ANALYSIS OF MACHINE LEARNING ALGORITHMS WITH FEATURE SELECTION FOR INTRUSION DETECTION USING UNSW-NB15 DATASET

Geeta Kocher[1] and Gulshan Kumar[2]

[1]Research Scholar, Department of Computational Sciences, MRSPTU, Bathinda, Punjab
[2]Assosiate Professor, Department of Computer Applications, SBSSTC, Ferozpur, Punjab

ABSTRACT

In recent times, various machine learning classifiers are used to improve network intrusion detection. The researchers have proposed many solutions for intrusion detection in the literature. The machine learning classifiers are trained on older datasets for intrusion detection, which limits their detection accuracy. So, there is a need to train the machine learning classifiers on the latest dataset. In this paper, UNSW-NB15, the latest dataset is used to train machine learning classifiers. The selected classifiers such as K-Nearest Neighbors (KNN), Stochastic Gradient Descent (SGD), Random Forest (RF), Logistic Regression (LR), and Naïve Bayes (NB) classifiers are used for training from the taxonomy of classifiers based on lazy and eager learners. In this paper, Chi-Square, a filter-based feature selection technique, is applied to the UNSW-NB15 dataset to reduce the irrelevant and redundant features. The performance of classifiers is measured in terms of Accuracy, Mean Squared Error (MSE), Precision, Recall, F1-Score, True Positive Rate (TPR) and False Positive Rate (FPR) with or without feature selection technique and comparative analysis of these machine learning classifiers is carried out.

KEYWORDS

Intrusion Detection System, MSE, SGD, UNSW-NB15, Machine Learning Algorithms

1. INTRODUCTION

Nowadays, it is challenging to protect confidential data from the eye of attackers. The traditional methods like firewall and antivirus failed to handle all types of attacks. So, there is a need for additional security along with traditional methods. IDS play a significant role in this regard. It carefully keeps a track on the network traffic data and differentiates the data as normal or attack.

1.1. Intrusion Detection System

An IDS is used to monitor the network traffic for detecting malicious activity. It can easily detect the attacks that are bypassed by the firewall. It continuously monitors the network, finds vulnerable parts of the network, and communicates with the administrator about intrusions [1]. It can be separated into two classes: anomaly detection and misuse detection. Misuse detection operates with prior prepared patterns of known attacks, also called signatures [2]. It has high accuracy and low false alarm rates (FAR) but cannot detect novel attacks [3]. One solution to address this problem is to regularly update the database, which is not feasible and costly. So, anomaly detection techniques came into existence. Anomaly Detection deals with profiling user behaviour [4]. In this approach, a particular user regular activity model is defined, and any deviation from this model is known as abnormal. Fig. 1 shows the diagram of IDS.

DOI: 10.5121/ijnsa.2021.13102

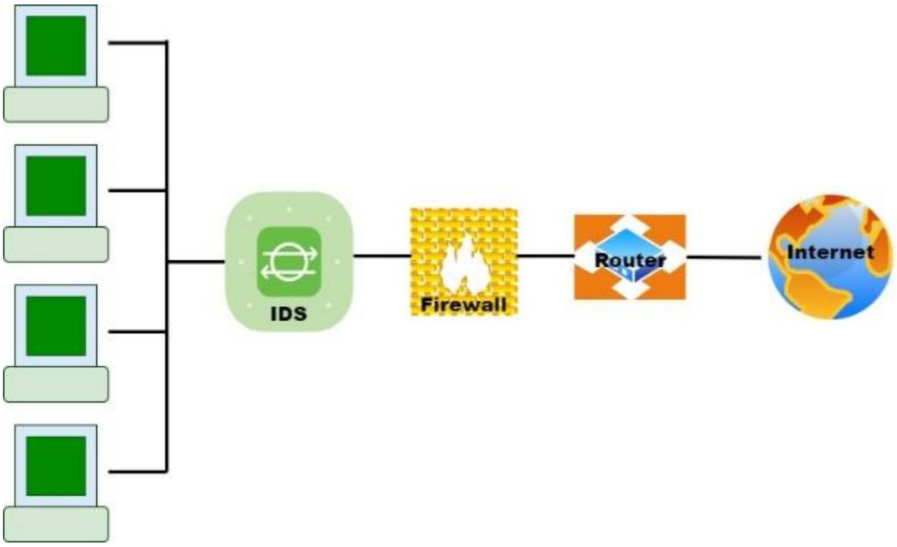

Fig 1: Intrusion Detection System

In literature, different types of machine learning (ML) algorithms are used for intrusion detection. The literature found that there is little work done on comparative analysis of ML algorithms. Hence, this paper aims to conduct a performance comparison of several ML algorithms using the recent dataset for intrusion detection.

The structure of the paper is divided into seven sections. Section2 gives light on the background behind it. In section 3, the taxonomy of the classifiers is discussed. A brief introduction about the dataset used for experimental work is described in Section 4. In Section 5, a methodology to pre-process the dataset is presented. Section 6 explains the feature selection techniques. Experimental work is shown in Section 7, and Section 8 gives a conclusion and future scope.

2. BACKGROUND

This section provides the literature survey on the ML algorithms. This section's main motive is to give an overview of the research work done in the field of intrusion detection. It is found in the literature that researchers have put a lot of efforts into ML algorithms, and some of their contributions are described below:

Narudin et al. (2014) [5] described an evaluation of ML classifiers, namely RF, J-48, MLP, NB, and KNN, to detect mobile malware using MalGenome and private datasets using Weka Tool. The performance metrics such as TPR, FPR, precision, recall and F-measure were used to validate ML algorithms' performance. The accuracy obtained using RF Classifier is 99.99% during experimental work on MalGenome dataset. The author has suggested the use of feature selection methods for improving the results in their future work.

Belavagi & Muniyal, (2016) [6] designed a NIDS with the various supervised machine learning classifiers. NSL-KDD dataset was used to check the performance of various classifiers. The result shows that RF classifier outperforms other classifiers. It results in the lowest FPR and the highest TPR and accuracy obtained is 99%. But still, there is a need for classifiers that can be used for the multiclass classification.

Ashfaq et al. (2017) [7] described a semi-supervised learning (SSL) approach based on novel fuzziness. To improve the classifier performance, it utilizes unlabelled samples along with a supervised learning algorithm. NSL-KDD dataset was used for the evaluation of this model. The limitation of this model was that its performance was studied only for the Binary classification task.

Yaseen et al. (2017) [8] described a multilevel hybrid intrusion detection model using SVM and EVM. The evaluation was done on KDD 99 dataset. The accuracy obtained was 95.75% and shorter training time in this proposed model. This technique is better only for known attacks and for novel attacks, efficient classifiers are required.

Aljumah, (2017) [9] described a trained algorithm to detect DDoS attacks that were based on Artificial Neural Network (ANN). ANN showed 92% accuracy when it was trained with older datasets, and when the system is trained with updated datasets, the accuracy obtained was 98%. The accuracy of the ANN model depends upon the dataset. So, there is a need forupto date and balanced dataset.

Roshan et al. (2018) [10] discussed an adaptive design of IDS based on Extreme Learning Machines (ELM). The NSL-KDD dataset was applied for the evaluation. It was found that it can detect novel attacks and known attacks with an acceptable rate of detection and false positives.

Ali et al. (2018) [11] proposed a PSO-FLN classifier for intrusion detection. The benchmark dataset KDD99 was used to validate the results. PSO-FLN has outperformed ELM and FLN classifiers in terms of accuracy. But for some classes like R2L, it does not show accurate results.

Anwer et al. (2018) [12] proposed a feature selection framework for efficient network intrusion detection. The UNSW-NB15 data set was used to evaluate five different strategies. J48 and NB methods were used for evaluation. The best strategy was to use a filter ranking method to select 18 features, and then J48 was applied as a classifier. By using this strategy, 88% accuracy and a speedup factor of 2 was achieved.

Hajisalem et al. (2018) [13] proposed a new hybrid classification method based on Artificial Fish Swarm and Artificial Bee Colony algorithms to enhance the Detection Rate (DR) of IDS. In this approach, generated rules were used to train the hybrid method. The simulated results on UNSW-NB15 dataset showed that the proposed method achieved 98.6% DR, 98.9% accuracy and 0.13% FPR.

Viet et al. (2018) [14] described a scanning method based on Deep Belief Network (DBN). It was performed by supervised and unsupervised ML methods along with DBN. UNSW-NB15 dataset was used to find out the attack in the form of binary classification. DBN results were compared with Support Vector Machine (SVM) and RF. The results obtained were TPR 99.74%, 99.80%, 99.86% and FAR 3.20%, 3.31%, 2.76% for SVM, RF and DBN respectively.

The literature survey concluded that most of the research had been validated using older datasets. These datasets lack novel attacks and contain imbalanced network audit data. Non-uniform data distribution may lead to ML algorithms' biased training, and this problem needs to be resolved. The new dataset can be used to detect novel attacks. A few researchers have utilized the UNSW-NB15 dataset, a newer dataset containing novel attacks but still needs to be explored. So, in this paper, the UNSW-NB15 dataset is used. There are many classifiers used for research in literature out of which the RF classifier shows better results.

3. TAXONOMY OF CLASSIFIERS

The classifiers are divided into two classes based on their learning methods, i.e. lazy and eager learners [15-18]. The proposed taxonomy of classifiers is based on theoretical analysis and is shown in Fig. 2.

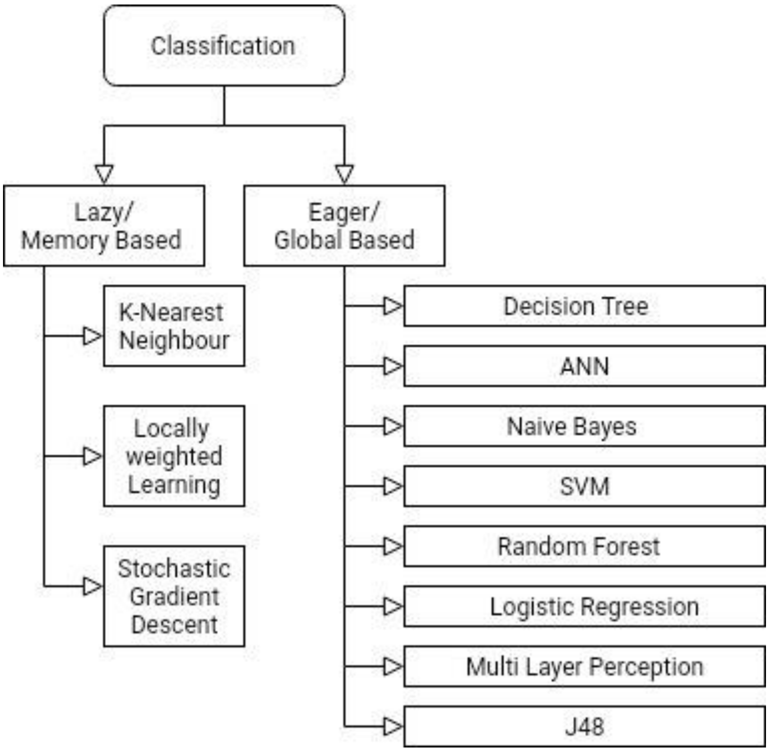

Fig 2: Taxonomy of classifier

Lazy learners can store examples and solve multiple problems with these examples. These learners adapt automatically to changes in the problem domain and easy to maintain. But the limitation of these learners is that they stored the same kind of examples often and require high memory and time-consuming learners. Eager learners firstly build a classification model on given training data and then perform a classification. These learners take more time for learning and less time for classifying the data.

KNN, SGD, NB, RF, and LR classifiers are used for conducting experimental work in this paper by using the proposed taxonomy. The description of these classifiers is provided follows:

3.1. Lazy Learners

These learners use the training data for storage and wait for testing data to appear. KNN, locally weighted learning (LWL) and SGD are examples of Lazy learners.

3.1.1. K- Nearest Neighbour

Itis a lazy learning algorithm that firstly stores all the training data. At the time of classification, it uses this data and tries to find the similarities between the new data and the available data. It places the new data in the category that is most similar to the available data. It is based on the

Euclidean distance [19]. The test data is allotted to the class of its K nearest neighbours. As you increase the value of K, accuracy might increase. It can be used for both regression and classification but is often used for classification problems.

3.2. Eager Learners

Eager learners take a long time for training and less time for predicting. NB, LR, SVM, RF, MLP, and J48 are examples of eager learners.

3.2.1. Logistic Regression

It is applied to solve both binary class and multiclass classification problems. The probability of occurrence of an event is predicted by giving fitting data to Logistic function. The output of this function lies between 0 and 1. The median value, i.e. 0.5 is considered as the threshold between class 1 and class 0. The output greater than 0.5 is considered class 1 and if output is below 0.5, then it is considered class 0 [6].

3.2.2. Random Forest

It was proposed by Breiman in 2001. This method is based on the proximity search and can be used both for regression and classification. It is a decision tree-based classifier. In this technique, random samples are used to create decision trees, and then the prediction is made from each tree. The best solution is found out by the voting method [19]. The random forest has many applications like image classification, feature selection and recommendation engines.

3.2.3. Naive Bayes

It is a classification algorithm used both for two-class and multiclass classification problems. It assumes that every feature's probabilities belonging to each class are used for prediction [6]. It also assumes that the probability of every feature belonging to a given class value is independent of other features. For the known value of the feature, the probability is known as conditional probabilities. Prediction can be attained by calculating each class's instance probabilities and selecting the highest probability class value [15].

4. DATASET USED

The benchmark datasets used in the literature are older datasets and contain repeated records due to which ML algorithms give unfair results. So, selected ML algorithms are tested using UNSW-NB15 dataset, which is novel dataset [20]. This dataset comprises 49 attributes, including a class label and 25, 40, 044 labelled instances, each being labelled either normal or attack. A detailed description of the features is given in Table 1.Table 2 gives the details of the attacks.

Table 1: Description of the attributes of UNSW-NB15 dataset

S.No.	Type of attributes	Name of attributes	Sequence No.
1	Flow	Script, Sport, Dstip. Dsport, Proto	1-5
2	Basic	State, Dur, Sbytes, Dbytes, Sttl, Dttl, Sloss, Dloss, Service, Sload, Dload, Spkts, Dpkts	6-18
3	Content	Swin, Dwin, Stepb, Dtcpb, Smeansz, Dmeansz, trans_depth, res_bdy_len	19-26
4	Time	Sjit, Djit, Stime,Ltime, Sintpkt, Dintpkt, Tcprtt, Synack, Ackdat	27-35
5	General Purpose	is_sm_ips_ports, ct_state_ttl, ct_flw_http_mthd,is_ftp_login, ct_ftp_cmd	36-40
6	Connection	ct_srv_src,ct_srv_dst, ct_dst_ltm, ct_src_ ltm ,ct_src_dport_ltm,ct_dst_sport_ltm, ct_dst_src_ltm	41-47
7	Labelled	attack_cat, Label	48-49

When the UNSW-NB15 dataset is used for evaluation, out of 49 attributes, we got 45 attributes only. The four ID attributes are combined to make a single attribute as ID from flow attribute category. Two attributes of time category (Stime and Ltime) are combined in one attribute known as Rate.

Table 2: Types of attacks in the dataset

Type	**Whole**	**Training**
	No. of Records	**No. of Records**
Normal	2218761	56000
Fuzzers	24246	18184
Analysis	2677	2000
Backdoors	2329	1746
DOS	16353	12264
Exploits	44525	33393
Generic	215481	40000
Reconnaissance	13987	10491
ShellCode	1511	1133
Worms	174	130
		175341

5. Feature Selection Techniques

This section describes different feature selection methods. The feature selection methods are used to select the relevant features [12]. These methods select a subset of relevant features for constructing the model and are categorized into the filter, wrapper and hybrid techniques. Filter methods measure each feature independent from the classifier, rank them and take the superior ones. Chi-square is one of the examples of filter-based methods. Wrapper methods are dependent on the selected classifier. These methods take a subset of the feature set to measure classifier performance, and then the next subset is assessed on the classifier. The subset which gives utmost performance is selected. RFE is an example of Wrapper methods. Hybrid methods utilize the merits of both filter and wrapper methods. In this paper for the experimental purpose, Chi-square from Filter method is used. Different Feature selection techniques are shown in Fig. 3.

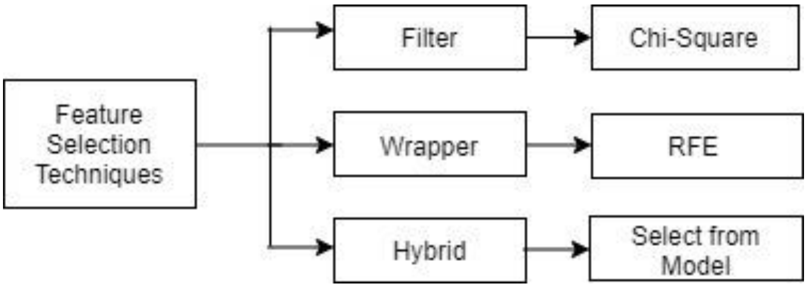

Fig 3: Feature Selection Techniques

6. METHODOLOGY

The pre-processing steps are shown in Fig.4, and Fig. 5 shows the methodology used. In pre-processing, first of all, the null values present in the dataset are handled. Then the categorical data is converted into the numerical form using label encoder. After this, one hot encoder is used to break the relation between the values obtained through label encoder. The standard scaler is used to standardize the values at one scale.

After performing pre-processing, significant features are selected using chi-square, and after this, the pre-processed data is separated into training and testing data. 80% of data is used for training, and 20% of data is used for testing. The KNN, LR, NB, SGD and RF classifiers are used to construct the models. Then the prediction of labels of test data is made using these models. A comparison is carried out between actual labels and predicted labels. The performance metrics used to evaluate the models are accuracy, precision, mean square error (MSE), recall, f1-score, TPR and FPR.

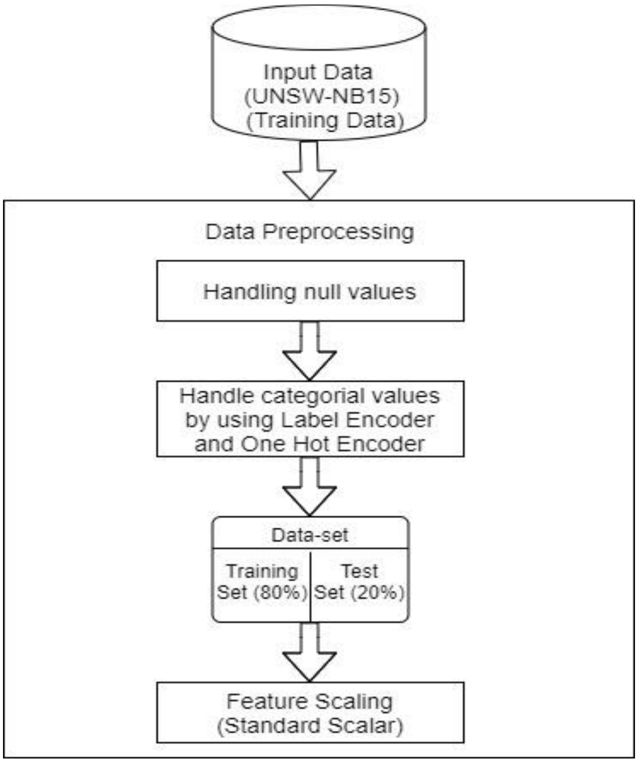
Input Data (UNSW-NB15) (Training Data)
Data Preprocessing
Handling null values
Handle categorial values by using Label Encoder and One Hot Encoder
Data-set
Training Set (80%)
Test Set (20%)
Feature Scaling (Standard Scalar)

Fig 4: Pre-processing steps on the UNSW-NB15 dataset

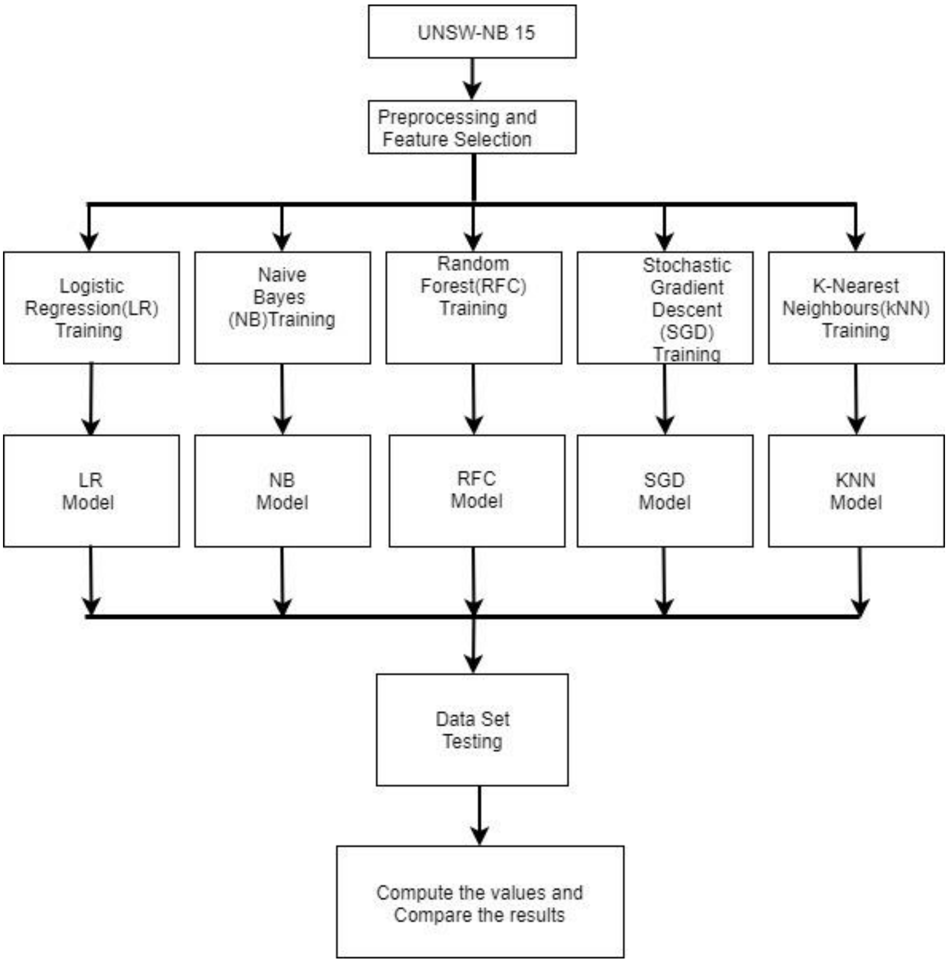
UNSW-NB 15
Preprocessing and Feature Selection
Logistic Regression(LR) Training
Naive Bayes (NB)Training
Random Forest(RFC) Training
Stochastic Gradient Descent (SGD) Training
K-Nearest Neighbours(kNN) Training
LR Model
NB Model
RFC Model
SGD Model
KNN Model
Data Set Testing
Compute the values and Compare the results

Fig 5: Experimental methodology

The procedural steps to construct the models are given below:

1. Start with pre-processing of the dataset.
2. Select the significant features using the chi-square method.
3. Divide dataset into two parts, i.e. training and testing.
4. Construct the classifier model using training data for KNN, LR, NB, RF and SGD.
5. Take the test data.
6. Testing of classifier models using training data.
7. Calculate and compare Accuracy, Recall, Precision, F1-Score and MSE for the selected models.

7. EXPERIMENTAL WORK

The selected ML algorithms, namely LR, NB, RF, SGD and KNN classifiers, are tested on UNSW-NB15 dataset, the novel dataset for intrusion detection. The experimental work is done on Intel Core (TM) i3-1005G1 CPU @1.20 GHz, 4GB RAM using Python. After performing pre-processing steps, 23 significant features are selected, and the dataset is divided into two parts: training and testing data. Then five classifiers are used for training as shown in Fig.5. Performance is evaluated based on several parameters using all features and selected features, as shown in Table 3 and Table 4, respectively. Fig. 6and Fig. 7show the pictorial representation of selected classifiers' accuracy on all features and using selected features, respectively.

It can be observed from the results shown in Table 3 that the RF classifier outperforms the other methods in terms of accuracy 99.57%, TPR 0.997 and MSE 0.004. In contrast, the NB shows the highest MSE 0.234 and lowest accuracy of 76.59% in the selected group of classifiers.

Table 3: Performance comparison of selected classifiers using UNSW-NB15 dataset with train test split method

Classifier	Accuracy	Precision	Recall	F1-Score	MSE	TPR	FPR
LR	98.42	0.98	0.99	0.99	0.015	0.994	0.048
NB	76.59	1.00	0.69	0.82	0.234	0.693	0.007
RF	99.57	1.00	1.00	1.00	0.004	0.997	0.009
SGD	98.16	0.98	1.00	0.99	0.018	0.983	0.043
KNN	98.28	0.99	0.99	0.99	0.017	0.989	0.039

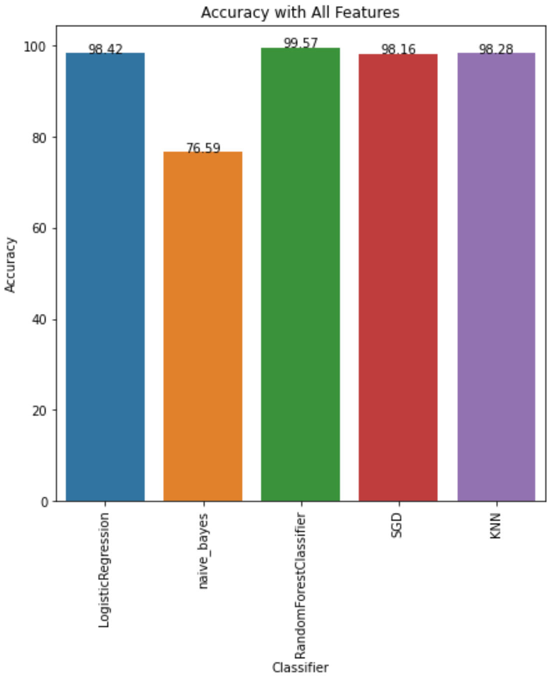

Fig 6: Accuracy of selected classifiers using UNSW-NB15 dataset

Table 4: Performance comparison of selected classifiers using UNSW-NB15 dataset with the feature selection method

Classifier	Accuracy	Precision	Recall	F1-Score	MSE	TPR	FPR
LR	98.17	0.98	1.00	0.99	0.018	0.996	0.063
NB	75.16	1.00	0.67	0.80	0.248	0.674	0.005
RF	99.64	1.00	1.00	1.00	0.003	0.998	0.009
SGD	97.99	0.98	1.00	0.99	0.020	0.997	0.073
KNN	98.90	0.99	0.99	0.99	0.011	0.993	0.025

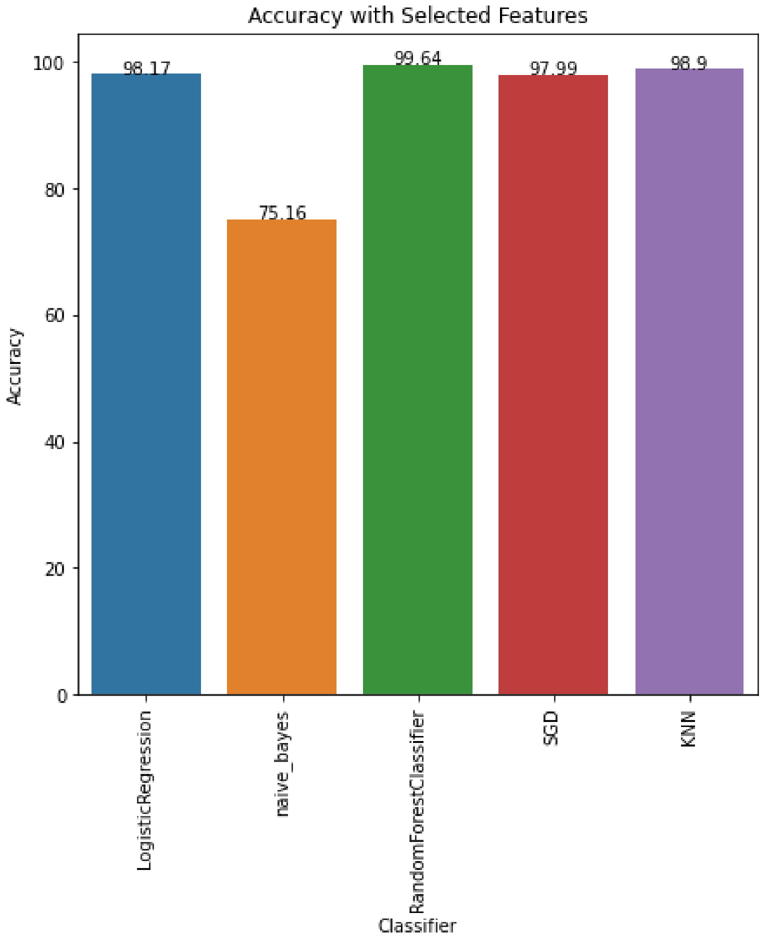

Fig 7: Accuracy of selected classifiers with selected features using UNSW-NB15 dataset

8. Conclusion and Future Scope

Experimental work has been carried out to evaluate the ML classifiers' performance, namely KNN, LR, NB, SGD and RF for detection of intrusion. These classifiers are tested onUNSW-NB15 dataset. The classifiers are compared based on precision, MSE, recall, F1-Score, accuracy, TPR and FPR. The results show that RF classifier is better than other classifiers on UNSW-dataset using selected parameters. The accuracy of RF classifier comes out to be 99.57% with all features and 99.64% with selected features. In future, this work can be extended for multiclass classification intrusion detection.

References

1. Sarmah, A. (2001). Intrusion detection systems: Definition, need and challenges.
2. Omer, K. A. A., & Awn, F. A. (2015). Performance Evaluation of Intrusion Detection Systems using ANN. Egyptian Computer Science Journal, 39(4).
3. Diro, A. A., & Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for Internet of Things. Future Generation Computer Systems, 82, 761-768.
4. Khan, J. A., & Jain, N. (2016). A survey on intrusion detection systems and classification techniques. Int. J. Sci. Res. Sci., Eng. Technol., 2(5), 202-208.
5. Narudin, F. A., Feizollah, A., Anuar, N. B., &Gani, A. (2016). Evaluation of machine learning classifiers for mobile malware detection. Soft Computing, 20(1), 343-357.

6. Belavagi, M. C., &Muniyal, B. (2016). Performance evaluation of supervised machine learning algorithms for intrusion detection. Procedia Computer Science, 89, 117-123.
7. Ashfaq, R. A. R., Wang, X. Z., Huang, J. Z., Abbas, H., & He, Y. L. (2017). Fuzziness based semi-supervised learning approach for intrusion detection system. Information Sciences, 378, 484-497.
8. Al-Yaseen, W. L., Othman, Z. A., &Nazri, M. Z. A. (2017). Multilevel hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system. Expert Systems with Applications, 67, 296-303.
9. Aljumah, A. (2017). Detection of Distributed Denial of Service Attacks Using Artificial Neural Networks. International Journal of Advanced Computer Science and Applications (IJACSA), 8(8).
10. Roshan, S., Miche, Y., Akusok, A., & Lendasse, A. (2018). Adaptive and online network intrusion detection system using clustering and Extreme Learning Machines. Journal of the Franklin Institute, 355(4), 1752-1779.
11. Ali, M. H., Al Mohammed, B. A. D., Ismail, A., &Zolkipli, M. F. (2018). A new intrusion detection system based on Fast Learning Network and Particle swarm optimization. IEEE Access, 6, 20255-20261.
12. Anwer, H. M., Farouk, M., & Abdel-Hamid, A. (2018, April). A framework for efficient network anomaly intrusion detection with features selection. In 2018 9th International Conference on Information and Communication Systems (ICICS) (pp. 157-162). IEEE.
13. Hajisalem, V., &Babaie, S. (2018). A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection. Computer Networks, 136, 37-50.
14. Viet, H. N., Van, Q. N., Trang, L. L. T., & Nathan, S. (2018, June). Using deep learning model for network scanning detection. In Proceedings of the 4th International Conference on Frontiers of Educational Technologies (pp. 117-121).
15. Bhavani, D.D., Vasavi, A. & Keshava PT (2016). Machine Learning: A Critical Review of Classification Techniques. IJARCCE, 5(3), 22-28.
16. Wei, C. C. (2015). Comparing lazy and eager learning models for water level forecasting in river-reservoir basins of inundation regions. Environmental Modelling & Software, 63, 137-155.
17. Rafatirad, S., & Heidari, M. (2018). An Exhaustive Analysis of Lazy vs. Eager Learning Methods for Real-Estate Property Investment.
18. https://towardsdatascience.com/machine-learning-classifiers-a5cc4e1b0623
19. Narudin, F. A., Feizollah, A., Anuar, N. B., &Gani, A. (2016). Evaluation of machine learning classifiers for mobile malware detection. Soft Computing, 20(1), 343-357.
20. Moustafa, N., & Slay, J. (2016). The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. Information Security Journal: A Global Perspective, 25(1-3), 18-31

AUTHORS

Ms. Geeta Kocher

She is MCA, M.Tech, M.Phil. She is pursuing a Ph.D. in the field of Artificial Intelligence-Deep learning. She has published more than 15 papers in various conferences and journals. She has more than 16 years of teaching experience.

Dr. Gulshan Kumar

Dr. Gulshan Kumar has received his MCA degree from Guru Nanak Dev University Amritsar (Punjab) India in 2001, and his M.Tech. in Computer Science & Engineering from JRN RajasthanVidyapeeth Deemed University, Udaipur (Rajasthan)-India, in 2009. He got his Ph.D. from Punjab Technical University, Jalandhar (Punjab)-India. He has 17 years of teaching experience. He has 56 international and national publications to his name. He is currently working as an Associate Professor in Computer Applications department at Shaheed Bhagat Singh State Technical Campus, Ferozepur (Punjab)-India. He has supervised 06 M. Tech. students for their final thesis, students for projects MCA, and supervising 02 Ph.D. research scholars. His current research interests involve Artificial Intelligence, Network Security, Machine Learning, and Databases.