Performance Evaluation of Machine Learning Algorithms for Hypertext Transfer Protocol Distributed Denial of Service Intrusion Detection

Rukayya Umar
Department of Cyber Security Science
Federal University of Technology
Minna, Nigeria.
umarrukayya1@gmail.com

Morufu Olalere
Department of Cyber Security Science
Federal University of Technology
Minna, Nigeria
lerejide@futminna.edu.ng

Ismaila Idris
Department of Cyber Security Science
Federal University of Technology
Minna, Nigeria
ismi.idris@futmina.edu.ng

Raji Abdullahi Egigogo
Department of Cyber Security Science
Federal University of Technology
Minna, Nigeria
raji.pg610868@st.futminna.edu.ng

Bolarin G
Department of Computer Science
Federal University of Technology
Minna, Nigeria
bolarin@futminna.edu.ng

Abstract*— As this paper has expounded, the techniques against DDoS attacks borrow greatly from the already tested traditional techniques. However, no technique has proven to be perfect towards the full detection and prevention of DDoS attacks. Intrusion detection system (IDS) using machine learning approach is one of the implemented solutions against harmful attacks. However, achieving high detection accuracy with minimum false positive rate remains issue that still need to be addressed. Consequently, this study carried out an experimental evaluation on various machine learning algorithms such as Random forest J48, Naïve Bayes, IBK and Multilayer perception on HTTP DDoS attack dataset. The dataset has a total number of 17512 instances which constituted normal (10256) and HTTP DDoS (7256) attack with 21 features. The implemented Performance evaluation revealed that Random Forest algorithm performed best with an accuracy of 99.94% and minimum false positive rate of 0.001%.

Keywords— ***DDoS , IDS , Machine Learning, Random Forest.***

I. INTRODUCTION

As the technology of internet infrastructure advances, so also do attacker's advance in their ways of attacking these network resources. This includes attacks on network availability, confidentiality, destination and integrity of the packets using various techniques such as Denial of Service, Structured Query Language injection (Sql) , Cross Site Scripting and Distributed denial of service (DDoS) attack [1].

Distributed Denial of Service (DDoS), is a comparatively simple, however very prevailing technique used to attack computers as well as internet resources. Several distributed agents devour the critical resources of the target within the shortest of time and deny legitimate clients from accessing services [2]. For the last two decades, DDoS have been one out of the greatest threats affecting the internet infrastructure. Curving DDoS attacks are a particularly challenging task. Literature have shown that signature-based detection techniques seems to be inefficient in mitigating DDoS attacks as this type of attacks can mask itself among legitimate traffic [3]. The primary goal of DOS attacks is to deny several services of the end users , temporarily. Overall, it usually overloads the system with undesired requests, thereby consuming network resources. As such, DOS acts as a large cover for all forms of attacks which consume system and network resources. There are several forms of DDoS attack which includes Structured query language DDoS (SIDDoS), Smurf, UDP flood and HTTP-DDoS.

Simply put, HTTP-DdoS is a form of attack which generates attack traffic that closely simulates legitimacy of a human user. Thereby it becomes very difficult for a victim to differentiate between normal and attack traffic. Because of this type of attacks, the server becomes unavailable to legitimate users. The main impact of application layer DDoS attacks are, unnecessary slow network performance (accessing document or sharing files), unavailability of a particular website, unable to access known website, dramatic increase in the number of spam emails received [4].

To overcome such problems, an intrusion detection system (IDS) comes into play. Intrusion Detection System (IDS) is the most commonly used mechanism in detecting various types of attacks. The IDS plays an important role in network security. It not only detects known attacks, but also many known and unknown attacks [5]. IDS are defined to preserve the confidentiality, integrity, and availability of network [6]. IDS could be software, hardware or a combination of both. It captures the packets under examination and then alert network manager by logging the intrusion event [7].

Over the years, studies have been done on either detection and prevention of HTTP-DDoS attack based on machine learning techniques but only few of the machine learning algorithms are considered which make it difficult to know the actual algorithm that perform better than other. Also, the relevant evaluation metrics are not employed . Most literature focus mainly on accuracy, recall and precision thereby neglecting the misclassification rate.

Sequel to the highlighted research gap, this study set out to evaluates twelve machine learning algorithms such as J48, Naïve Bayes, IBK, Kstar, SMO, Simple logistics, Multilayer perception. Decision Table, PART and Random forest for detecting HTTP-DDoS attacks. The dataset was composed of normal and HTTP-DDoS attack; the normal has 10,256

978-1-7281-5160-1/19/$31.00 ©2019 IEEE

instances and HTTP-DDoS has 7,256 instances making a total of 17,512 instances. This dataset was evaluated using different performance evaluation matrix on the twelve (12) algorithms in WEKA environment. The result of the evaluation showed that Random forest algorithms has a better performance in terms of TP Rate, FP Rate , Precision rate, Recall rate, F-Measure rate and detection accuracy.

The rest of the paper is structured into 5 sections; section I introduced the study and highlighted some weaknesses of other studies. In Section II, a brief discussion of related work using machine learning as it has been applied to intrusion detection in the past was presented. Section III describe the various machine learning used in this study. Section IV presented the result of the experiment and evaluations. Section V gives the conclusion that will help to strengthen future research.

II. RELEVANT LITERATURE ON HTTP-DDOS ATTACK DETECTION

Reference [5] studied four machine learning algorithms for botnet DdoS attack detection. The machine learning. algorithms used were support vector machine, ANN, NB, DT, and USML (K-means, X-means, etc.). The evaluation was carried out on UNBS-NB 15 and KDD99 datasets, while using Accuracy, False Alarm Rate (FAR), Sensitivity, Specificity, False positive rate (FPR), AUC and MCC as parameter measures. USML (unsupervised learning) was reported to be the best at classifying between Botnet and normal network traffic.

Another experiment was done using the benchmarking dataset in [8]. The bat algorithm was adopted in the work. First, feature metrics was defined to identify if the request stream behavior is of attack or normal. Secondly, the bat algorithm was customized to train and test. Even though the devised bat algorithm amplified detection accuracy, it had maximal process complexity. The experiment achieved an accuracy of 98.4% using the CAIDA dataset.

Reference [9] collected a new dataset that includes modern types of attack, which they claim has not been used in previous research. The dataset contains 27 features and five classes. A network simulator (NS2) was utilized in the work. Three machine learning algorithms (MLP, Random Forest, and Naïve Bayes) were applied on the collected dataset to classify the DDoS types of attack namely: Smurf, UDP-Flood, HTTP-DDoS and SIDDOS. The MLP algorithm achieved the highest accuracy rate with (98.63%).

A filtering tree, which works like a service was developed. The XML consumer request is converted into a tree form and uses a virtual Cloud defender to defend against these types of attacks. The Cloud defender basically consists of five steps: sensor filtering (check number of messages from a user), hop count filtering (number of nodes crossed from source to destination—this cannot be forged by the attacker), IP frequency divergence (the same range of IP addresses is suspect), puzzle (it sends a puzzle to a user: if it is not resolved, the packet is suspect) and double signature. The first four filters detect HTTP-DDoS attacks while the fifth filter detects XML-DDoS attacks [10].

Reference [11] proposed an approach for detecting HTTP based DDoS attacks. It entails a five-step filter tree approach of cloud defense. These steps include filtering of sensors and Hop Counts, diverging IP frequencies, Double signatures, and puzzle solving. The approach helped in determining anomalies with the various Hop Counts and treating the sources of such anomaly as attack source

The methodology of applying MADM in the cloud is proposed in [12]. The experiments were conducted using real private test bed. The result of the study has shown high performance of MADM in detecting the HTTP-flooding attacks in the cloud environment based on the confusing matrices and AUC results. And it has been concluded that MADM performance using 4 thresholds is higher as compared with using 3 thresholds with 86.77% detection accuracy.

A new dataset that includes modern types of attack, which were not been used in previous research was collected in [13]. The dataset contains 27 features and five classes. The collected data has been recorded for different types of attack that target the Application and network layers. Four machine learning algorithms (Naïve bayes, Decision Trees, MLP, and SVM) were applied on the collected dataset to classify the DDoS types of attack namely: Smurf, UDP-Flood, HTTP-DDoS and SIDDOS. The MLP algorithm achieved the highest accuracy rate with (98.91%). They recommend examining the different features for feature selection technique and include the more types of modern attacks in different OSI layers, such as the transport layer for future work.

A Cloud service queuing defender (CSQD) technique that aims at protecting the cloud from HTTP and XML forms of DDoS attacks was proposed. Using this approach, a server must be up before a request is processed which is uniquely prefixed with an ID [14].

Reference [15] proposes a system that effectively detects DDoS attacks using the clustering technique of data mining followed by classification. This method uses a Heuristics Clustering Algorithm (HCA) to cluster the available data and Naïve Bayes (NB) classification to classify the data and detect the attacks created in the system based on some network attributes of the data packet. They point out that clustering algorithm is based on unsupervised learning technique and is sometimes unable to detect some of the attack instances and few normal instances, therefore classification techniques are also used along with clustering to overcome this classification problem and to enhance the accuracy. They performed series of experiment using two types of dataset; The CAIDA UCSD DDoS Attack 2007 Dataset and DARPA 2000 The efficiency of the proposed system was tested based on the following accuracy, detection rate and False Positive Rate and the result obtained from the proposed system has been found that Naive Bayes Classification results in better in all the parameters.

Reference [16] has introduced the Heuristic clustering algorithm to cluster the data and detect DDoS attacks in DARPA 2000 datasets and has obtained better results in terms of detection rate and false positive rate in comparison to K-Means and K-Medoids algorithm. Sharmila and Roshan (2018) performed series of experiment using the CAIDA UCSD DDoS Attack2007 Dataset and DARPA 2000 and proposed a system that detects DDoS attacks using the

clustering technique followed by classification. Based on some network attributes of the data packet, Heuristics Clustering Algorithm (HCA) was adopted to cluster the available data and Naïve Bayes (NB) classification was also adopted to classify the data. Since clustering algorithm is based in unsupervised learning technique and is sometimes unable to detect some of the attack instances and few normal instances, therefore classification techniques are also used along with clustering to overcome this classification problem and to enhance the accuracy. The system's efficiency was tested using the following parameters; accuracy with 99.45% and false positive rate with 0.54%. Though the number of misclassifications need to be reduced.

More recent, [17] proposed a detection system of HTTP DDoS attacks in a Cloud environment. The system which is based on Information Theoretic Entropy and data learning algorithm consists of three main steps: entropy estimation, preprocessing, and classification. A time-based sliding window algorithm was used in estimating the entropy of the network heeder features of the incoming network traffic and then classify the data into normal and HTTP DDoS traffic. Performance metrics based on accuracy, FPR, Area Under Curve (AUC), and running time metrics were used for the evaluation of the proposed detection system achieving an accuracy rate of 99.54% with 0.4 FPR.

This paper [18] gave a comparative study of malware detection using fifteen different Machine Learning algorithms which include J45, LMT, Naïve Bayes, Random Forest, MLP Algorithm, Random Tree, REP Tree, Bagging, AdaBoost, KStar, SimpleLogistic, IBK, LWL, SVM, and RBF Network. The experiment was performed on ClaMP dataset on WEKA environment. Even though Random Forest algorithm outperforms the other algorithms, the accuracy rate is still low.

Reference [19] perform the execution of eight machine learning algorithms, namely, decision trees, random forest, artificial neural network, support vector machine, linear discriminant analysis, k-nearest neighbors, logistic regression and Naive Bayes to classify the ecological data. The performance evaluation in terms of recall, precision, accuracy and F-score reveals that LDA & NB classification algorithms are considered an accurate algorithms and outperformed many other supervised ML algorithms

Table I shows the summary of algorithms used in relevant literature while Table II depicts the different performance measures applied in previous studies.

TABLE I. SUMMARY OF RELATED ALGORITHMS COMPARED WITH LITERATURE

Reference	Year	Random forest	J48	Naive Bayes	IBK	kStar	SVM	Simple Logistics	MLP	Decision Table	Naive Bayes Simple	BayesNet	USML	HCA
[2]	2016	✓					✓							
[5]	2019			✓			✓			✓			✓	
[6]	2019	✓												
[9]	2016	✓		✓					✓					
[16]	2018			✓										✓
[17]	2018	✓												
[18]	2019	✓		✓		✓	✓	✓	✓					
[19]	2019	✓		✓			✓			✓				
[28]	2017				✓					✓		✓		

TABLE II. SUMMARY OF RELATED PERFOMANCE METRICS USED AND COMPARED WITH LITERATURE

Reference	Precision	Recall	Accuracy(%)	False Positive	True Positive	F-measure
[2]	0.99	0.99	99.00			
[5]			98.08	0.91		
[6]	0.99		96.00	0.002		
[9]	0.48	0.93	98.63			
[16]			99.45	0.54		
[17]			97.54	0.4		
[18]	0.99	0.99	99.20			
[19]	0.94	0.95	94.58			0.95
[28]			63.71	0.36		

III. MACHINE LEARNING ALGORITHMS

By machine-learning we mean algorithms that are first trained with reference input to "learn" its specifics (either supervised or unsupervised), to be later deployed on previously unknown input for the actual detection process. An overview of the machine learning algorithms used in this work has been described below:

The Naïve Bayes is a simple probabilistic algorithm [17]. It assumes that the effect of a variable values on a given class is independent of the values of other

variables which is referred to as class conditional independence.

In classification and regression, Support Vector Machines are the most common and popular method for machine learning tasks [20]. In this approach, a set of training examples are given with each example marked belonging to one of the two categories. Then, by using the Support Vector Machines algorithm, a model that can predict whether a new example falls into one category or other is built.

J48 algorithm was designed to enhance the implementation of the C.4.5 algorithm which is implemented by [20] in 1993. The expected outcome based on this algorithm is in the form of decision binary trees but with more grounding between computing time and accuracy [21]. As regard the decision tree structure, the leaf node had a decision of known output.

Multi-layer Perceptron (MLP) Algorithm is one of the most common functions algorithms that prove its effectiveness to deal with several application areas e.g. time series, classification and regression problems. [22] The testing phase can be implemented within short period of time. On the other hand, the training phase is typically implemented in a long period of time. MLP algorithm can be implemented with various transfer functions e.g. Sigmoid, Linear and Hyperbolic. The number of outputs or expected classes and number of hidden layers are important design considerations of the MLP algorithm implementations. At the beginning, every node within the neural network had its randomly weight and bias values, the large weight values present the most effective attributes within a dataset, and on the contrary, the small weight values present the lowest effective attributes within a dataset.

In study of [23], the K* algorithm can be defined as a method of cluster analysis which mainly aims at the partition of „n" observation into k" clusters in which each observation belongs to the cluster with the nearest mean. We can describe K* algorithm as an instance-based learner which uses entropy as a distance measure. The benefits are that it provides a consistent approach to handling of real valued attributes, symbolic attributes and missing values.

Reference [24] K* is a simple, instance-based algorithm, like K-Nearest Neighbor (K-NN). New data instances, x, are assigned to the class that occurs most frequently amongst the k-nearest data points, yj where j = 1, 2…k. Entropic distance is then used to retrieve the most similar instances from the data set. By means of entropic distance as a metric has several benefits including handling of real valued attributes and missing values.

A BayesNet learns Bayesian networks made in nominal attributes and any missing values are being replaced. Bayesian networks or Bayes Nets are graphical representation for probabilistic relationships among a set of random variables. Given a finite set X= {X1…Xn} of discrete random variables where each variable Xi may take values from a finite set represented by Val (Xi).

Reference [25] explains IBK as a algorithm that uses the same distance metric. The number of nearest neighbors can be specified explicitly in the object editor or determined automatically using leave-one-out cross-validation focus to an upper limit given by the specified value. IBK is a k-nearest neighbor algorithm. A kind of different search algorithms can be used to speed up the task of finding the nearest neighbors. The default is linear search, but further options involves KD-trees, ball trees, and the cover trees. The distance function used is a parameter of the search method. The remaining thing is the same as for IBL—that is, the Euclidean distance; other options include Chebyshev, Manhattan, and Makowski distances. [26] Predictions from more than one neighbor can be weighted according to their distance from the test instance and two different formulas are implemented for converting the distance into a weight. [25] [27].

The Simple Logistic Regression solves the classification issues. SLR works for both binary classification and multiclass classification. The chance of an event occurring is predicted by fitting data to the Logistic function. The values selected by the logistic function is between the range zero and one. If the value is 0.5 and above then it is labeled as 1, otherwise 0 [28].

Decision tables are one type of analysis method (or technique), that is commonly used by developers (for design and specification documents), and by others in software engineering and other disciplines –e.g., by test engineers, test analysts, test managers, etc. Decision tables are used mainly because of their visibility, clearness, coverage capabilities, low maintenance and automation fitness. Also, decision tables are vital in sourcing information for model based testing, and work well on rule based mechanisms or modeling techniques [29]. Decision tables are composed of rows and columns. Each of the columns explains the conditions and actions of the rules [30].

A. Dataset Description and Analysis

The dataset used for this study was obtained from [32]. The dataset comprises of four different DDoS attack types of which HTTP-DDoS attack is one of the attack types. The dataset contains 27 features and five classes. The five classes are a representation of the four attack types and normal. For the purpose of this study, 7256 (42%) instances of HTTP-DDoS attacks were extracted from the dataset and 10256(58%) of normal traffic were also extracted from the dataset. The details of the dataset's instances and features are presented in the Table III and Table IV. The dataset was imported to the Weka 3.8 to implement various machine learning algorithms. Fig.1. presents the main steps of the intrusion dataset import..

The performance evaluation was tested on windows 8 having the specification; Processor: Intel Pentium (R) Core ™ i7-5500U CPU @ 2.40GHz 2.30GHz, Installed Memory (RAM): 16.00 GB, System Type: 64-bit Operating System.

TABLE III. DATASET ATTRIBUTES

Class Type	*Number of Records*
Normal	*10256 packets*
HTTP-DDoS	*7256 packets*

TABLE IV. DATASET VARIABLES

Variable No	Features	Type
1	SRC ADD	Continuous
2	DES ADD	Continuous
3	PKT ID	Continuous
4	FROM NODE	Continuous
5	TO NODE	Continuous
6	PKT TYPE	Continuous
7	PKT SIZE	Continuous
8	FLAGS	Continuous
9	FID	Symbolic
10	SEQ NUMBER	Continuous
11	NUMBER OF PKT	Continuous
12	NUMBER OF BYTE	Continuous
13	NODE NAME FROM	Continuous
14	NODE NAME TO	Symbolic
15	PKT IN	Symbolic
16	PKTOUT	Continuous
17	PKTR	Continuous
18	PKT DELAY NODE	Continuous
19	PKTRATE	Continuous
20	BYTE RATE	Continuous
21	PKT AVG SIZE	Continuous
22	UTILIZATION	Continuous
23	PKT DELAY	Continuous
24	PKT SEND TIME	Continuous
25	PKT RESEVED TIME	Continuous
26	FIRST PKT SENT	Continuous
27	LAST PKT RESEVED	Continuous

Fig. 1. Experimental procedure flow

B. Performance Evaluationn

In this section, measure for assessing the accuracy of a algorithm is presented.To evaluate the performance of these algorithms, parameters that are often called metrics such as True Positive Rate(TPR), False Positive Rate (FPR), Precision Rate (PR), Recall (RR) F- Measure and accuracy are used in line with the parameter used in [3][5][9][32].

Below are mathematical representations of the performance metrics used.

- **True Positive**: - Define attacks correctly detected as attack.

$$TP = \frac{TP}{TP - FN} \quad (1)$$

- **False Positive: -** true and false signifies the expectation of the algorithm, while positive and negative represents the prediction of the algorithm..

$$FN = \frac{FN}{FN - FP} \quad \textbf{(2)}$$

- **Precision** this gives the class agreement of the packet labels with the positive labels by the algorithm.

$$p = \frac{TP}{TP - FP} \quad (3)$$

- **Recall** shows the effectiveness of a algorithm to identify the positive labels.

$$r = \frac{TP}{TP - FP} \quad (4)$$

- **F-measure** this gives the relation between data's positive labels and the one given by a algorithm.

$$f\text{measure} = \frac{2pr}{p + r} \quad (5)$$

- **Detection accuracy**: -this shows the numbers attacks that are detected correctly, expressed in %.

$$Accuracy = FN = \frac{TP - FN}{TP - FN - FP - FN} = p = \frac{TP - FN}{N} \quad (6)$$

IV. RESULTS AND DISCUSSION

In order to obtain a better model for detection of DDoS attack, 12 machine learning algorithms namely J48, Random forest, Naive Bayes, IBK, Kstar, SMO, simpleLogistics, Multilayer perception, Decision Table, PART, NaivebayesSimple, BayesNet were evaluated. To effectively determine the performance of the machine algorithms, 17512 records were extracted, each of the algorithm was trained on the dataset using 70% of the collected data and the 30% were used as a test data. The result of the experiment is represented in a table as well as in graph. Fig. 3 shows the false positive rate for the above-mentioned machine learning algorithms with estimated probabilities values ranging from 0 to 0.06. The random forest has the lowest FP rate with 0.001 while Naïve Bayes has the highest with 0.056. From the graph shown in Fig. 3 and Table .V, it can be concluded that the Random Forest algorithm outperforms the other

methods in identifying the network traffic as normal or an attack. Whereas the Kstar identifies the intrusion with the lowest probability estimate. Random Forest has highest TPR. By observing the graphs, it can be concluded that the Random forest algorithm has lowest FPR and highest accuracy in detecting attacks when compared with others. Whereas NaiveBayes has highest FPR and lowest accuracy for intrusion detection.

TABLE V. RESULT COMPARISON OF TWELVE MACHINE LEARNING ALGORITHM

Algorithms	TP Rate	FP Rate	Precision	Recall	F-Measure	Accuracy
Random Forest	**0.999**	**0.001**	**0.999**	**0.999**	**0.999**	**99.9371**
J48	0.994	0.006	0.994	0.994	0.994	99.3713
Naivebayes	0.935	0.056	0.942	0.935	0.935	93.524
IBK	0.999	0.001	0.999	0.999	0.999	99.9057
Kstar	0.991	0.008	0.991	0.991	0.991	99.0883
SMO	0.984	0.015	0.984	0.984	0.984	98.3967
simpleLogistics	0.994	0.006	0.994	0.994	0.994	99.4027
Multilayeperseption	0.995	0.005	0.995	0.995	0.995	99.497
Decision Table	0.995	0.005	0.995	0.995	0.995	99.5285
PART	0.997	0.003	0.997	0.997	0.997	99.7485
NaivebayesSimple	0.946	0.045	0.952	0.946	0.946	94.6226
BayesNet	0.995	0.005	0.995	0.995	0.995	99.4656

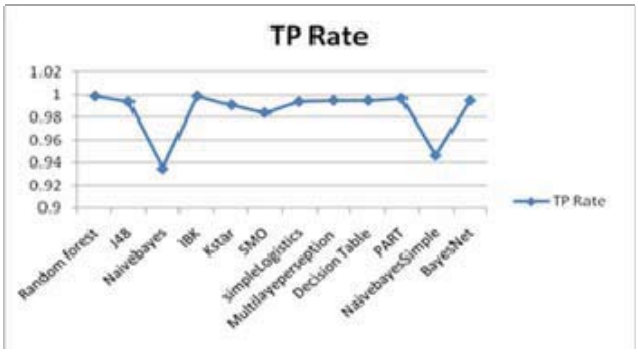

Fig. 2. Algorithms true positive rate

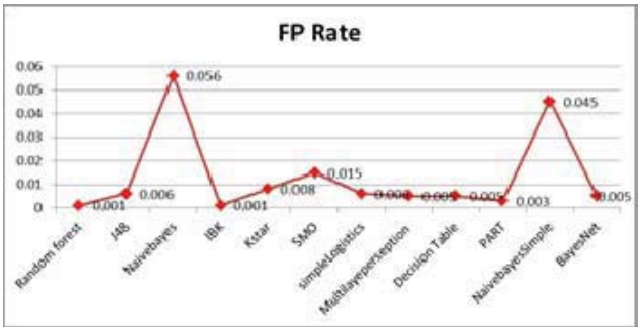

Fig. 3. Algorithms false positive rate

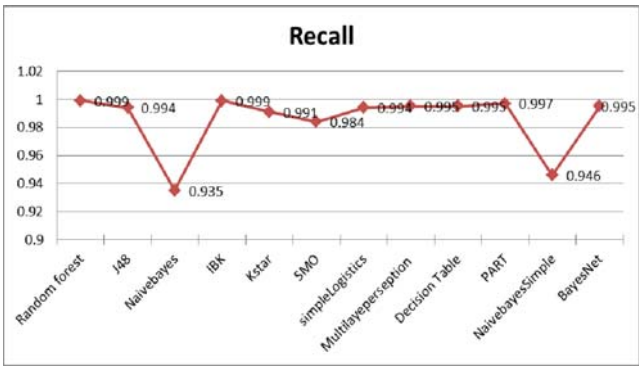

Fig. 4. Algorithms true positive rate

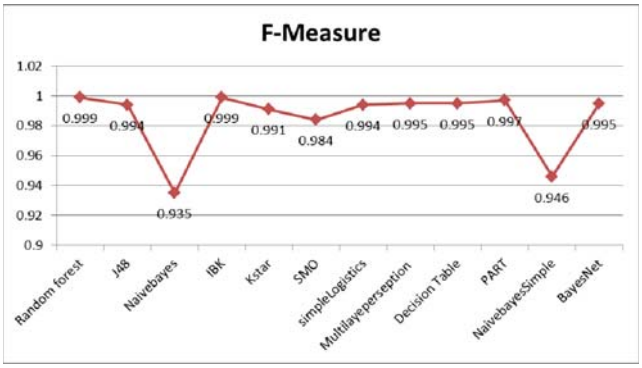

Fig. 5. Algorithms f-measure

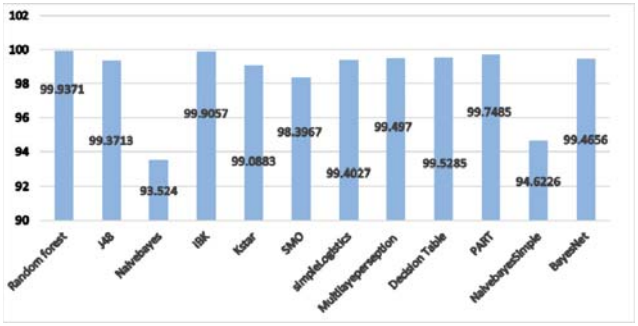

Fig. 6. Algorithms accuracy

V. CONCLUSION

This paper studied and compared different machine learning algorithms. Several experiments was performed and tested to evaluate the efficiency and the performance of the following machine learning algorithms: J48, Random forest, Naive Bayes, IBK, Kstar, SMO, simpleLogistics, Multilayeperception, Decision Table, PART, NaivebayesSimple, Bayes Net. All the tests were based on the intrusion detection dataset. The rate of the different type of attacks in the intrusion detection dataset is approximately 41% of HTTP attacks, and 59% of normal packets. The analysis of the algorithms was then performed using the set metrics, the efficiency and usability of the Random Forest for the detection of HTTP-DDoS attack performs comparatively better as

compared to the other algorithms. Also, the comparison with other relevant literatures depicts the higher performance of random forest in terms of accuracy and lower false positive rate.

We recommend that more publicly available HTTP-DDoS dataset should be used in evaluating the performance of other machine learning algorithms as well as using other relevant performance metrics.

REFERENCES

[1] Z., He, T., Zhang and Ruby B. L. Machine Learning Based DDoS Attack Detection From Source Side in Cloud. 2016. Retrived form http://palms.ee.princeton.edu/system/files/Machine_Learning_Based_DDoS_Attack_Detection_From_Source_Side_in_Cloud_camera_ready.pdfand accesed on 20th januray 2017

[2] M. C. Belavagi and B. Muniyal. "Perfomance evaluation of supervised machine learning algorithms for intrusion detection." Twelveth international multi conference on information processing, Manipal 576, 104, India, 2016.

[3] Irfan S., Amit M., Vibhakar M., Machine Learning Techniques used for the Detection and Analysis of Modern Types of DDoS Attacks "International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056 Volume: 04 Issue: 06 | June -2017

[4] S., Indraneel, V., Praveen and K. Vuppala HTTP flood attack detection in application layer using machine learning metrics and bio inspired bat algorithm 2017 https://www.sciencedirect.com/science/article/pii/S221083271730165

[5] T. A. Tuan, H. V. Long, L. H. Son, I. Priyadarshini, N. T. Son."Perfomance evaluation of botnet DdoS attack using machine learning." Evolutionary Intelligence, DOI: 10.1007/s12065-019-00310-w, 2019.

[6] I.S. Jacobs and C.P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.

[7] K. Elissa, "Title of paper if known," unpublished.

[8] I. Sreeram and V.P.K Vuppala, "HTTP flood attack detection in application layer using machine learning mmetrics and bio inspired bat algorithm," 2017.

[9] M.Alkasassbeh, G.Al-Naymar, A.B.A Hassanat and M.Almeidin, "Detecting istributed denial of service attacks using data mining techniques," in International journal of advanced computer science and application, Vol. 7, No.1, 2016.

[10] T. Kanwal, T. Sivakumar and G. Aghila, "A comber approach to protect cloud computing against XML DDoS an HTTP DDoS attack," in proceedings of the IEEE students; conference, Bhopal, Inia, 2012.

[11] P. Ankita and K. Fenil, "Survey on DDoS attack detections andd prevention in cloud," in international journal of engineering technology, mangement and applied science, 2015.

[12] A. Aborujilah and S. Musa, "Cloud-Based DDoS HTTP Attack Detection Using Covariance Matrix Approach," 2015.

[13] S.Irfan ., M. Amit and M. Vibhakar, "Machine Learning Techniques used for the Detection and Analysis of Modern Types of DDoS Attacks," in International Research Journal of Engineering and Technology, 2017.

[14] R.M. Sahardi and G. Vahid, "New Approach to Mitigate XML-DOS and HTTP-DOS Attacks for Cloud Computing. International Journal of Computer Applications, 72, 27-31. https://doi.org/10.5120/12579-9201

[15] S. Bista and R. Chitrakar, "DDoS Attack Detection Using Heuristics Clustering Algorithm and Naïve Bayes Classification", In Journal of Information Security, 2018, 9, 33-44http://www.scirp.org/journal/jis.

[16] B. Sharmila and C. Roshan . "DDoS attack detection using heuristics clustering algorith and naïve bayes classification." Journal of Information Security, 9(1), 33-44, 2018.

[17] I. Mohamed , A. Karim, and B. Mustapha." Detection system of HTTP DDoS attacks in a cloud environment based on information theoretic entropy and random Forest." *Security and Communication Networks,* 2018(2), 1-13.

[18] E. G. Dada, J. S. Bassi, Y. J. Hurcha and A. H. Alkali. "Performance evaluation of machine learning algorithms for detection and prevention of malware attacks detection." *Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 21, Issue 3, Ser. I (May -June 2019), PP 18-*27www.iosrjournals.org

[19] S. U. Doulah and A. Alam. "Performance evaluation of machine learning algorithm in ecological dataset." *IJAMML 10:1 (2019) 15-45*

[20] K. Sarmila and G. Kavin, "A Clustering Algorithm for Detecting DDoS Attacks in Networks," in International Journal of Recent Engineering Science, 1, ISSN: 2349-7157, 2014.

[21] T. Mitchell," Machine Learning," McGraw Hill, New York, 1997.

[22] V. Vapnik, "The Nature of Statitical Learning Theory," Springer, Heidelberg, 1995.

[23] J.R. Quilan, "C4.5: Programs for machine learning," in Elsevier, 2014.

[24] M.S.Bhuller and A. Kair, "Use of data mining in educstion sector," in proceedings of the world congresson engineering and computer science, vol. 1, 2012.

[25] S.K. Pal andS. Mitra, "Multilayer perception, fuzzy sets and classification," in IEEE transactions on neural networks, vol. 3, 1992.

[26] S. Vijayarani and M. Muthulakshmi, "Comparative Analysis of Bayes and Lazy Classification Algorithms," in International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 8, August 2013 Copyright to IJARCCE www.ijarcce.com 3118

[27] T. C. Sharma and M. Jain, "WEKA Approach for Comparative Study of Classification Algorithm"

[28] K. H. Raviya and B. Gajjar, "Performance Evaluation of Different Data Mining Classification Algorithm Using WEKA"

[29] S. Ghosh, S. Roy and S. K. Bandyopadhyay, "A tutorial review on Text Mining Algorithms"..

[30] S.Vijayarani and S.Sudha,"Comparative Analysis of Classification Function Techniques for Heart Disease Prediction"

[31] M. C. Belavagi and B. Muniyal, "Performance Evaluation of Supervised Machine Learning Algorithms for Intrusion Detection," in Twelfth International Multi-Conference on Information Processing-2016 (IMCIP-2016)

[32] A. Mouhammd, A. Ghazi, B.A. Ahmad, and M.A. Hassanat. "Detecting Distributed Denial of Service Attacks Using Data Mining Techniques." *In International Journal of Advanced Computer Science and Applications, 7, 1, 2016.*