University of Baghdad
College of Engineering
JE
JOURNAL OF ENGINEERING

Journal of Engineering

journal homepage: www.jcoeng.edu.iq

Volume 32 Number 8 August 2026

A High-Precision Machine Learning Algorithm for Detecting DDoS Attacks in a Cloud Computing Environment

Noor Takla [1, *], **Mohamed Mohamed** [2]

[1] Web Science, Syrian Virtual University, Damascus, Syria
[2] Faculty of Informatics & Communications Engineering, Arab International University, Daraa, Syria

ABSTRACT

A Distributed Denial of Service (DDoS) attack is an evil attempt to flood a website or a network with malicious traffic to force it to slow down or stop working and it is now extending to other technologies like cloud computing, IoT, and edge computing. This attack has different types, the attacker can exploit the UDP protocol to flood the victim's devices with a huge amount of data, or exploit vulnerabilities in network protocols, or may target the application layer. All of the above attempts to overwhelm all available resources including memory, CPU, and potentially the entire network aiming to incapacitate the victim's machine or server. Despite numerous proposed defensive mechanisms, these mechanisms often fall short as attackers continuously adapt using new automated tools. This is why we propose a machine learning-based approach for DDoS attack detection in cloud computing environments. Using machine learning classifiers, Random Forest (RF) and K-Nearest Neighbors (KNN) and compared based on classification performance and computational efficiency. Experimental results showed that the Random Forest classifier achieved the best performance by reaching an accuracy of 99.97% with minimal false positives. Finally, integrate the best selected model into a Flask-based real-time detection system able to classify generated traffic as either Normal or DDoS Attack.

Keywords: Attacks, Cloud computing, DDoS, Machine learning, Random Forest.

1. INTRODUCTION

Digital infrastructure has now become a key part of business continuity with the advent of digital transformation. This dependency has made companies more prone to cybersecurity threats, including DDoS attacks **(Ayele, 2022)**. In contrast to previous attacks, where only one computer was used as an attack vector, DDoS attacks use a network of compromised devices to overload the targeted system. Compromised devices, often called "bots" or "zombies," send similar types of traffic **(Nazario, 2008)**. These bots are centrally controlled by the attackers and are responsible for sending a flood of malicious requests to the

*Corresponding author
Peer review under the responsibility of University of Baghdad.
https://doi.org/10.31026/j.eng.2026.08.01

Article received: 15/02/2026
Article revised: 23/06/2026
Article accepted: 09/07/2026
Article published: 01/08/2026

victimized machine instead of genuine user requests **(Amrish et al., 2022)**. With increasing complexity and scalability in cloud computing and SDN networks, the risk of DDoS attacks has been amplified **(Dong et al., 2019)**. Traditional firewall and detection techniques have become ineffective due to the evolving nature of attacks **(Mehboob et al., 2024)**. As a result, these DDoS attacks cause substantial economic losses for companies, especially small and medium-sized enterprises (SMEs) **(Herath, 2024)**.

There are usually three types of DDoS attacks. Namely, volumetric attacks, protocol-based attacks, and application-layer attacks **(Afraji et al., 2025)**. In turn, there are three types of defense mechanisms, namely, detection, mitigation, and prevention strategies **(Somani et al., 2017)**. A key characteristic of DDoS attacks is their non-compliance with standard network congestion control mechanisms, as they transmit packets at extremely high rates **(Bahashwan et al., 2024)**. Hence, it is crucial to filter malicious traffic as close to its source as possible **(Nur, 2021)**. However, an inherent drawback of relying solely on monitoring traffic volume at the victim side is the inability to identify whether the increase in traffic is caused by flash crowds or real attacks **(Peng et al., 2004)**. Furthermore, more sophisticated attack types, such as link-flooding attacks, utilize low-rate and seemingly legitimate traffic, making them significantly harder to detect compared to traditional flooding attacks. These attacks depend on coordinated botnet behavior to target specific network links rather than directly attacking servers **(Singh et al., 2019)**.

Therefore, attack detection techniques that are precise and efficient should be developed and implemented. These techniques should work in real-time **(Aktar and Nur, 2023; Awan et al., 2021; Ahmad et al.,2020)**. Deep learning and machine learning can be relied upon to detect DDoS attacks in cloud computing environments **(Deniz and Sertas, 2023; Osanaiye et al.,2018)**, especially when using Anomaly-Based Detection methods **(Dincalp et al., 2018)** because, although scalable and flexible, cloud computing systems remain exposed to dangerous DDoS attacks. Fuzzy logic and neural networks should also be explored for possible prevention and counterattack techniques **(Khordadpour and Ahamdi, 2023)**. There are also various defense strategies, such as signature-based detection, anomaly-based detection, filtering, rate limiting, and multi-layered defense strategies **(Bonguet and Bellaich, 2017)**. In general, early intelligent detection of attacks with accurate attack classification and dynamic response (filtering + restriction + resource allocation) may be among the best ways to prevent DDoS attacks **(Rahman et al., 2019)**.

Information theory is also very significant due to its ability to spot any abnormal changes without the need for prior training, thus helping in identifying changes that might be an attack **(Alarqan et al., 2025)**. Particularly, when using conditional entropy with multiple machine learning classifiers **(Pandey and Mishra, 2025)**. And since entropy alone gives false alarms and machine learning alone is costly, combining the two enables us to detect attacks early and accurately **(Hassan et al., 2024)**.

As attacks are constantly evolving, defense mechanisms require continuous research and development in order to be able to function in real-time, not only based on offline information **(Mittal et al.,2023)**. To address key challenges such as data manipulation and leakage **(Alouffi et al., 2021)**.

If we take a look at the studies in this field, for example, researchers from **(Bawany et al., 2017)** proposed a novel framework towards detection and prevention of DDoS attacks in an extensive network that comprises a smart city built on SDN infrastructure. Furthermore, researchers in **(Alduailij et al., 2022)** exploited the capabilities of the random forest algorithm to detect DDoS attacks.

Some of the solutions, such as **(Das et al.,2019)** designed a Network Intrusion Detection System (NIDS) by integrating multiple classifiers through ensemble modeling techniques and considered the fact that every classifier is capable of identifying particular types of attacks. They tested this NIDS system on an NSL-KDD data set with a reduced feature set and found that this NIDS system was able to identify 99.1% of the DDoS attack cases. However, it would be appropriate if we looked at the research works that compared the performance of different machine learning classification algorithms, such as **(Khuphiran et al., 2018)**. They analyzed the traditional SVM and a newly emerged deep learning approach called deep feed forward (DFF) and concluded that DFF should be preferred in the case of maximum accuracy, while SVM should be used when the classification speed is vital.

(Doshi et al., 2018) used IOT device sources to automatically detect DDoS attacks using low-cost machine learning algorithms and traffic data that is flow-based and protocol-agnostic, using IoT-specific network behaviors (e.g. limited number of endpoints and regular time intervals between packets).

Another research study explored the performance of the Naive Bayes algorithm compared to RF and KNN **(Bhushan and Gupta, 2019; Dahiya and Gupta, 2020; Al-Qerem et al., 2020)**. Founding that Naive Bayes offers better accuracy, making it a strong candidate for detecting DDoS attacks.

We can observe that no single classification algorithm is always the best for every problem. Rather, the performance of algorithms varies depending on the nature of the data. Therefore, the goal is to choose the most appropriate algorithm for each problem instead of always using the same algorithm **(Pise and Kulkarni, 2016)**.

In this work, we present a machine learning-based framework for DDoS attack detection in cloud computing environments. The framework evaluates and compares Random Forest and K-Nearest Neighbor classifiers using the CICDDoS2019 dataset and deploys the best-performing model within a real-time Flask-based detection environment, which allows testing the proposed framework in simulated real-time applications.

This paper provides a machine learning framework for detecting DDoS attacks in cloud environments, where traditional security approaches often struggle to identify evolving attack patterns in real time, especially since attackers are constantly employing new attack patterns. Therefore, the proposed methodology integrates data analysis, feature selection, model comparison and real-time deployment using a user interface to improve attack detection performance.

- This research aims to develop an effective intrusion detection model by applying two machine learning algorithms, such as random forest and KNN, to identify and classify network traffic into normal traffic or DDoS attacks. Experimental results showed that the random forest classifier outperformed KNN by achieving an accuracy of 99.98%, making it the final selected model.
- Analysis network traffic features and identify the most relevant attributes affecting attack detection. Implementing correlation analysis to select the top nine most influential features, which can reduce data complexity while maintaining high classification performance.
- Enhance cybersecurity in cloud and network environments by providing an intelligent detection mechanism capable of identifying malicious traffic patterns at an early stage, helping maintain service availability and network continuity.
- Provide real-time solution by using the best trained detection model in a real-time environment using a flask-based web application. This deployment enables live traffic

generation, feature visualization and instant classification of network traffic as either normal or DDoS attack.

2. METHODOLOGY

Fig.1 illustrates the steps involved in the proposed approach:

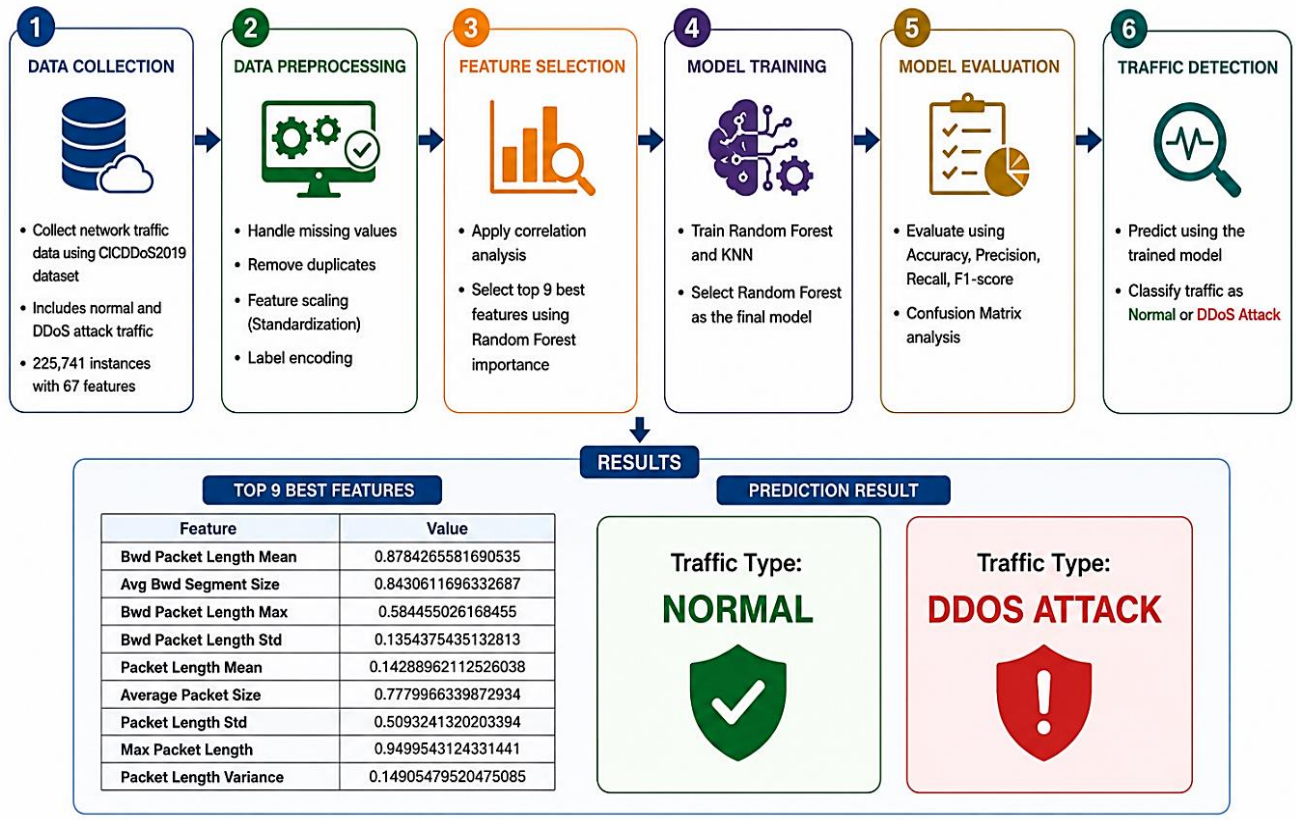

Figure 1. Research methodology.

2.1 Data Collection

This study used the dataset CICDDoS2019, which contains 67 network traffic features that describe different characteristics of communication between the sender and receiver. These features were grouped into several categories.

- Features describe the communication channel and destination service being accessed, such as destination port and flow duration.
- Features represent the number of packets transmitted during communication; these features are very important in DDOS attack domain. For example, total packets sent and total packets returned.
- Another important feature measures the size of transmitted packets, such as packet length and average packet size.
- Traffic timing features measure packet transmission timing.
- Protocol flag features represent TCP control flags used during communication.
- Other features, such as header features.

Fig. 2 shows the class distribution, indicating that the dataset contains slightly more attack traffic samples than normal traffic samples. Where 56.72% of the dataset represents DDoS

attacks, while 43.28% represents normal traffic. Although the class distribution is moderately imbalanced, this imbalance is not severe and remains within an acceptable range for binary classification tasks. In addition, model performance was further assessed using classification metrics analysis to ensure that the results were not biased toward the majority class.

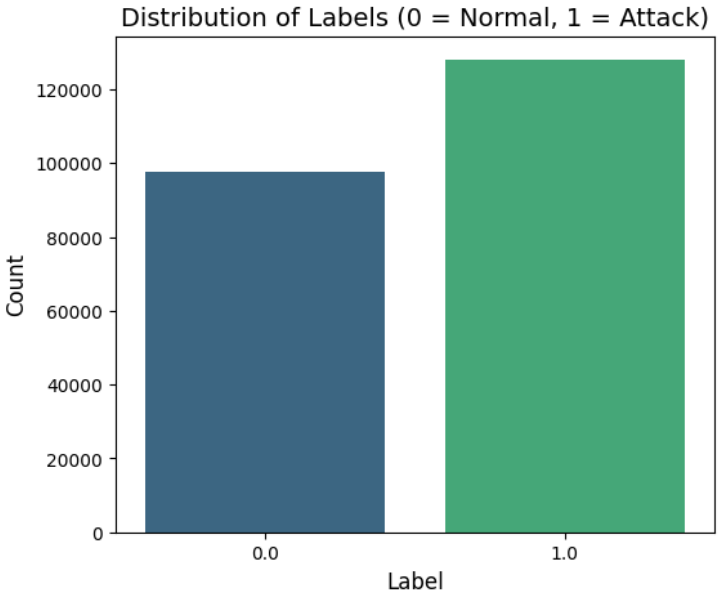

Figure 2. Distribution of samples on dataset.

2.2 Preprocessing

In this step, we prepare the collected data for analysis. This means dealing with missing values by either deleting or replacing them with appropriate ones, and ensuring all features are suitable and consistent for modeling by resizing or normalizing the data. If there are categorical variables, they are converted to numerical variables using techniques such as one-hot encoding to facilitate processing by machine learning algorithms. Also, divide the dataset into two sets to test the performance of the model. These two sets are the training group and the test group. The training group contain 80% of the chosen data and is used to train the model. On the other hand, the remaining 20% is the test group and represents the data that the model will be tested on.

2.3 Features Selection

This step aims to reduce the complexity of model and enhance training efficiency. So, this study identifies the most influential features for DDoS attack detection by applying correlation analysis between each feature and the target label to measure the linear relationship. This helps focus on traffic patterns that are most discriminative between Normal and DDoS Attack traffic. As shown in **Fig. 3,** the top features indicate which network traffic characteristics are most indicative of malicious activity. Features like packet length have the highest correlation with DDoS attacks, suggesting that attack traffic often exhibits larger or more variable packet sizes in backward flows. On the other hand, packet that based on statistics, such as average packet size, also contribute significantly to distinguishing normal from malicious traffic. In this study select the best 9 features. The main objective of this study is to decrease time of training and testing, therefore enhance model performance.

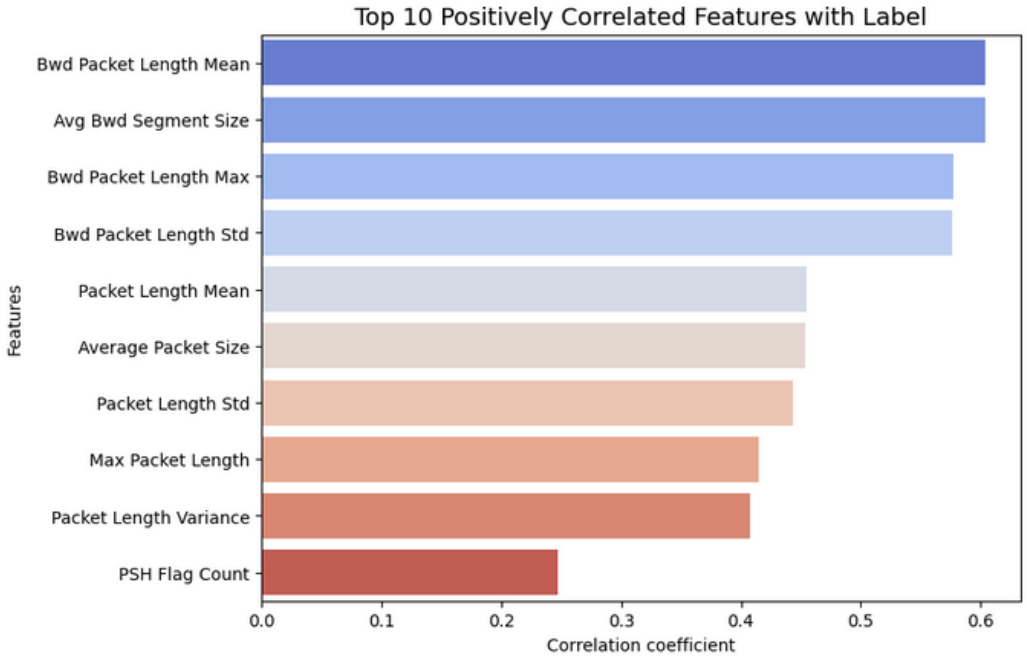

Figure 3. The best features.

2.4 Machine Learning Models

In this step, we choose the two machine learning models, that includes random forest and KNN. The random forest model is constructed using 10 estimators, the number of trees in the random forest. The KNN has 20 neighbors. These models were trained and tested on the train and test groups.

2.5 Model Evaluation

In the final step, we aim to evaluate the performance of the trained model. We use several metrics, such as calculating the Confusion Matrix, data recall, F1 Score, and Precision. These metrics aim to determine the accuracy and reliability of the model and they demonstrate the model's effectiveness in real-world environments and whether it can accurately predict potential attacks without false alarms.

2.6 Deployment Model

Deploying the best model as a Flask-based prototype to simulate real-time DDoS traffic classification, where the best model was exported using Joblib and integrated into the application for inference without retraining.

3. RESULTS AND DISCUSSION

3.1 Random Forest Evaluation Results

The Confusion Matrix is a powerful analytical tool that provides a detailed view of the performance of a random forest model used to detect network anomalies. This matrix helps evaluate the model's effectiveness in classifying network activities as either benign (normal) or potentially malicious.

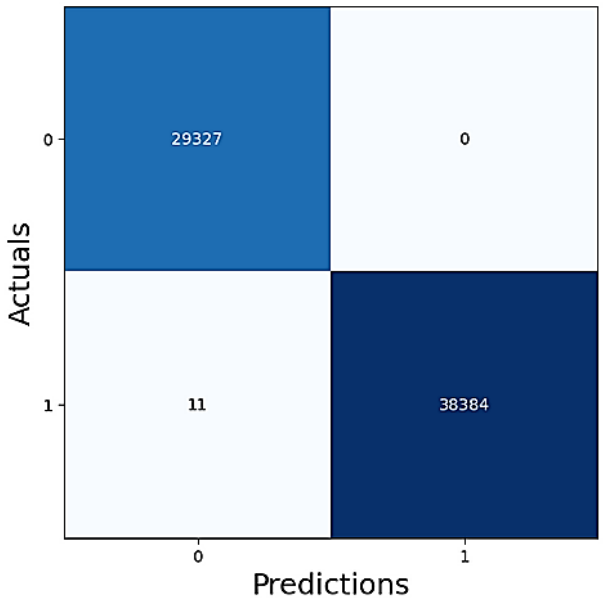

Figure 3. The confusion matrix.

According to **Fig. 3**, the confusion matrix, we analyze the following:

- True Positives (TP): has a value of 29,327. Suggesting that this value indicates the number of times the model was able to correctly identify an attack. This is an indication that the model is very effective in identifying real attacks. This implies that the model is able to detect the majority of the attacks as they happen, thus improving the system's ability to defend against the attacks.
- True Negatives (TN): with a value of 38,384. Indicating the number of instances where the model correctly identified benign activities. We can also note that the number is high, and that demonstrates that the model is also adept at recognizing normal network behavior. This is a positive indicator of the model's ability to avoid false alarms, thus minimizing unnecessary interventions and maintaining efficient system performance.
- False Positives (FP): with a value of 0. This value indicates the number of benign activities that were wrongly identified as attacks, and as we can see, it is none, which is an indication that the model does not mislabel benign activities as attacks, which is a very good thing in a business setting where false alarms should be avoided. This improves the dependability of the model in detecting actual threats.
- False Negatives (FN): with a value of 11. This value represents the number of instances where the model failed to detect an attack. While the ideal number should be zero, a low number of false negatives (only 11) indicates that the model misses very few attacks, meaning it has high reliability in detecting actual threats.

We use the values extracted from **Fig. 3** to calculate a set of important metrics to evaluate the performance of the random forest model used to detect network anomalies according to the following equations:

- Accuracy

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{67711}{67722} = 99.98 \quad (1)$$

The high accuracy (99.98%) shown in Eq. (1) indicates that the model does an exceptional job of correctly classifying network activity, whether these activities are attacks or benign (ordinary) activity. This result reflects the effectiveness of the model in classifying data with high accuracy.

- Precision:

$$Quality\ Precision = \frac{TP}{TP+FP} = \frac{29327}{29327} = 1 \quad (2)$$

A qualitative precision of 1 (or 100%) means that each time the model predicts an attack, the prediction is always correct. This implies that there are no false positives (FPs), which means it has the capability of being 100% precise in distinguishing between benign activities and attack behaviors.

- Recall:

$$Recall = \frac{TP}{TP+FN} = \frac{29327}{29338} = 99.96 \quad (3)$$

Thus, from the given high sensitivity value of 99.96%, this model is excellent in detecting most of the actual attacks, at the same time missing only a few. This shows how highly sensitive this model is towards an attack; therefore, it will detect threats in the majority of cases.

- Specificity:

$$Specificity = \frac{TN}{TN+FP} = \frac{38.384}{38.384} = 1 \quad (4)$$

Specificity of 1 (or 100%) means the model is ideal in correctly predicting benign activities. It implies the model does not make mistakes in labeling legitimate activities as attacks, hence zero or minimal intervention.

- F1 score:

$$F1 - Score = \frac{Precision \times Recall}{Precision+Recall} = 99.98 \quad (5)$$

The F1 measure, which gives the harmonic mean of accuracy and recall, shows that there is a balance between attack detection and accuracy of prediction. A high value of F1, which is 99.98%, signifies that it has a high level of effectiveness in attack detection and correct categorization of normal activities.

3.2 Comparison Between Random Forest and KNN

As shown in **Fig. 4,** both models achieve extremely high accuracy, suggesting that they correctly classify almost all instances in the dataset. But for the precision metric, random forest achieves perfect precision (100%), indicating no false positives in the evaluated results. The result ensures that random forest consistently outperforms KNN, especially in precision, which is critical in intrusion detection systems where false alarms can be costly.
In real-time detection, the testing time is a critical issue, where the model that works in real-time must have very low testing time. As shown in **Fig. 5**, the random forest achieves the lowest testing time compared with KNN, that make it very suitable for real-time detection,

even though it has training time. Due to in real-time not important training time, the most important factor is testing time.

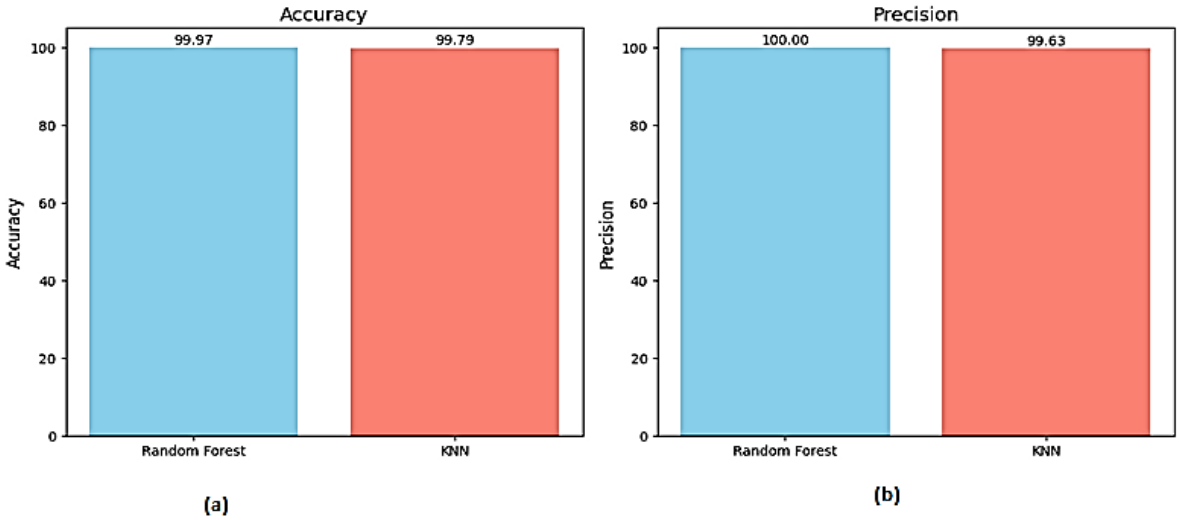

Figure 4. (a) Comparison between Models based on Accuracy, (b) Comparison between Models based on Precision.

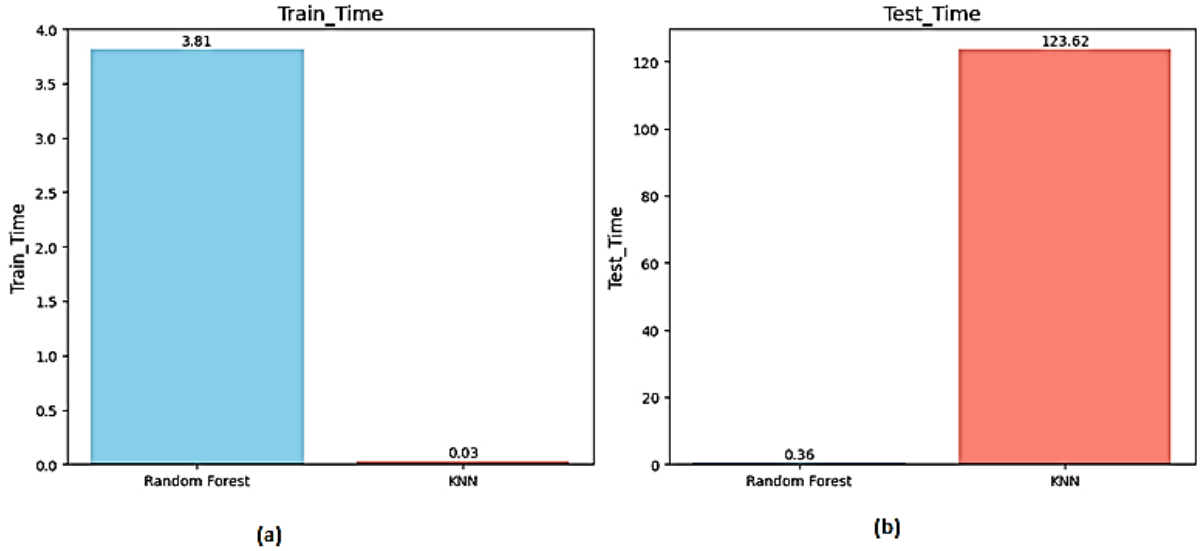

Figure 5. (a) Comparison between Models based on Training Time, (b) Comparison between Models based on Testing Time.

3.3 Deployment Results

According to previous results, the best model is random forest; after training it, we save it into Joblib to use it in real-time detection without retraining. According to **Fig.6**, the random forest model succeeded in classifying network traffic into normal and attack. In this case, when the user clicks on the generate button, our system generates normal or attack traffic as the user selects. After generating traffic, the random forest classifies traffic as normal or an attack. Suggesting that the random forest model can be deployed in real scenarios.

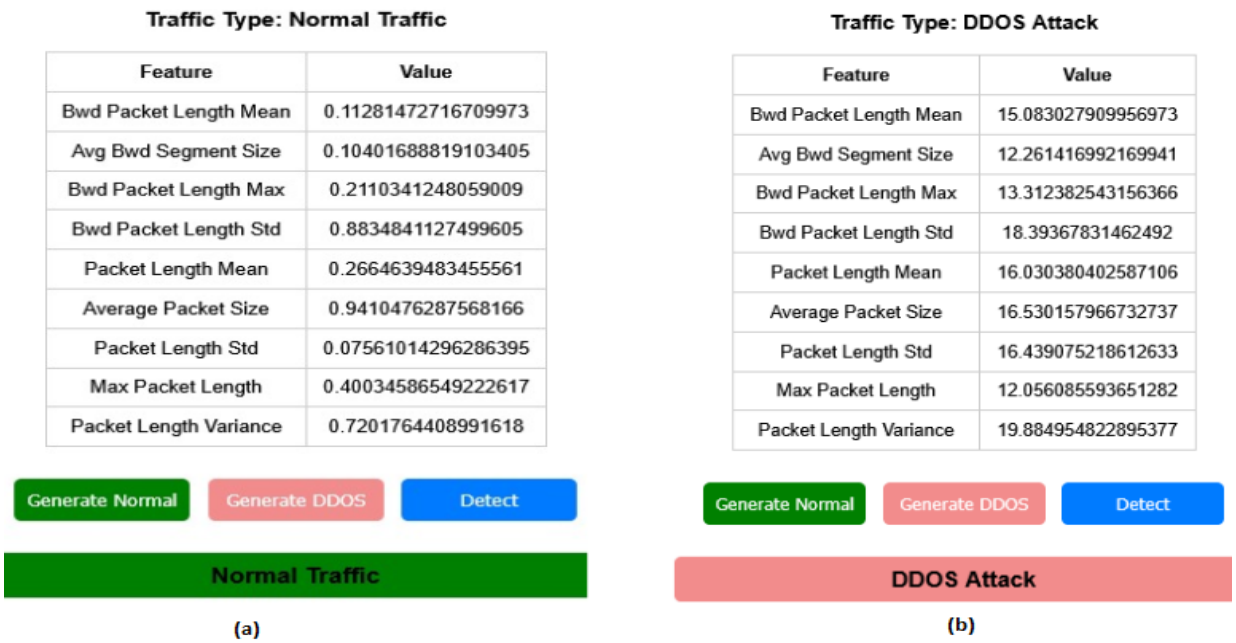

Figure 6. Traffic Detection System (a) Normal Traffic, (b) DDOS Traffic.

3.4 Comparison with Similar Recent Studies

As shown in **Table 1**, machine learning models, particularly random forest, consistently achieve high detection performance for DDoS attacks, while deep learning models like LSTM can perform well if the dataset contains sufficient temporal patterns. Our study contributes by validating the robustness of random forest and showing that even classical models like KNN can provide high detection rates when carefully tuned. But in real-time detection, the random forest is the best selection, which outperformed all previous models. In summary, the proposed model achieved performance comparable to previous studies in this table, while maintaining low testing time and demonstrating suitability for real-time deployment scenarios.

Table 1. Comparison with Similar Studies.

Study	Models Used	Dataset	Results
(Ouhssini et al., 2024)	CNN-LSTM-Transformer AutoCNN-DT and GA	CIDDS-001	F1-score: 99.97%
(Kumar et al., 2023)	LSTM, DNN, KNN	CIC-DDoS2019	Accuracy: 98% for LSTM
(Berríos et al. 2025)	Random Forest, XGBoost, LSTM	CIC-DDoS2019, N-BaIoT	Accuracy: 99.96% for Random Forest
Our study	Random Forest, and KNN	CIC-DDoS2019	Accuracy: 99.97% for Random Forest, and 99.79% for KNN.

4. CONCLUSIONS

In this paper, we develop a DDoS detection model using a random forest algorithm in cloud computing. The results of the experiment show that the model is efficient and provides an accuracy of 99.97% for network anomaly detection and 100% qualitative and qualitative accuracy, meaning that the model does not raise any false alarms and will not misinterpret any other attacks. Also, the high F1 value shows the balance between the capability of attack detection and the precision of prediction for benign activities. The confusion matrix

presented a detailed analysis, proving that the model can detect the vast majority of the attacks with very low false negatives; this makes it a robust and reliable model to work with in real environments. The study epitomizes the effectiveness of machine learning in enhancing network security and detecting cyberattacks in cloud computing, utilizing the Random Forest algorithm. This model provides an overall and effective solution to cybersecurity threats that are targeting cloud networks. Despite the excellent results achieved by the model in this study, some areas can be explored in future research to enhance the effectiveness and efficiency of the proposed system, such as:
First, model optimization using hybrid algorithms: The random forest algorithm can be combined with other machine learning algorithms or deep learning techniques to further improve accuracy and reduce false negatives. Hybrid algorithms can offer more sophisticated solutions for handling complex patterns of cyberattacks. Second, Real-time data integration: Improving the model to work better with real-time data to enable early attack detection and immediate responses. Handling real-time data and processing data flows can enhance model performance in real-world environments. Third, Cost and efficiency analysis: Examining the impact of model implementation in terms of cost and resource efficiency. Understanding computing costs, training time, and forecasting is crucial for widespread model adoption across industries.

Credit Authorship Contribution Statement

Noor Takla: draft writing, validation, methodology, and proofreading. Mohamed Mohamed: Writing – review & editing.

Declaration of Competing Interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

Afraji, D. M., Lloret, J., and Penalver, L., 2025. Deep learning-driven defense strategies for mitigating DDoS attacks in cloud computing environments. KeAi publishing, *Cyber Security and Applications* (3), https://doi.org/10.1016/j.csa.2025.100085

Ahmad, A., Mohamed, M., and Mohamed, I., 2020. Mitigate distributed denial of service attack using dynamic threshold. *Tartous University journal for research and Scientific Studies – engineering Sciences Series*, 4(7), pp. 1-20. https://doi.org/10.5281/zenodo.21227496

Aktar, S. and Nur, A.Y., 2023. Towards DDoS attack detection using deep learning approach. *Computers & Security*, 129(9), P. 103251. https://doi.org/10.1016/j.cose.2023.103251

Al-Qerem, A., Alauthman, M., Almomani, A., and Gupta, B.B., 2020. IoT transaction processing through cooperative concurrency control on fog–cloud computing environment. *Soft Computing*, 24(2-C), pp. 5695-5711. http://doi.org/10.1007/s00500-019-04220-y

Alarqan, M., Belaton, B., Almomani, A., Alauthman, M., Al-Betar, M.A., and Arya, V., 2025. Information theory-based DDoS attack detection in cloud computing. *International Journal of Cloud Applications and Computing*, 15(1), pp. 1-38. http://doi.org/10.4018/IJCAC.369817

Alduailij, M., Khan, Q. W., Taher, M., Sardaraz, M., Alduailij, M., and Malik, F.,2022. Machine-learning-based DDoS attack detection using mutual information and Random Forest feature importance method. *Symmetry*, 14(6). http://doi.org/10.3390/sym14061095

Alouffi, B., Hasnain, M., Alharbi, A., Alosaimi, W., Alyami, H., and Ayaz, M., 2021. A systematic literature review on cloud computing security: threats and mitigation strategies. *IEEE Access*, 9, pp. 57792-57807. http://doi.org/10.1109/ACCESS.2021.3073203

Amrish, R., Bavapriyan, K., Gopinaath, V., Jawahar, A., and Kumar, C. V., 2022. DDoS detection using machine learning techniques. *Journal of IoT in Social, Mobile, Analytics, and Cloud*, 4(1), pp. 24-32. http://doi.org/10.36548/jismac.2022.1.003

Awan, M. J., Farooq, U., Babar, H. A., Yasin, A., Nobanee, H., Hussain, M., Hakeem, O., and Zain, A. M., 2021. Real-Time DDoS attack detection system using big data approach. *Sustainability*, 13(19). http://doi.org/10.3390/su131910743

Ayele, W.Y., 2022. Improving the utilization of digital services: Evaluating contest-driven open data development and the adoption of cloud services*. arXiv preprint arXiv:2212.04491*. http://doi.org/10.48550/arXiv.2212.04491

Bahashwan, A. A., Anbar, M., Manickam, S., Issa, G., Aladaileh, M. A., Alabsi, B. A., Rihan, S. D. A., 2024. HLD-DDoSDN: High and low-rates dataset-based DDoS attacks against SDN. *PLoS ONE,* 19(2). https://doi.org/10.1371/journal.pone.0297548

Bawany, N.Z., Shamsi, J.A. and Salah, K.,2017. DDoS attack detection and mitigation using SDN: Methods, practices, and solutions. *Arab J Sci Eng.*, 42, pp. 425–441. http://doi.org/10.1007/s13369-017-2414-5

Berríos, S., Garcia, S., Hermosilla, P., and Allende-Cid, H., 2025. A machine-learning-based approach for the detection and mitigation of distributed denial-of-service attacks in internet of things environments. *Applied Sciences,* 15(11), P. 6012. https://doi.org/10.3390/app15116012

Bhushan, K., and Gupta, B. B.,2019. Distributed denial of service (DDoS) attack mitigation in software defined network (SDN)-based cloud computing environment. *Journal of Ambient Intelligence and Humanized Computing*, 10(4), pp. 1985-1997. http://doi.org/10.1007/s12652-018-0800-9

Bonguet,A., and Bellaich, M., 2017. A Survey of denial-of-service and distributed denial of service attacks and defenses in cloud computing. *Future Internet*,9(3), P. 43. http://doi.org/10.3390/fi9030043

Dahiya, A., and Gupta, B,2020. A reputation score policy and Bayesian game theory based incentivized mechanism for DDoS attacks mitigation and cyber defense. *Future Generation Computer Systems*, 117(12), pp. 193-204. http://doi.org/10.1016/j.future.2020.11.027

Das, S., Mahfouz, A. M., Venugopal, D., and Shiva, S., 2019. DDoS intrusion detection through machine learning ensemble. *A Review. in Proceedings IEEE 19th International Conference on Software Quality, Reliability and Security Companion, QRS-C. 22-26 July 2019, Sofia, Bulgaria*. pp. 471-477. http://doi.org/10.1109/QRS-C.2019.00090

Deniz, E., and Serttas, S., 2023. Deep learning-based distributed denial of service detection system in the cloud network. *Journal of Scientific Reports-A,* 48, pp. 16–33. http://doi.org/10.59313/jsr-a.1333839

Dincalp, U., Guzel, M. S., Sevine, O., Bostanci, E., and Askerzade, I., 2018. Anomaly based distributed denial of service attack detection and prevention with machine learning. *A Review. in Proceedings 2nd*

International Symposium on Multidisciplinary Studies and Innovative Technologies, ISMSIT. October 2018, Ankara. pp. 1-4. http://doi.org/10.1109/ISMSIT.2018.8567252

Dong, S., Abbas, k., and Jain, R., 2019. A survey on distributed denial of service (DDoS) attacks in SDN and cloud computing environments. *IEEE Access*, 7, pp. 80813–80828. http://doi.org/10.1109/ACCESS.2019.2922196

Doshi, R., Apthorpe, N., and Feamster, N., 2018. Machine learning DDoS detection for consumer internet of things devices. *in Proceedings International Conference on IEEE Security and Privacy Workshops, SPW. IEEE Xplore, 24-24 May 2018, San Francisco, CA*. pp. 29-35. http://doi.org/10.1109/SPW.2018.00013

Hassan, A.I., Abd El Reheem, E., and Guirguis, S.K., 2024. An entropy and machine learning based approach for DDoS attacks detection in software defined networks. *Scientific Reports,* 14(1), P. 18159. http://doi.org/10.1038/s41598-024-67984-w

Herath, B.M.T.I.T, 2024. The economic impact of cyberattacks research paper the economic impact of cyberattacks: A comprehensive analysis. *SSRN Electronic Journal*. http://doi.org/10.2139/ssrn.4885666

Khordadpour, P., and Ahamdi, S., 2023. FIDS: Fuzzy intrusion detection system for simultaneous detection of DoS/DDoS attacks in ؤloud computing. *arXiv preprint arXiv:2305.16389.* http://doi.org/10.48550/arXiv.2305.16389

Khuphiran, P., Leelaprute, P., Uthayopas, P., Ichikawa, K., and Watanakeesuntorn, W., 2018. Performance comparison of machine learning models for DDoS attacks detection. *A Review. in Proceedings 22nd International Computer Science and Engineering Conference, ICSEC. IEEE Xplore, 21-24 November 2018, Chiang Mai, Thailand*. pp. 1-4. http://doi.org/10.1109/ICSEC.2018.8712757

Kumar, D., Pateriya, R. K., Gupta, R. K., Dehalwar, V., and Sharma, A., 2023. DDoS detection using deep learning. *Procedia Computer Science*, 218, pp. 2420–2429. https://doi.org/10.1016/j.procs.2023.01.217

Mehboob, T., Sumra, I. A., and Shahzadi, I.,2024. Detection of DDoS/DoS attack methodologies in cloud computing network: A survey. *Journal of Cybersecurity and Information Management*, 8(1), pp. 1–11.

Mishra, A., Gupta, B. B., Peraković, D., Peñalvo, F. J. G., and Hsu, C. H. 2021. Classification based machine learning for detection of DDoS attack in cloud computing. *in Proceedings International Conference on consumer electronics, ICCE. IEEE Xplore, 10-12 January 2021, Las Vegas, NV, USA*. pp. 1-4. http://doi.org/10.1109/ICCE50685.2021.9427665

Mittal, M., Kumar, K., and Behal, S., 2023. Deep learning approaches for detecting DDoS attacks: A systematic review. *Soft computing*, 27(18), pp. 13039-13075. http://doi.org/10.1007/s00500-021-06608-1

Nazario, J., 2008. DDoS attack evolution, *Network Security*, 2008(7), pp. 7-10. http://doi.org/10.1016/S1353-4858(08)70086-2

Nur, A. Y., 2021. Combating DDoS attacks with fair rate throttling. A Review. *in Proceedings IEEE International Systems Conference, SysCon. 15 April - 15 May 2021, Vancouver, BC, Canada*. P. 3. http://doi.org/10.1109/SysCon48628.2021.9447054

Osanaiye, O., Choo, K. R., Dehghantanha, A., Xu, Z., and Dlodlo, M., 2018. Ensemble-based multi-filter feature selection method for DDoS detection in cloud computing. *arXiv preprint arXiv:1807.10443*. http://doi.org/10.48550/arXiv.1807.10443

Ouhssini, M., Afdel, K., Agherrabi, E., Akouhar, M., and Abarda, A., 2024. Deep defend: A comprehensive framework for DDoS attack detection and prevention in cloud computing. *Journal of King Saud University - Computer and Information Sciences, 36*, Article 101938. https://doi.org/10.1016/j.jksuci.2024.101938

Pandey, N., and Mishra, P.K., 2025. Conditional entropy-based hybrid DDoS detection model for IoT networks. *Computers & Security*, 150, P. 104199. http://doi.org/10.1016/j.cose.2024.104199

Peng, T., Leckie, C. and Ramamohanarao, K., 2004. Proactively detecting distributed denial of service attacks using source IP address monitoring. *A Review. in Proceedings International Conference on Research in Networking, 9-14 may, 2004, Springer, Berlin, Heidelberg*. pp. 771–782. http://doi.org/10.1007/978-3-540-24693-0_63

Pise, N., and Kulkarni, P., 2016. Algorithm selection for classification problems. A Review. *in Proceedings SAI Computing Conference, SAI. 13-15 July 2016, London, UK*. pp. 203-211. http://doi.org/10.1109/SAI.2016.7555983

Rahman, O., Quraishi, M. A. G., and Lung, C.,2019. DDoS attacks detection and mitigation in SDN using machine learning. A Review. *in Proceedings IEEE World Congress on Services, SERVICES. 08-13 July 2019, Milan, Italy*. pp. 184-189. http://doi.org/10.1109/SERVICES.2019.00051

Singh, P., Rehman, S., and Manickam, S., 2019. Comparative analysis of state-of-the-art EDoS mitigation techniques in cloud computing environment. *arXiv preprint arXiv:1905.13447*. http://doi.org/10.48550/arXiv.1905.13447

Somani, G., Gaur, M. S., Sanghi, D., Conti, M., and Buyya, R., 2017. DDoS attacks in cloud computing: Issues, taxonomy, and future directions. *Computer Communications*, 107, pp. 30-48. http://doi.org/10.1016/j.comcom.2017.03.010

خوارزمية تعلم آلي عالية الدقة للكشف عن هجمات DDoS في بيئة الحوسبة السحابية

نور تقلا[1]، *، محمد محمد[2]

[1] علوم الويب، الجامعة الافتراضية السورية، دمشق، سوريا

[2]كلية الهندسة المعلوماتية والاتصالات، الجامعة العربية الدولية الخاصة، درعا، سوريا

الخلاصة

يعتبر هجوم الحرمان المُوزّع للخدمات (DDoS) هو محاولة خبيثة لإغراق موقع ويب أو شبكة بحركة مرور ضارة بهدف إبطائها أو تعطيلها عن العمل، وهو الآن يمتد ليشمل تقنيات أخرى مثل الحوسبة السحابية، وإنترنت الأشياء، والحوسبة الحدودية (الحوسبة الطرفية). لهذا الهجوم أنواع مختلفة، إذ يمكن للمهاجم استغلال بروتوكول UDP لإغراق أجهزة الضحية بكم هائل من البيانات، أو استغلال ثغرات في بروتوكولات الشبكة، أو قد يستهدف طبقة التطبيقات. وتهدف كل هذه المحاولات إلى استنزاف جميع الموارد المتاحة بما في ذلك الذاكرة، والمعالج، وربما الشبكة بأكملها، بهدف تعطيل جهاز أو خادم الضحية. على الرغم من العديد من آليات الدفاع المُقترحة، فإن هذه الآليات غالباً ما تكون قاصرة لأن المهاجمين يستخدمون باستمرار أساليب جديدة في شن الهجوم. ولهذا السبب نقترح نهجاً قائماً على تعلّم الآلة للكشف عن هجمات الحرمان المُوزّع للخدمات في بيئات الحوسبة السحابية. باستخدام مصنّفات تعلّم الآلة، وهما الغابة العشوائية (Random Forest) وجار الأقرب (K–Nearest Neighbors)، وقمنا بمقارنتهما بناءً على أداء التصنيف والكفاءة الحسابية. أظهرت النتائج التجريبية أن مصنّف الغابة العشوائية حقق أفضل أداء حيث وصلت دقته إلى 99.97% مع أقل نسبة إنذارات كاذبة. وأخيراً، قمنا بدمج النموذج الأفضل اختياراً في نظام كشف فوري قائم على Flask، قادر على تصنيف حركة المرور المُولّدة إما على أنها عادية أو أنها هجوم DDoS.

الكلمات المفتاحية: التعلم الآلي، الحوسبة السحابية، الغابة العشوائية، هجوم حجب الخدمة الموزع.