International Journal of Security Sciences (IJSS)
July-December-Issue, Vol. 1, No. 1 (2026) | DOI: 10.66261/f1me6k95

Article – e006

Balanced and Reduced-Dimensionality DDoS Detection: A SMOTE-ENN and RFE Approach with Naïve Bayes and SVM

Reena[1], Abhishek Kajal[2], Narender Kumar[3], Delisha Kajal[4] Neha Sharma[5]

[1,2,3,5] Department of CSE, Guru Jambheshwar University of Science and Technology, Hisar, India
[4] Indian Institute of Technology Madras (IITM), Chennai, Tamil Nadu, India

Received: 2026-09-15 **Revision Received:** 2026-10-01 **Accepted:** 2026-10-20

ABSTRACT

Distributed Denial of Service (DDoS) attacks threaten the availability of network services, and rule-based intrusion detection systems (IDS) struggle to recognise their evolving patterns. This paper presents a machine-learning IDS pipeline for DDoS detection that combines min–max normalisation, hybrid class balancing with SMOTE-ENN, and Recursive Feature Elimination (RFE) with a Random Forest estimator, which reduces 78 flow features to 20. Naïve Bayes and Support Vector Machine (SVM) classifiers are trained on the Friday-afternoon DDoS capture of CIC-IDS2017 (225,745 flow records, obtained from Kaggle), with resampling applied to the training partition only, and are evaluated on a held-out 20% test set using accuracy, precision, recall, F1-score and execution time. Naïve Bayes achieved 98.91% accuracy, a 98.91% F1-score and a 1.2 s execution time, compared with 98.57%, 98.56% and 2.5 s for SVM, and it missed no attack flows in the test set. Both classifiers reached a ROC AUC of 0.99. These results come from a single dataset and a single train–test split without significance testing, so the small differences between the two classifiers should be regarded as indicative rather than conclusive. Within these limits, a lightweight classifier on a balanced, reduced feature space detected DDoS traffic accurately at low computational cost.

KEYWORDS: DDoS Detection, IDS, SMOTE-ENN, Recursive Feature Elimination, Naïve Bayes, Support Vector Machine

1. INTRODUCTION

Distributed Denial of Service (DDoS) attacks, which are typically organised as botnets, flood a server or network with manipulated traffic from numerous compromised systems until the service becomes unavailable to genuine users. ICMP, SYN/ACK, UDP and HTTP request floods are common variants of DDoS attacks. Traditional defences such as firewalls, content delivery networks and signature-based intrusion detection systems (IDS) struggle with the large volume and variety of contemporary attacks, and the repercussions range from financial loss to reputational harm and the exposure of critical sensitive data [1], [2].

Machine learning (ML) is the most effective alternative to counter DDoS attacks. Labelled network flows are utilized to train machine learning based methods so as to enable them classification of new flows automatically, learn the statistical signature of benign and malicious network traffic, and be retrained even in case the attack behaviour changes. However, machine learning-based detection

© 2026 The Author(s) | Published by - Scholiora Research Publications
Creative Commons Attribution 4.0 International License (CC BY 4.0)

methods faces several challenges: firstly, it depends on the class balancing and quality of the training data; secondly, false alarms generated by these get interfere with genuine users; thirdly, it needs to be scaled up with very high traffic volumes; and lastly, it can be circumvented by adaptive attackers [1], [2].

This paper addresses two key challenges of flow based data analysis issues. First, flow datasets are mostly imbalanced, resulting models trained on skewed data tends to favour the majority class. Second, flow records contain less informative or redundant features, resulting add noise and increase computational cost. We therefore examine whether a pipeline that integrates lightweight ML classifiers, wrapper-based feature selection and hybrid resampling can detect DDoS traffic reliably while remaining computationally affordable under resource constraints.

The contributions of this work are: (i) a leakage-aware pipeline that applies SMOTE-ENN to the training partition only and uses Random Forest-based RFE to reduce 78 flow features to 20; (ii) a comparison of Naïve Bayes and SVM using accuracy, precision, recall, F1-score, ROC AUC and execution time; and (iii) an error analysis based on confusion matrices that shows the practical trade-off between missed attacks and false alarms.

The remainder of the paper is organised as follows. Section 2 reviews related work, Section 3 describes the methodology, Section 4 presents and discusses the results, Section 5 states the limitations, and Section 6 concludes and outlines future work.

2. RELATED WORK

Classical ML classifiers. Saravanababu et al. [3] classified and predicted DDoS attacks with KNN and DNN models on StandardScaler-normalised data and evaluated them with confusion matrices. Kumari and Mrunalini [4] compared Logistic Regression and Naïve Bayes for DoS detection on the CAIDA 2007 dataset in the Weka environment. In software-defined networks (SDN), Sekar et al. [5] obtained their highest accuracy and detection rate with a hybrid of Decision Tree and SVM, and Deepa et al. [6] reported that a hybrid ML model outperformed a baseline in accuracy, detection rate and false-alarm rate when protecting the SDN controller. Alhalabi et al. [7] reviewed ML-based DDoS detection for IoT-oriented intelligent information systems and built a decision-tree detector.

Feature selection and dimensionality reduction. Avçı and Koca [8] combined an ANN predictor with an SVM and Slime Mould Optimisation-based feature selection to predict DDoS attacks in building management systems, reporting 97.44% accuracy against 96.46% for KNN. Abbas and Almhanna [9] preprocessed data with encoding, logarithmic scaling and principal component analysis to reduce dimensionality. Feature selection has also been combined with semi-supervised K-means [10], with several ML models in SDN [11], and with newly engineered features in SDN [12].

Deep learning. Tennakoon and Fernando [13] used an autoencoder for feature selection and a DNN for application-layer DDoS detection on the CIC DoS 2017 data, reporting a detection rate of 99.84% and a false-negative rate of 0.17%. Aswad et al. [14] fused RNN, LSTM-RNN and CNN components into a bidirectional CNN-Bi-LSTM model for DDoS detection in IoT networks and compared it with each component model.

Hybrid and other approaches. Hybrid schemes have been proposed for previously unseen attacks [15] and, using entropy, for early detection in SDN [16], [17]. Further ML-based detectors and mitigation schemes have been developed for general networks [18], [19], SDN [20], ISP networks [21] and IoT-botnet scenarios [22]. Kajal and Nandal [23] combined a genetic algorithm for feature selection with an ANN classifier to detect DDoS, malware and spoofing attacks. Jangra and Kajal [24] reviewed deep-learning-based IDS and, for an LSTM-based IDS, observed that discarding less significant attributes shortened training and testing time; the review states that the discarded attributes did not affect accuracy but reports no ablation to quantify this. Table I summarises the studies for which the technique and setting are clear.

© 2026 The Author(s) | Published by - Scholiora Research Publications
Creative Commons Attribution 4.0 International License (CC BY 4.0)

TABLE I. Summary of selected related work

Ref.	Technique	Setting / dataset
[3]	KNN, DNN	—
[4]	Logistic Regression, Naïve Bayes	CAIDA 2007
[5]	Decision Tree + SVM hybrid	SDN
[6]	Hybrid ML model	SDN controller
[8]	ANN + SVM, SMOA feature selection	Building management systems
[9]	Encoding, log scaling, PCA	—
[13]	Autoencoder + DNN	CIC DoS 2017
[14]	CNN-Bi-LSTM	IoT networks
[23]	Genetic Algorithm + ANN	KDD Cup99; Kaggle malware data
This work	**SMOTE-ENN, RFE, Naïve Bayes, SVM**	**CIC-IDS2017 DDoS (Kaggle)**

Most of the studies above concentrate on the choice of classifier or on feature reduction in isolation. The present work combines hybrid resampling, wrapper-based feature selection and lightweight classifiers, and reports detection quality together with execution time.

3. PROPOSED METHODOLOGY

The workflow is shown in Fig. 1. The data are cleaned and encoded, split into training and test partitions, scaled, balanced with SMOTE-ENN (training partition only) and reduced with RFE before the classifiers are trained and evaluated on the untouched test partition. Experiments were implemented in Python using the pandas, scikit-learn and imbalanced-learn libraries.

© 2026 The Author(s) | Published by - Scholiora Research Publications
Creative Commons Attribution 4.0 International License (CC BY 4.0)

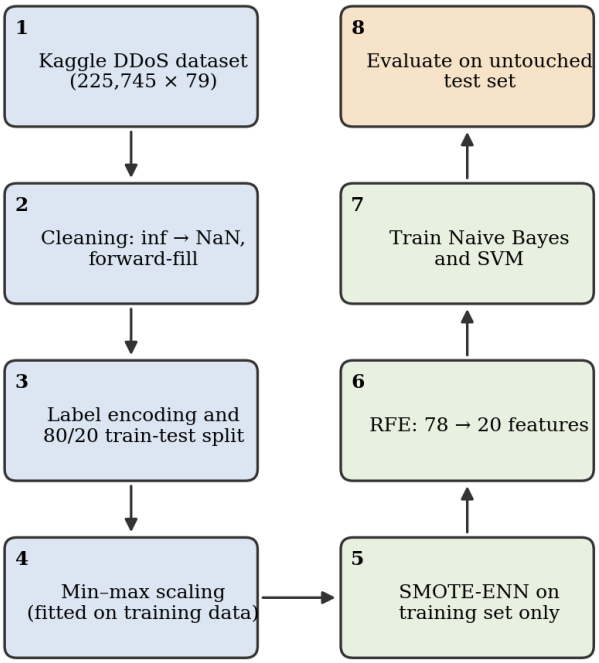


Fig. 1. Workflow of the proposed DDoS detection pipeline.

3.1 Dataset

The study uses the CIC-IDS2017 dataset [25], which contains 225,745 network-flow records, each described by 78 numerical flow features and a binary label (BENIGN or DDoS). The copy used here was obtained from Kaggle (see Data Availability). The classes are moderately imbalanced: roughly 43% of flows are benign and 57% are DDoS (Fig. 2a shows the training portion). The minority class is therefore benign traffic, which is the class that SMOTE-ENN augments.

3.2 Preprocessing

Infinite values were replaced by NaN, and missing values were forward-filled, so the data shape remained 225,745 × 79 (78 features and the label). Labels were encoded as 0 (BENIGN) and 1 (DDoS). The data were then split 80/20 into a training set of 180,596 records and a test set of 45,149 records. Features were scaled to [0, 1] by min–max normalisation so that no feature dominates because of its numeric range. The scaler was fitted on the training data and then applied to the test data, so that no test information influences preprocessing.

3.3 Class Balancing with SMOTE-ENN

SMOTE [26] creates synthetic minority samples by interpolating between a minority sample and its nearest minority neighbours, and Edited Nearest Neighbours (ENN) [27] removes samples whose class differs from that of most of their neighbours. Their combination [28] oversamples the minority class and then cleans noisy or overlapping samples near the class boundary. SMOTE-ENN was applied to the training set only, which produced 204,176 training records with an almost equal class split (Fig. 2b). The test set was left unchanged so that the reported results reflect the real class distribution and are not inflated by synthetic samples.

© 2026 The Author(s) | Published by - Scholiora Research Publications
Creative Commons Attribution 4.0 International License (CC BY 4.0)

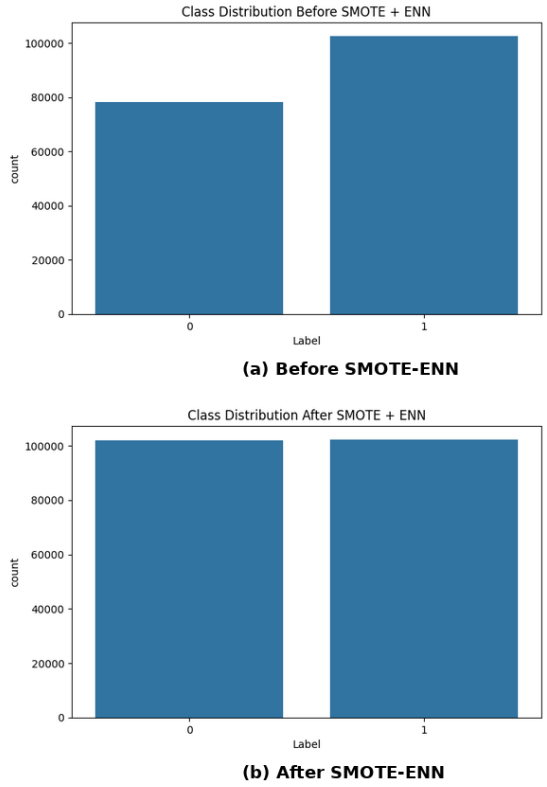


Fig. 2. Class distribution of the training data (a) before and (b) after SMOTE-ENN (0 = benign, 1 = DDoS).

3.4 Feature Selection with RFE

Recursive Feature Elimination was introduced for SVM classifiers [29]: it repeatedly fits an estimator, ranks the features by importance and discards the weakest until the requested number remains. In this work the estimator is a Random Forest [30], whose feature-importance scores provide the ranking at each elimination step, following the Random Forest-based RFE (RF-RFE) approach described by Granitto et al. [31]. RFE was run on the balanced training data, and the features with rank 1, 20 in total, were retained (Table II). They are dominated by forward-direction packet-length statistics, inter-arrival times, header lengths and segment sizes, together with the destination port and the initial TCP window size.

TABLE II. Features retained by RFE

#	Feature	#	Feature
1	Destination Port	11	Fwd Header Length
2	Total Fwd Packets	12	Bwd Header Length
3	Total Length of Fwd Packets	13	Packet Length Mean
4	Fwd Packet Length Max	14	Average Packet Size
5	Fwd Packet Length Mean	15	Avg Fwd Segment Size
6	Fwd Packet Length Std	16	Avg Bwd Segment Size
7	Bwd Packet Length Mean	17	Subflow Fwd Packets
8	Bwd Packet Length Std	18	Subflow Fwd Bytes
9	Fwd IAT Total	19	Init_Win_bytes_forward
10	Fwd IAT Std	20	act_data_pkt_fwd

© 2026 The Author(s) | Published by - Scholiora Research Publications
Creative Commons Attribution 4.0 International License (CC BY 4.0)

3.5 Classifiers

Two supervised classifiers were trained on the 20 selected features. The SVM separates the classes with a maximum-margin decision boundary and copes well with high-dimensional data. Naïve Bayes is a probabilistic classifier that assumes the features are conditionally independent given the class; it is very fast to train and to apply, and its Gaussian variant is used here because the features are continuous.

3.6 Evaluation Metrics

With DDoS as the positive class, and TP, TN, FP and FN denoting true positives, true negatives, false positives and false negatives, the following metrics were computed on the test set:

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

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

$$\text{Recall} = TP / (TP + FN) \quad (3)$$

$$\text{F1-score} = 2 \times \text{Precision} \times \text{Recall} / (\text{Precision} + \text{Recall}) \quad (4)$$

Precision, recall and F1-score in Table III are weighted averages over the two classes. The false-positive rate FPR = FP / (FP + TN) and false-negative rate FNR = FN / (FN + TP) are reported separately, together with the area under the ROC curve (AUC) and the execution time of each model.

4. RESULTS AND DISCUSSION

4.1 Overall Performance

Table III and Fig. 3 compare the two classifiers on a test set of 45,149 records. Naïve Bayes scored slightly higher than SVM, with an accuracy of 98.91% against 98.57%, and it took about half the time (1.2 s against 2.5 s). The accuracy gap of 0.34 percentage points is small. Because these results come from a single split without repeated runs or a statistical significance test, the gap should not be interpreted as a statistically established difference.

TABLE III. Model performance on the test set (%)

Model	Acc.	Prec.	Rec.	F1	Time (s)
SVM	98.57	98.59	98.57	98.56	2.5
Naïve Bayes	98.91	98.93	98.91	98.91	1.2

© 2026 The Author(s) | Published by - Scholiora Research Publications
Creative Commons Attribution 4.0 International License (CC BY 4.0)

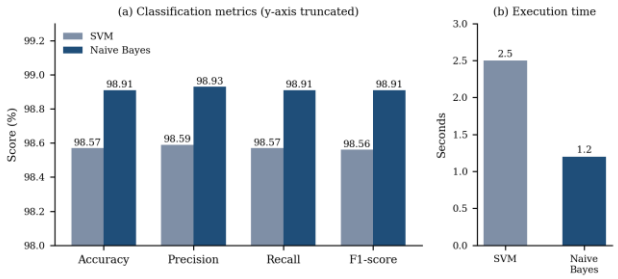


Fig. 3. Performance of SVM and Naïve Bayes: (a) classification metrics (y-axis truncated to show the difference) and (b) execution time.

4.2 Error Analysis

Fig. 4 presents the confusion matrices, which show how the two models differ in their classification outcomes. Naïve Bayes did not miss a single attack flow (FN = 0) but flagged 492 benign flows as DDoS (FPR 2.52%). SVM missed 45 attack flows (FNR 0.18%) and raised 602 false alarms (FPR 3.08%). For DDoS flows, precision was 98.11% for Naïve Bayes and 97.70% for SVM. Naïve Bayes therefore had fewer errors than SVM on this particular split, but the differences are small and were not tested for statistical significance.

TABLE IV. Confusion-matrix counts and error rates

Model	TN	FP	FN	TP	FPR	FNR
SVM	18,942	602	45	25,560	3.08%	0.18%
Naïve Bayes	19,052	492	0	25,605	2.52%	0.00%

© 2026 The Author(s) | Published by - Scholiora Research Publications
Creative Commons Attribution 4.0 International License (CC BY 4.0)

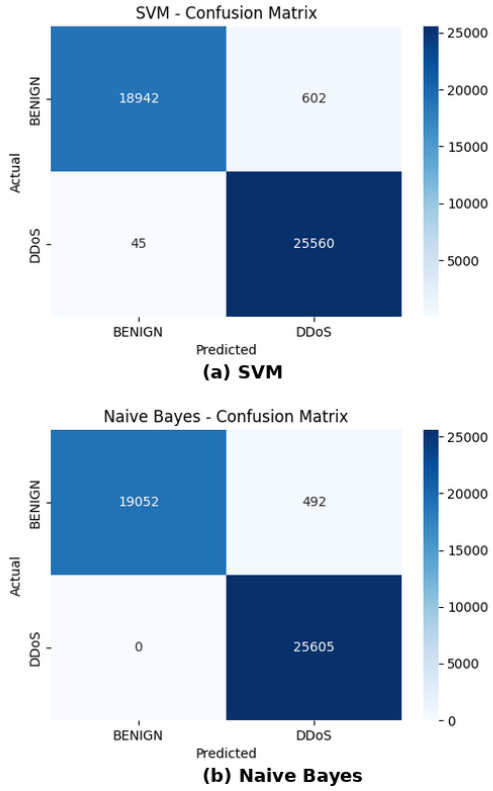


Fig. 4. Confusion matrices on the test set for (a) SVM and (b) Naïve Bayes.

In operational terms, a false-negative rate of zero is attractive, but a 2.5% false-positive rate on benign traffic means that some legitimate users could be filtered out. The acceptable balance depends on the deployment, and a decision threshold or a downstream verification stage could be tuned to it.

4.3 ROC Analysis

Both ROC curves (Fig. 5) lie close to the top-left corner and both give an AUC of 0.99, so the two models rank benign and DDoS flows almost equally well. An AUC of 0.99 indicates excellent separability, not error-free detection: the confusion matrices show that each model still makes errors at its operating point. The SVM curve rises more gradually at low false-positive rates, whereas the Naïve Bayes curve reaches a true-positive rate close to 1.0 earlier.

© 2026 The Author(s) | Published by - Scholiora Research Publications
Creative Commons Attribution 4.0 International License (CC BY 4.0)

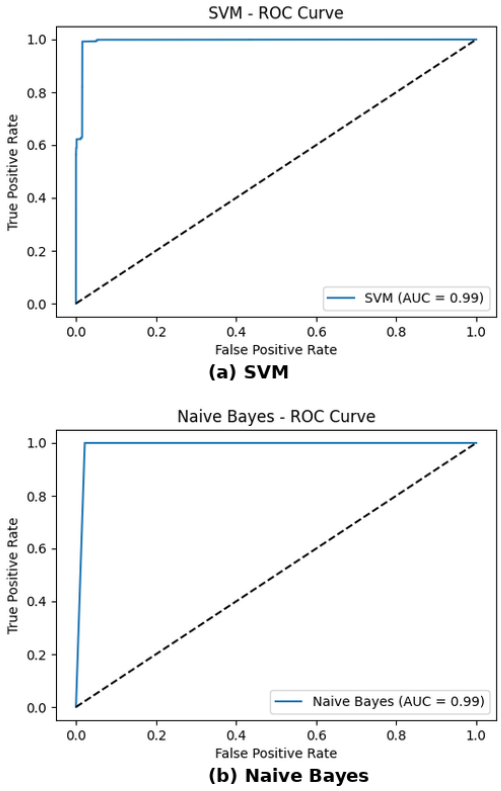


Fig. 5. ROC curves for (a) SVM and (b) Naïve Bayes (AUC = 0.99 for both).

4.4 Discussion

On this dataset and split, Naïve Bayes combined the highest detection scores with the shortest execution time, which suggests that it is a suitable lightweight baseline for near-real-time detection, although this has not been confirmed on other data. A plausible, but untested, explanation is that the balanced, low-dimensional feature space suits its simple probabilistic model, and that the cost of SVM training and prediction grows with the number of samples. Establishing this would require an ablation study that varies resampling, feature selection and classifier independently. The ablation study was not conducted in this work, so the individual contribution of RFE and SMOTE-EEN to the final performance remain unquantified. Final result observed the benefits of each of the technique, but don't count their individual effectiveness to final outcome. Likewise, showing that the difference between the two classifiers is robust would require repeated runs or cross-validation together with a paired significance test.

5. LIMITATIONS

The findings of this work should be considered with the following limitations. (i) The evaluation was conducted on a single dataset in a controlled environment that may not fully reflect the characteristics of real-world production traffic. (ii) The task is binary (benign vs. DDoS) and does not identify attack types. (iii) A single 80/20 split was used, with no cross-validation, repeated runs, confidence intervals or significance tests, so the differences between the two classifiers cannot be regarded as statistically established. (iv) Accuracy above 98% for both models suggests that the classes are highly separable in this dataset, which leaves little room to distinguish the methods. (v) There is no ablation of SMOTE-ENN or RFE, and no hyperparameter tuning is reported. (vi) Robustness to adaptive or adversarial attackers was not evaluated, and execution times depend on the hardware and implementation.

© 2026 The Author(s) | Published by - Scholiora Research Publications
Creative Commons Attribution 4.0 International License (CC BY 4.0)

6. CONCLUSION AND FUTURE WORK

This paper presented a DDoS detection pipeline that combines min–max normalisation, SMOTE-ENN balancing of the training data, and Random Forest-based RFE feature selection (78 to 20 features) with Naïve Bayes and SVM classifiers. On a held-out test set of 45,149 flows, Naïve Bayes reached 98.91% accuracy, a 98.91% F1-score, no missed attacks and the shortest execution time (1.2 s), while SVM reached 98.57% accuracy in 2.5 s. Both achieved an AUC of 0.99. Because these results come from one dataset and one train–test split without significance testing, the differences between the classifiers are indicative only and do not establish that Naïve Bayes is superior to SVM. The results suggest that a simple model on a balanced, reduced feature set can be an accurate and inexpensive option for DDoS detection on this dataset.

Future work will (i) validate the pipeline on additional and more recent datasets using cross-validation or repeated runs; (ii) add ablation studies and statistical tests; (iii) extend the task to multi-class attack identification; (iv) evaluate deep-learning and Transformer-based detectors; (v) use explainability tools such as SHAP and LIME to interpret model decisions; and (vi) test real-time deployment and robustness against adaptive attackers.

DATA AVAILABILITY

The dataset analysed in this study is publicly available from the Canadian Institute for Cybersecurity at https://www.unb.ca/cic/datasets/ids-2017.html. The copy used here was downloaded from Kaggle: https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset

CONFLICT OF INTEREST STATEMENT

Author declare that no conflict of interests exists in the paper.

ACKNOWLEDGMENTS

Authors declare that no financial support received for this research work.

REFERENCES:

1. M. Najafimehr, S. Zarifzadeh, and S. Mostafavi, “DDoS attacks and machine-learning-based detection methods: A survey and taxonomy,” Eng. Rep., vol. 5, no. 12, 2023, Art. no. e12697. DOI: https://doi.org/10.1002/eng2.12697
2. M. A. Al-Shareeda, S. Manickam, and M. A. Saare, “DDoS attacks detection using machine learning and deep learning techniques: Analysis and comparison,” Bull. Electr. Eng. Informatics, vol. 12, no. 2, pp. 930–939, 2023. DOI: https://doi.org/10.11591/eei.v12i2.4466
3. S. Siva Saravanababu et al., “A DDoS attack categorization and prediction method based on machine learning,” J. Popul. Ther. Clin. Pharmacol., vol. 30, no. 9, pp. e300–e307, 2023. DOI: https://doi.org/10.47750/jptcp.2023.30.09.030
4. K. Kumari and M. Mrunalini, “Detecting Denial of Service attacks using machine learning algorithms,” J. Big Data, vol. 9, no. 1, Art. no. 56, 2022. DOI: https://doi.org/10.1186/s40537-022-00616-0
5. R. R. Sekar, A. M. Jenny, D. Sreshta, M. Vikas, D. B. N. Ajay, and M. Ganesh, “Prediction of distributed denial of service attacks in SDN using machine learning techniques,” in Proc. 3rd Int. Conf. Intell. Technol. (CONIT), 2023. DOI: https://doi.org/10.1109/CONIT59222.2023.10205887
6. V. Deepa, K. M. Sudar, and P. Deepalakshmi, “Detection of DDoS attack on SDN control plane using hybrid machine learning techniques,” in Proc. Int. Conf. Smart Syst. Inventive Technol. (ICSSIT), Tirunelveli, India, 2018, pp. 299–303. DOI: https://doi.org/10.1109/ICSSIT.2018.8748836
7. W. Alhalabi, A. Gaurav, V. Arya, I. F. Zamzami, and R. A. Aboalela, “Machine learning-based distributed denial of services (DDoS) attack detection in intelligent information

© 2026 The Author(s) | Published by - Scholiora Research Publications
Creative Commons Attribution 4.0 International License (CC BY 4.0)

systems," Int. J. Semant. Web Inf. Syst., vol. 19, no. 1, pp. 1–17, 2023. DOI: https://doi.org/10.4018/IJSWIS.327280
8. İ. Avçı and M. Koca, "Predicting DDoS attacks using machine learning algorithms in building management systems," Electronics, vol. 12, no. 19, Art. no. 4142, 2023. DOI: https://doi.org/10.3390/electronics12194142
9. S. A. Abbas and M. S. Almhanna, "Distributed denial of service attacks detection system by machine learning based on dimensionality reduction," J. Phys.: Conf. Ser., vol. 1804, no. 1, Art. no. 012136, 2021. DOI: https://doi.org/10.1088/1742-6596/1804/1/012136
10. Y. Gu, K. Li, Z. Guo, and Y. Wang, "Semi-supervised K-means DDoS detection method using hybrid feature selection algorithm," IEEE Access, vol. 7, pp. 64351–64365, 2019. DOI: https://doi.org/10.1109/ACCESS.2019.2917532
11. H. Polat, O. Polat, and A. Cetin, "Detecting DDoS attacks in software-defined networks through feature selection methods and machine learning models," Sustainability, vol. 12, no. 3, p. 1035, 2020. DOI: https://doi.org/10.3390/su12031035
12. W. G. Gadallah, N. M. Omar, and H. M. Ibrahim, "Machine learning-based distributed denial of service attacks detection technique using new features in software-defined networks," Int. J. Comput. Netw. Inf. Secur., vol. 13, no. 3, pp. 15–27, 2021. DOI: https://doi.org/10.5815/ijcnis.2021.03.02
13. C. Tennakoon and S. Fernando, "Deep learning model for distributed denial of service (DDoS) detection," Int. J. Adv. Appl. Sci., vol. 9, no. 2, pp. 109–118, 2022. DOI: https://doi.org/10.21833/ijaas.2022.02.012
14. F. M. Aswad, A. M. S. Ahmed, N. A. M. Alhammadi, B. A. Khalaf, and S. A. Mostafa, "Deep learning in distributed denial-of-service attacks detection method for Internet of Things networks," J. Intell. Syst., vol. 32, no. 1, Art. no. 20220155, 2023. DOI: https://doi.org/10.1515/jisys-2022-0155
15. M. Najafimehr, S. Zarifzadeh, and S. Mostafavi, "A hybrid machine learning approach for detecting unprecedented DDoS attacks," J. Supercomput., vol. 78, no. 6, pp. 8106–8136, 2022. DOI: https://doi.org/10.1007/s11227-021-04253-x
16. R. Li and B. Wu, "Early detection of DDoS based on φ-entropy in SDN networks," in Proc. IEEE 4th Inf. Technol., Netw., Electron. Autom. Control Conf. (ITNEC), Chongqing, China, 2020, pp. 731–735. DOI: https://doi.org/10.1109/ITNEC48623.2020.9084885
17. A. Yadav, A. S. Kori, N. D. G, P. Shettar, and M. M. Moin, "A hybrid approach for detection of DDoS attacks using entropy and machine learning in software defined networks," in Proc. 12th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT), Kharagpur, India, 2021, pp. 1–7. DOI: https://doi.org/10.1109/ICCCNT51525.2021.9580057
18. G. Ajeetha and G. Madhu Priya, "Machine learning-based DDoS attack detection," in Proc. Innov. Power Adv. Comput. Technol. (i-PACT), Vellore, India, 2019, pp. 1–5. DOI: https://doi.org/10.1109/i-PACT44901.2019.8959961
19. P. S. Saini, S. Behal, and S. Bhatia, "Detection of DDoS attacks using machine learning algorithms," in Proc. 7th Int. Conf. Comput. Sustain. Global Dev. (INDIACom), New Delhi, India, 2020, pp. 16–21. DOI: https://doi.org/10.23919/INDIACom49435.2020.9083716
20. K. M. Sudar, M. Beulah, P. Deepalakshmi, P. Nagaraj, and P. Chinnasamy, "Detection of distributed denial of service attacks in SDN using machine learning techniques," in Proc. Int. Conf. Comput. Commun. Informat. (ICCCI), Coimbatore, India, 2021, pp. 1–5. DOI: https://doi.org/10.1109/ICCCI50826.2021.9402517
21. N. N. Tuan, P. H. Hung, N. D. Nghia, N. V. Tho, T. V. Phan, and N. H. Thanh, "A DDoS attack mitigation scheme in ISP networks using machine learning based on SDN," Electronics, vol. 9, no. 3, p. 413, 2020. DOI: https://doi.org/10.3390/electronics9030413
22. B. Al-Duwairi et al., "SIEM-based detection and mitigation of IoT-botnet DDoS attacks," Int. J. Electr. Comput. Eng., vol. 10, no. 2, pp. 2182–2191, 2020. DOI: https://doi.org/10.11591/ijece.v10i2.pp2182-2191
23. A. Kajal and S. K. Nandal, "Cyber security against DDoS, malware, spoofing attacks using machine learning with genetic algorithm," Int. J. Adv. Sci. Technol., vol. 29, no. 5, pp. 5388–5400, 2020.

© 2026 The Author(s) | Published by - Scholiora Research Publications
Creative Commons Attribution 4.0 International License (CC BY 4.0)

24. R. Jangra and A. Kajal, "A review of deep learning based intrusion detection systems," in Proc. Int. Conf. Comput., Commun., Intell. Syst. (ICCCIS), 2023, pp. 1004–1009. DOI: https://doi.org/10.1109/ICCCIS60361.2023.10425595
25. I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in Proc. 4th Int. Conf. Inf. Syst. Secur. Privacy (ICISSP), 2018, pp. 108–116. DOI: https://doi.org/10.5220/0006639801080116
26. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," J. Artif. Intell. Res., vol. 16, pp. 321–357, 2002. DOI: https://doi.org/10.1613/jair.953
27. D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited data," IEEE Trans. Syst., Man, Cybern., vol. SMC-2, no. 3, pp. 408–421, Jul. 1972. DOI: https://doi.org/10.1109/TSMC.1972.4309137
28. G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," ACM SIGKDD Explor. Newsl., vol. 6, no. 1, pp. 20–29, 2004. DOI: https://doi.org/10.1145/1007730.1007735
29. I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," Mach. Learn., vol. 46, no. 1–3, pp. 389–422, 2002. DOI: https://doi.org/10.1023/A:1012487302797
30. L. Breiman, "Random forests," Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001. DOI: https://doi.org/10.1023/A:1010933404324
31. P. M. Granitto, C. Furlanello, F. Biasioli, and F. Gasperi, "Recursive feature elimination with random forest for PTR-MS analysis of agroindustrial products," Chemom. Intell. Lab. Syst., vol. 83, no. 2, pp. 83–90, 2006. DOI: https://doi.org/10.1016/j.chemolab.2006.01.007

© 2026 The Author(s) | Published by - Scholiora Research Publications
Creative Commons Attribution 4.0 International License (CC BY 4.0)