net/selecting-good-features-part-ii-linear-models-and-regularization/. [Accessed:28-Feb-202]