https://doi.org/10.46335/IJIES.2025.10.4.17 **e-ISSN: 2456-3463**
Vol. 10, No. 4, 2025, PP. 83-91

International Journal of Innovations in Engineering and Science, www.ijies.net

Detection of DDoS Attacks in IoT Networks Using KNN, SVM, and Random Forest Classifiers

Ranjan Kumar Gupta[1], Dr. Ranu Pandey[2]

[1]*Research Scholar, Department of Computer Science, Shri Rawatpura Sarkar University, Raipur, Chhattisgarh, India*
[1]*kumarranjange@gmail.com*
[2]*Assistant Professor, Department of Computer Science, Shri Rawatpura Sarkar University, Raipur Chhattisgarh, India*
[2]*ranu_pandey8@hotmail.com*

Received on: *19 April, 2025* ***Revised on:*** *3 June,2025* ***Published on:*** *06June ,2025*

Abstract – *The rising cases of Distributed Denial of Service (DDoS) attacks have posed major concerns to the cybersecurity especially in the context of Internet of Things (IoT). The conventional way of detecting these attacks is failing to respond to the changing tactics of these attacks. The study discusses the use of machine learning models of classifying DDoS attacks, namely: K-Nearest Neighbors (KNN), Support Vector Machines (SVM), and Random Forest to identify the IoT data. The experiment determines the performance of these classifiers in classifying DDoS as attacks and non-attacks of the publicly available dataset. Some different performance measures, where we estimate and compare, we used can include accuracy, precision, recall, F1-score, and the Matthews Correlation Coefficient (MCC) to estimate the performance of various classifiers in finding out what was detected. The findings of this study offer a good source of information about the merits and drawbacks of each of the classifier and their subsequent use to create more productive and efficient systems of detecting DDoS attacks within the IoT environment. These results provide a reference to the subsequent choice of machine learning methods to enhance the precision and certainty of DDoS anticipation in the practice.*

Keywords- *DDoS Attack, Internet of Things (IoT), K-Nearest Neighbors (KNN), Support Vector Machines (SVM), and Random Forest.*

INTRODUCTION

IoT devices have grown at an unparalleled rate resulting in augmenting traffic on the network and the complexity of handling those devices. The increase has come with the unfortunate development of new cyber security challenges, especially in the form of Distributed Denial of Service (DDoS) attacks [1]. DDoS attacks have the purpose of overloading a target system or network with traffic and making them available to legitimate users. These attacks may lead to large losses in funds, service denial and long-standing damage to reputation. DDoS attacks are changing and getting sophisticated and rule-based mechanisms of detecting them are becoming ineffective. Due to these developments, it is becoming immediately necessary to look into more adaptable and automated modes of addressing the issue that can recognize new patterns of attack in real-time.

This study aims to discuss how machine learning (ML) may be used in order to enhance the detection of a DDoS attack in an IoT environment, especially with K-Nearest Neighbors (KNN), Support Vector Machines (SVM) and Random Forest, among others. The advantages of ML algorithms are evident in that it can automatically learn information through data and adapt to new patterns of attack and fewer manual rules are necessary. With the help of such advanced methods, we will pursue to provide the field with better-developed DDoS detectors that could not only anticipate such pattern of attacks but also scale accordingly with a risk increase on the IoT networks. Also, through the comparison of several classifiers, we also aspire to determine the best method that suits various attack scenarios.

Contributions of the study include the following:

- *Application of Machine Learning Classifiers:* We examine application of three of the most salient machine learning approaches, KNN, SVM, and Random Forest, to IoT data to be

https://doi.org/10.46335/IJIES.2025.10.4.17 e-ISSN: 2456-3463
Vol. 10, No. 4, 2025, PP. 83-91

used to classify DDoS attacks. All these classifiers possess various strengths that may make them fit in different forms of attacks and data features.

- *Performance and Comparison with Other Metrics:* The performance of the classifiers can be tested on numerous metrics using a publicly available dataset with accuracy, error rate, sensitivity, specificity, precision, F1-score, and Matthews Correlation Coefficient (MCC) just to mention a few. This comparison will enable the identification of the strengths and weaknesses of each of the classifiers as regards DDoS detection.
- *Beyond of DDoS Detection in IoT Environments:* We bring new knowledge to the reader regarding the risks and limitations of the application of machine learning to detect DDoS attacks. These are such issues as imbalance of data, feature selection, false positives/negatives and scalability issues.
- *Future System Recommendations:* The results will aid in finding the most appropriate classifiers under different DDoS attack settings, and will give guidance on how machine learning-based, detection systems might be optimized in a real-world IoT setting.

Having completed this chapter, the readers will be more informed on the ways that machine learning methods can be utilized to accurately identify DDoS attacks and the way in which a suitable model can be chosen to appropriately address the problem at hand. The conducted research is expected to be used as a basis to create the more adaptive, efficient, and accurate DDoS detection systems in an IoT-based environment.

LITERATURE REVIEW

Solutions proposed in [2] rely on Hybrid AES-ECC approach of protecting IO data with a particular emphasis on the effectiveness of crypto algorithm. In their work, they write the comparison of the AES and ECC as applied in IoT environment and they mention that the ECC would perform better and offer reduced key sizes when compared to RSA. Their model is energy efficient and secure to use AES encryption and ECC key exchange, which is an aspect that is metered by their proposed model. The AES-ECC hybrid model will consume less computational load than the conventional encryption method and therefore, most apt on the IoT tools with resource constraints. In the given paper, the amount of power consumed by the hybrid cryptosystem is also analyzed, and it is proved that this cryptosystem can be applied in real-time IoT environments where the aspects of energy efficiency and reduced energy consumption are the most significant.

To overcome security issues associated with the fact that medical data is stored, the authors of [2] offer a mixed strategy based on blockchain technology-based security and decentralized learning, in particular, neural networks. As it is noteworthy, this approach only shares the parameters of the classifier and does not disclose sensitive information. The listed security measures notwithstanding, described results of this approach are below the level of satisfaction, which indicates the need to continue further improvement and streamlining.

The paper [4] offered a solution in two approaches, which involve the usage of Dempster-Shafer Theory (DST) steps and fault tree analysis (FTA) implemented in the cloud network on the virtual machine (VM) detection system (IDS) attacks to identify and analyze the cloud network during DDoS attacks in cloud computing facilities. The quantitative aspect of the uncertainty element is solution which is largely beneficial in the IDSs to reduce the false alarm rate.

New norms are forming in the processes of communicating the IoT based sensor networks. However, the availability of resources is threatened by the DDoS attacks because sensor nodes are exposed to grotesque attacks in the network. To prevent it, the authors of [6] provided an IoT sensors DDoS detection scheme based on ADE (Averaged Dependence Estimator).

In the discussion of [6] of the anomalies on the network system, the authors indicated that this would be because of defects or in the case of an attack network. It was differentiated by using the ML based method of classification. As per the findings, supervised machine learning algorithmic methods possess quite high levels of accurateness in regards to classifying a failure or an attack.

The subject of predicting the network traffic forms an excellent means of detection and capturing network anomalies within a communications network. The accuracy of the predictions in the employed artificial neural network (NARX) allowed making 62 steps of prediction with 5% accuracy. Such system will help in identifying fake traffic within the network. The results

https://doi.org/10.46335/IJIES.2025.10.4.17 **e-ISSN: 2456-3463**
Vol. 10, No. 4, 2025, PP. 83-91

suggest the network traffic analyzer has turned out to be a feasible device to intercept the DDoS attacks [7].

In [8], the authors designed an intrusion detection system based on machine learning algorithms to detect and prevent the attacks experienced by mobile distributors of energy sources and the dynamic wireless charging that is dynamic. The 91% accuracy of detecting the intrusion was implemented on the use of the proposed layered structure (IDS) and KNN or RF algorithms.

The authors of [9] suggested to use the CNN and RNN data learning model in the task of intrusion detection and offered PL-CNN, PL-RNN methods. Web Shell and HTTP datasets that were used in the study include CNTC-2017, Darpa-1998 and CSIC-2010. Such techniques are demonstrated with respect to representation of features of the first network packets. Therefore, the models do not rely on feature engineering and familiarity with the domain that involves network security.

Distributed intrusion detection systems could offer the capability of an intrusion detection system that has the capacity to learn and collect information however; this is easily susceptible to insider attack. This has led to formation of an automation of the process of establishing the intrusion sensitivity values based on expert knowledge based supervised approach to machine learning. The authors [10] utilized KNN, BPNN and DT models in their works.

In [11], the machine learning based detection system was developed and it operates on the premise of four features proposed by the authors to detect the strategies of GET Flood attacks and the system differentiates between fake users (bot) and genuine users. The classifiers that the machine running with Worldcup98 possess include NASA, Clarknet data, Naive Bayes, Random forest, SVM etc.

An OCSA and RNN based intrusion identification system to cloud computing services has been proposed by authors of [12]. It has come up with a metaheuristic OCSA algorithm which can also be used as it is, to select the personalities based on the basis of an OBL and CSA algorithms. The precision rate of the model constructed by RNN was 94.12% to categorize KDDcup99 dataset into specific categories.

Using the validity of applying the concepts of artificial neural networks to the objective of establishing network attacks, the authors of [13] in the paper classified the data set that contained 7 possible classes using the model of LSTM. It was observed that the LSTM model in the 2-class structure that was selected among the data set is a more successful model than MLP model in network attacks classification.

In their attempt to identify network attacks, the authors of [14] have examined the hybrid networks. It was a KDDcup99 and NSL-KDD based work. The discussed classification model utilized ANN-ID, ANN-NFC and ANN-SVM combination. The data of the different classifier would then be fed into a novel classifier and hybrids would be created.

PROPOSED METHODOLOGY

Figure 1 illustrates a complete procedure of detection of Distributed Denial of Service (DDoS) employing machine learning methods, in particular, SVM, KNN and Random Forest, classifier models on Internet of Things (IoT) data.

They begin with the collection of IoT data that is the major source of the input to the machine learning models. After gathering this data, it goes through a process known as pre-processing where data is cleaned up and normalized. The necessity of accomplishing this step lies in the fact that it is needed to remove any noise in the data and standardizes values so that the data is of high quality when entering the models for analysis.

After pre-processing, data is classified into two sets namely training and testing. The DDoS data is split into a training set that can be used to learn the models via the machine learning process and ends up learning the patterns and characteristics used to distinguish DDoS attacks and non-attacks. As soon as the models are trained, their functioning is estimated using the testing set. This stage of evaluation is carried out with the help of different measures such as accuracy, precision, sensitivity, and F1-score to determine the efficiency of the models in relation to detection of DDoS attacks.

The second process will involve classification, in which the trained models can be applied to classify incoming new IoT data and detect a possible DDoS attack. Lastly, the performance of the SVM, KNN and Random Forest classifier based on important performance metrics is examined based upon the outcome of the classification. This will assist in coming up with the best model of DDoS attack detection in IoT context.

https://doi.org/10.46335/IJIES.2025.10.4.17 e-ISSN: 2456-3463
Vol. 10, No. 4, 2025, PP. 83-91

As it can be summarized, this flowchart suggests a machine learning pipeline of DDoS attack detection where the focus is on several important steps, i.e., data pre-processing, training of a model, evaluation, and classification, and the roles of performance metrics when judging the effectiveness of a classifier.

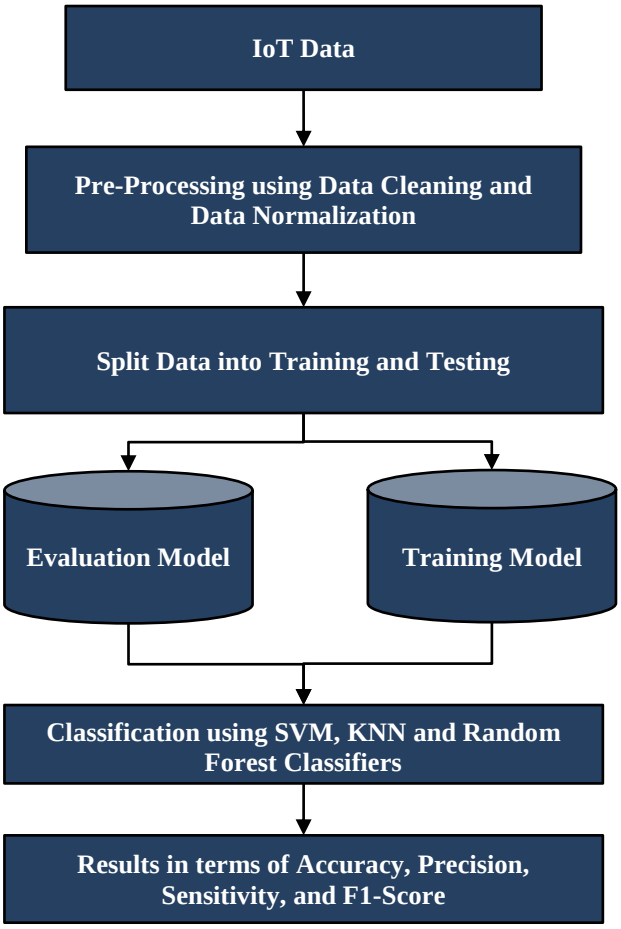

Fig. 1- Flow diagram for proposed approach

3.1 Pre-Processing

Predicting the occurrence of DDoS attacks relies on data mining that starts with a necessary pre-processing procedure. There are forms of challenges tackled at this stage which include irregularities, errors, and even missing values in the raw data. Before the mining is commenced, pre-processing is done to have a clean and viable data suitable to the mining process. Knowledge Discovery in Databases (KDD) is among the remarkable approaches used in this stage and it is vital in prepping up databases since it translates crude information into an analytical state. Correct pre-processing does not only increase the quality of the data but also makes further predictions and classifications more accurate and reliable.

3.2 Support Vector Machine (SVM)

SVM is one of the common machine learning algorithms in the classification of various tasks and these algorithms proved to be effective in DDoS attacks detection. In this respect, the operation of SVM differentiates. Normal network loads and malicious traffic which is typical of DDoS attacks. It does so by training itself to a decision boundary, or a hyperplane, that splits the two classes of data--normal and attack traffic.

The main idea on which SVM is based on is to maximize the distance between the decision boundary and the nearest data points of each class called support vectors. This would make the classifier as robust as it can be in order to classify new and unseen data. In the effort to identify the ideal margin that would allow them the best hyperplane to classify the normal and attack traffic, SVM strives to maximize this margin to avoid the chances of misclassification. This would be especially true in using SVM to recognize the complex patterns in the DDoS attack traffic since the approach is capable of processing non-linear relations and high dimensional data.

3.3 K-Nearest Neighbors (KNN)

One of the simplest, known at the same time as one of the most effective machine learning methods, is the K-Nearest Neighbors (KNN) algorithm. It is founded on a notion called lazy learning, whereby, there is no explicit training process or it is very small, and this makes it possible to have a quick process of training. KNN is termed to be a compute intensive algorithm as compared to other computational theories since the model building process can take a long time versus other algorithms where the data points are being used in predictions thus, the training is nearly instant.

When using KNN, the data lie in a feature space which may consist of either scalars, or of multidimensional vectors, as appropriate to the data. It applies the algorithm either in regression or classification, and in both cases, it can err by deciding the output of the computation on the closeness of the points within the space of a feature.

In order to label a new datum point, the KNN algorithm searches out the k nearest data points in the feature space to that data point. When these neighbors have been detected, the algorithm classifies the new point to another common category of these neighbors. Another parameter of importance, which is the number of

https://doi.org/10.46335/IJIES.2025.10.4.17 e-ISSN: 2456-3463
Vol. 10, No. 4, 2025, PP. 83-91

neighbors to be taken into consideration, is the value of k. The algorithm involves choosing the majority of classes among the nearest neighbors thus guaranteeing that the new data point belongs to the majorities of the trends seen in the training data.

Such ease and common sense-like nature are part of what makes KNN so useful in mechanisms where the boundary of the classification is irregular and non-linear, since it naturally adapts to varied forms of data groupings. Nevertheless, KNN may also be costly on computation as the dataset increases in size and the feature space involved is large, which needs effective computation of distances.

3.4 Random Forest Classifier

Random forests constitute an effective method of ensemble learning made of a set of binary trees. As the name implies, the model is constructed out of multiple individual decision trees, though each decision tree is only trained on a random sample of the data. This randomness is important in effectiveness of random forests hence it facilitates robustness and generalization of the model. There is no sameness in the trees of a random forest since each is trained on a randomly selected subset of the data thereby causing diversity in the model.

Training of these trees is founded on the concept of Bagging (Bootstrap Aggregating), which entails training of several models individually on distinct samples of the data, after which the outcomes are combined. However, random forests do one step further, in that, they bring in a decorrelation method and this minimizes the correlation among the trees in the forest. It does this by using a random sample of features at each decision tree node in deciding the best split and not the overall features. By adding such randomness to the feature selection, the trees will have less dependence upon each other, thus there is less risk of them overfitting and contributing to enabling better generalization of the model to new data.

The basic object of the random forests is to reduce the correlation in trees that do not cause much variances. Such trade-off between bias and variance will make the model very effective using both training and testing data to classify and regress data sets. Random forests can especially be used when the data has numerous features, when there is a complex relationship, as they can process highly dimensional data (including identifying complex relationships).

3.4.1 Principle and Algorithm of Random Forests

Assume that input and output data of the IoT use m instances represented as $S = \{x_i | y_i \dots \dots \dots . x_m, y_m\}$. Also, suppose S is a bootstrap sample of instances generated in the following way: S_t is a bootstrap sample of size m obtained by resampling S with replacement. Assume that a set of T decision trees, H_t is a set of T decision trees constructed out of S_t, that is denoted by h. In order to generate each of the nodes of the tree, it picks out an attribute to partition by randomly choosing a subset of the attributes. The last step is classifying a new instance with a random forest classifier comprising of a majority voting taking into account a uniform weighting of the classifiers in set h. The algorithm shows this principle.

Pseudocode:

randomforest (S,T)

Entrance: $S = \{x_i | y_i \dots \dots \dots . x_m, y_m\}$, *the training set.*

Input: T the number of random forest decision trees.

For $t = 1, \dots \dots T$

1. Generate Bootstrap size S_t *sample* m *from S.*

2. By reiterating the process, create a decision tree.

Recursively, do the following actions for each node of the tree:

a) Randomly pick attributes among the attributes

b) Select the partitioning attribute among the

c) Divide the node into two child nodes

End for

Output: the RF classifier

SIMULATION AND RESULTS

4.1 Evaluation Parameters

The simulations are accomplished through the use of MATLAB 2024a. Table 1 below indicates the evaluation parameters of this research work.

Table 1- Evaluation parameters

https://doi.org/10.46335/IJIES.2025.10.4.17 **e-ISSN: 2456-3463**
Vol. 10, No. 4, 2025, PP. 83-91

International Journal of Innovations in Engineering and Science, www.ijies.net

TP (True Positive)	"Indicated the number of DDoS attacks that were classified as correctly classified"
TN (True Negative)	"Indicated the number of DDoS attacks that were classified as not classified correctly"
FP (False Positive)	"Indicated the number of DDoS attacks that were classified as incorrectly classified"
FN (False Negative)	"Indicated the number of DDoS attacks that were classified as not classified incorrectly"

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Sensitivity = \frac{TP}{TP+FN} \quad (3)$$

$$Specificity = \frac{TN}{TN+FN} \quad (4)$$

$$ErrorRate = \frac{FP+FN}{TP+TN+FP+FN} \quad (5)$$

$$False\ Positive\ Rate\ (FPR) = \frac{FP}{FP+TN} \quad (6)$$

$$F-Score = \frac{2TP}{2TP+FP+FN} \quad (7)$$

$$Matthews\ Correlation\ Coefficient\ (MCC) = \frac{(TP\times TN)-(FP\times FN)}{\sqrt{(TP+FN)(TP+FP)(TN+FN)(TN+FP)}} \quad (8)$$

$$Kappa\ Statistics = \frac{2(TP\times TN-FN\times FP)}{(TP+FP)\times(FP+TN)+(TN+FN)\times(FN+TN)} \quad (9)$$

4.2 Results

Table 2- Performance evaluation of IoT data under learning rate of 60 % and testing of 40 % different classifiers

Parameters	**SVM**	**KNN**	**Random Forest Classifier**
Accuracy	0.9800	0.9990	0.9960
Error	0.0200	0.0016	0.0040
Sensitivity	0.9818	0.9713	0.9891
Specificity	0.9951	0.9989	0.9992
Precision	0.9800	0.9713	0.8927
False Positive Rate	0.0049	0.0011	8.0345e-04
F1-score	0.9799	0.9713	0.9321
Matthews Correlation Coefficient	0.9757	0.9703	0.9356

The performance of the IoT data with an evaluation of 60% learning rate as well as 40% testing rate on the various classifiers (SVM, KNN, and Random Forest Classifier) indicates a great promise on detecting DDoS attacks. The three classifiers were highly accurate with KNN having the highest accuracy of 99.90, Random Forest being higher with 99.60% and SVM with 98.00 which suggests that they can rightfully classify a majority of the instances provided in the dataset. All the classifiers had low error rates, 0.0200, 0.0016, and 0.0040 of SVM, KNN, and Random Forest respectively, which indicates that there were few wrong predictions. The Random Forest Classifier had the highest score on sensitivity which measures the true positive rate of 98.91 % with SVM and KNN having scores of 98.18% and 97.13% respectively. Presenting the results in terms of specificity, that is, the true negative rate, Random Forest came first once more with 99.92, followed closely by KNN with 99.89, and SVM with 99.51. Discussing the precision, both SVM and KNN demonstrated the same precision level of 97.13, and Random Forest demonstrated the lower precision level of 89.27. The lowest of them all was the Random Forest at 0.0803 followed by KNN at 0.11 and SVM at 0.49. Precision and sensitivity yielded a F1-score where Random Forest was on top (93.21%), followed by KNN (97.13%) and SVM (97.99%). Finally, the Matthews Correlation Coefficient calculated as the balanced performance accuracy indicators ranked Random Forest as a top-scoring model with 93.56%, followed by KNN, 97.03%, and SVM, which showed 97.57% accuracy. Comprehensively, all the classifiers performed satisfactorily; nevertheless, Random Forest showed better results compared to others, especially in sensitivity, specificity, F1-score, and Matthews Correlation Coefficient, which depicts that it is superior in classifying DDoS attacks.

Table 3- Performance evaluation of IoT data under learning rate of 50 % and testing of 50 % different classifiers

Parameters	**SVM**	**KNN**	**Random Forest Classifier**
Accuracy	0.975	0.9667	0.9853
Error	0.025	0.0333	0.0147
Sensitivity	0.975	0.9667	0.9853
Specificity	0.9917	0.9889	0.9951
Precision	0.9753	0.9682	0.9861
False Positive Rate	0.0083	0.0111	0.0049
F1-score	0.9749	0.9669	0.9853
Matthews Correlation Coefficient	0.9668	0.9562	0.9808

The performance measure of the IoT data at a learning rate of 50% and a testing rate of 50%, and with the various classifiers (SVM, KNN and Random Forest Classifier) indicates that all the three classifiers displayed similar good performance in classifying the

https://doi.org/10.46335/IJIES.2025.10.4.17 **e-ISSN: 2456-3463**
Vol. 10, No. 4, 2025, PP. 83-91

International Journal of Innovations in Engineering and Science, www.ijies.net

DDoS attacks. As to the accuracy, Random Forest Classifier performed the best, having 98.53%, SVM - 97.50%, KNN - 96.67%, which means that the respective classifiers were capable of classifying instances correctly at a very high rate. There were slight differences in the number of errors with SVM registering the lowest of 0.025 and KNN the highest of 0.0333 and Random Forest Classifier 0.0147. Sensitivity, which is a gauge of the true positive rate was equal in all classifiers which was SVM- 97.50%, KNN- 96.67% and the Random Forest Classifier- 98.53%. On specificity, Random Forest was dominant as usual with 99.51% followed by SVM with 99.17% and KNN with 98.89% showing that it is able to distinguish negative cases accurately with high probability. The highest precision in the proportion of the correctly predicted positive instances occurred with Random Forest, as 98.61%, followed by SVM, 97,53%, and KNN, 96.82%. False positive rate was the lowest in Random Forest (0.0049), SVM (0.0083), and KNN (0.0111), respectively, which means that Random Forest helps in minimizing false positives, referring to positive instances, whereas the others fail to do so effectively. When it comes to the measure of F1-score which is the composition of precision and sensitivity, Random Forest retains the most excellent at 0.9853, then SVM and KNN at 0 0.9749 and 0.9669 respectively. Lastly, the balanced measure of performance called the Matthews Correlation Coefficient indicated Random Forest as the best performing at 0.9808, SVM at 0.9668, and KNN with the score of 0.9562. On the whole, Random Forest Classifier performed better than SVM and KNN in most of the performance measures used such as specificity, precision, F1-score, and Matthews Correlation Coefficient, hence, the most successful classifier at predicting DDoS attacks in this test.

Figure 2 illustrates the comparison of the performance of three machine learning classification algorithms, which are SVM, KNN, and Random Forest in identifying DDoS attacks conducted on an IoT environment with two different ratios of training/test data 60/40 and 50/50. The bar chart shows the accuracies of the two separate splits of each of the classifiers. The 60/40 configuration gives the KNN classifier a top accuracy rate of 99.9% followed by the Random forest classifier to 99.6% and the SVM to 98%. In the 50/ 50 cut the KNN accuracy decreases to 96.67%, and Random Forest has the highest accuracy of 98.53% and SVM having accuracy of 97.5%. These findings demonstrate the various performances of each of the classifiers and how Random Forest has similar high accuracy in both ratios of splitting.

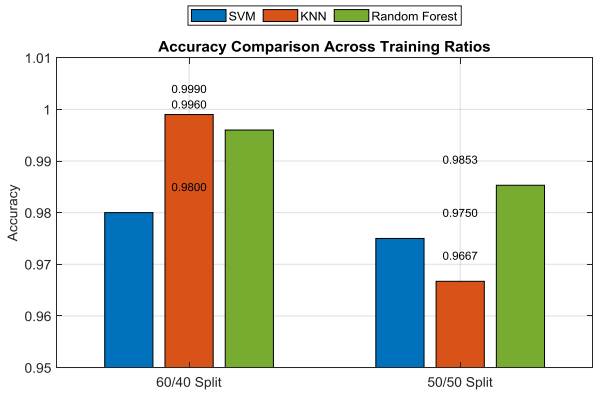

Fig. 2- Performance Comparison for SVM, KNN and Random Forest

Figure 3 shows the performance evaluation of the three models: SVM, KNN, and Random Forest in comparison with each other under two training/testing data distribution ratios: 60/40, and 50/50. The charts are line plots showing the performance of the each of the classifications in these splits. It is clear that in both 60/40 and 50/50 splits, Random Forest performs best as is evidenced by the steepest positive slope in both splits. On the other hand, KNN experiences an increment of accuracy albeit not that sharp, particularly in the 60/40 split. Upon contrast SVM has had little or no variation in its results under both split ratios as its line tends to be flat in the 60/40 split and in the 50/50 split it has a slight negative inclination. This takes into consideration that Random Forest works better compared to both KNN and SVM showing greater improvement as the training data varies between the two ratios.

https://doi.org/10.46335/IJIES.2025.10.4.17 e-ISSN: 2456-3463
Vol. 10, No. 4, 2025, PP. 83-91

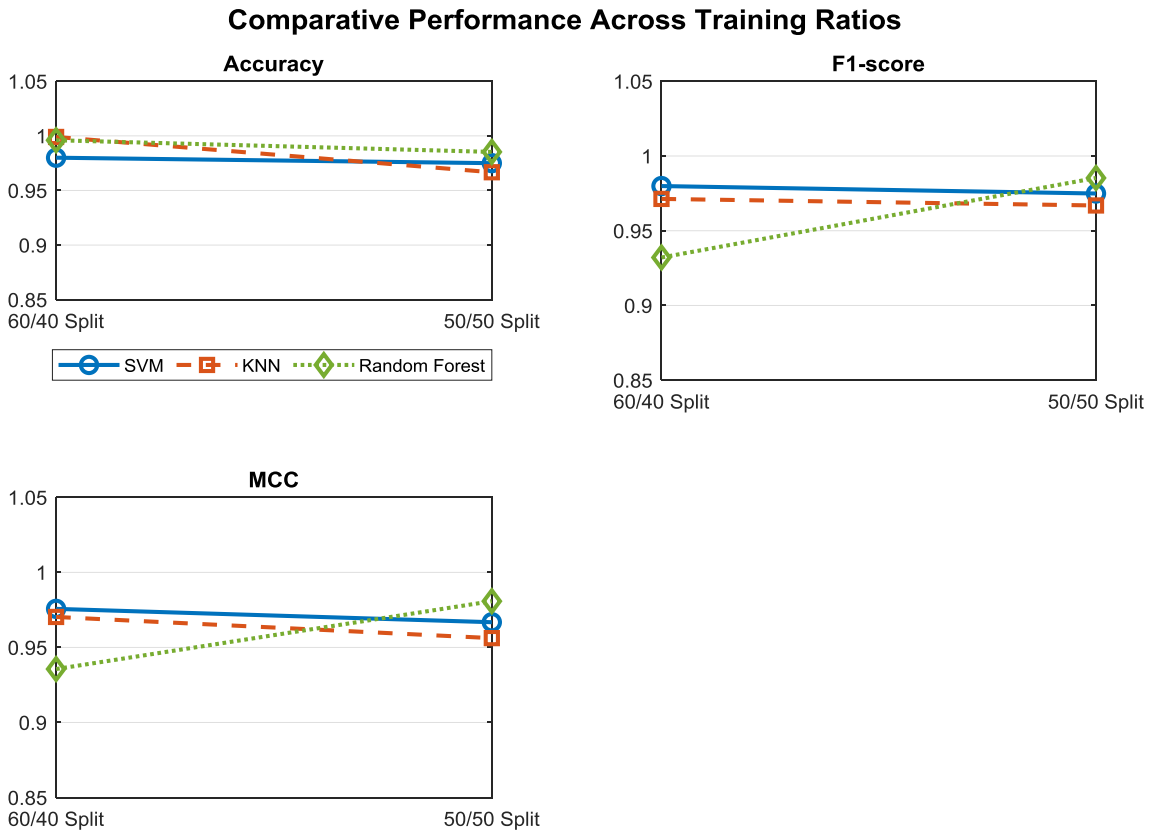

Fig. 3- Comparative Performance Across Training Ratios

CONCLUSION

The conducted research proved that machine learning classifiers, in this case K-Nearest Neighbors (KNN), Support Vector Machines (SVM), and Random Forest, are efficient in identifying Distributed Denial of Service (DDoS) attacks in IoT. When the performance of these classifiers was measured in terms of 50% learning rate and 50% testing rate, all the three classifiers performed adequately but Random Forest gave better results than SVM and KNN in most of the variables that include accuracy, specificity, precision, F1-score and Matthews Correlation Coefficient.

The classification attained its maximum value (98.53%), minimum value (0.0147), and maximum Matthews Correlation Coefficient (0.9808) indicating that the Random Forest has a good ability to classify DDoS attack and non-attack cases. Moreover, it also outperformed other metrics pertinent to identifying positive (sensitivity) and negative (specificity) cases, and thus it was the most consistent classifier of DDoS in the current experiment.

The results highlight the prospects of developing a machine learning-based detection method beyond the shortcomings of the past methodologies against DDoS attacks that have been continuously progressing. The given research sheds light on the advantages and disadvantages of each of the classifiers, and it could be used to recommend the most suitable model to build real-world DDoS detection apps in the realm of IoT. Using the outcomes of this paper, the detection systems of the DDoS can be more correct, effective, and trustworthy, which in turn will lead to even stronger security solutions of the IoT systems that will be associated with more and more connectivity. The scalability of these classifiers can also be tested in the future, and it should be seen how the detection systems will perform as long as there are changing attack strategies.

REFERENCES

[1] *Becerra-Suarez, F.L., Fernández-Roman, I. and Forero, M.G., 2024. Improvement of distributed denial of service attack detection through machine learning and data processing. Mathematics, 12(9), p.1294.*

[2] *Kumar, D., & Kumar, M. (2024). Hybrid Cryptographic Approach for Data Security Using Elliptic Curve*

Cryptography for IoT'. International Journal of Computer Network and Information Security (IJCNIS).

[3] *Połap, D., Srivastava, G., Jolfaei, A. and Parizi, R.M., 2024, July. Blockchain technology and neural networks for the internet of medical things. In IEEE INFOCOM 2024-IEEE conference on computer communications workshops (INFOCOM WKSHPS) (pp. 508-513). IEEE.*

[4] *Lonea, Alina Madalina, Daniela Elena Popescu, and Huaglory Tianfield. "Detecting DDoS attacks in cloud computing environment." International Journal of Computers Communications & Control 8, no. 1 (2022): 70-78.*

[5] *Baig, Zubair A., Surasak Sanguanpong, Syed Naeem Firdous, Tri Gia Nguyen, and Chakchai So-In. "Averaged dependence estimators for DoS attack detection in IoT networks." Future Generation Computer Systems 102 (2024): 198-209.*

[6] *Tertytchny, Georgios, Nicolas Nicolaou, and Maria K. Michael. "Classifying network abnormalities into faults and attacks in IoT-based cyber physical systems using machine learning." Microprocessors and Microsystems 77 (2024): 103121.*

[7] *da Silva, L. E., and Denis Vinicius Coury. "Network traffic prediction for detecting DDoS attacks in IEC 61850 communication networks." Computers & Electrical Engineering 87 (2024): 106793.*

[8] *Kosmanos, Dimitrios, Apostolos Pappas, Leandros Maglaras, Sotiris Moschoyiannis, Francisco J. Aparicio-Navarro, Antonios Argyriou, and Helge Janicke. "A novel intrusion detection system against spoofing attacks in connected electric vehicles." Array 5 (2024): 100013.*

[9] *Liu, Hongyu, Bo Lang, Ming Liu, and Hanbing Yan. "CNN and RNN based payload classification methods for attack detection." Knowledge-Based Systems 163 (2019): 332-341.*

[10] *Li, Wenjuan, Weizhi Meng, Lam-For Kwok, and H. S. Horace. "Enhancing collaborative intrusion detection networks against insider attacks using supervised intrusion sensitivity-based trust management model." Journal of Network and Computer Applications 77 (2017): 135-145.*

[11] *Singh, Karanpreet, Paramvir Singh, and Krishan Kumar. "User behavior analytics-based classification of application layer HTTP-GET flood attacks." Journal of Network and Computer Applications 112 (2018): 97-114.*

[12] *SaiSindhuTheja, Reddy, and Gopal K. Shyam. "An efficient metaheuristic algorithm based feature selection and recurrent neural network for DoS attack detection in cloud computing environment." Applied Soft Computing 100 (2021): 106997.*

[13] *Volkov, S. S., and I. I. Kurochkin. "Network attacks classification using Long Short-term memory based neural networks in Software-Defined Networks." Procedia Computer Science 178 (2024): 394-403.*

[14] *Branitskiy, Alexander, and Igor Kotenko. "Hybridization of computational intelligence methods for attack detection in computer networks." Journal of Computational Science 23 (2017): 145-156.*