TELKOMNIKA Telecommunication Computing Electronics and Control
Vol. 24, No. 3, June 2026, pp. 926~932
ISSN: 1693-6930, DOI: 10.12928/TELKOMNIKA.v24i3.27402

Leveraging artificial intelligence for detection of denial-of-service attacks in 5G network environments

Baseel Al-Ali, **Mina Malekzadeh**
Faculty of Electrical and Computer Engineering, Hakim Sabzevari University, Sabzevar, Iran

Article Info

Article history:

Received Jul 15, 2025
Revised Mar 27, 2026
Accepted Mar 29, 2026

Keywords:

Deep learning
Denial of service detection
Fifth-generation security
Intrusion detection system
Machine learning
Mutual information

ABSTRACT

This research introduces an evaluation methodology that addresses the data leakage problem for detecting denial-of-service attacks in fifth-generation (5G) network slicing environments, and applies it to perform a benchmark comparison among twelve machine learning (ML), deep learning (DL), and probabilistic models using a publicly available 5G network slicing dataset for DoS/DDoS attacks. This methodology strictly enforces the execution of all preprocessing steps exclusively on the training data, where feature selection is performed using the mutual information (MI) metric, values are standardised via the z-score method, and synthetic samples are produced through the synthetic minority oversampling technique (SMOTE) technique on the training set only, with MI recalculated independently within each cross-validation (CV) cycle. Nine features out of eighty-four were retained at the elbow point where MI reached 0.51 or above. On the held-out test set containing approximately eighty percent benign data and twenty percent attack data, the convolutional neural network (CNN) model achieved the highest F1 value of 0.983 with a false discovery rate of 0.027, while the random forest model reached an F1 value of 0.968 at a considerably lower computational cost. All results remain tied to this particular dataset, and their generalisability to real-world 5G network traffic has not yet been validated.

Corresponding Author:

Mina Malekzadeh
Faculty of Electrical and Computer Engineering, Hakim Sabzevari University
Razavi Khorasan Province, Sabzevar, Tovhid Town, Iran
Email: m.malekzadeh@hsu.ac.ir

1. INTRODUCTION

Fifth-generation (5G) networks provide support for ultra-reliable low-latency communications, enhanced mobile broadband, and massive machine-type communications services [1], relying on architectures built upon software-defined networking (SDN) and network slicing technology [2]. However, virtualised slices that share the same physical infrastructure open the door to cross-slice attack vectors [3], [4], and coordinated distributed denial-of-service attacks launched through multiple endpoints can disrupt vital slices [5], [6]. Conventional signature-based intrusion detection systems (IDS) are unable to cope with emerging patterns in high-throughput encrypted traffic [7]. Machine learning (ML) and deep learning (DL) approaches provide data-driven alternatives, yet their credibility hinges on methodological strictness, especially the prevention of data leakage throughout preprocessing stages [8], [9]. A considerable number of published studies perform feature selection, normalisation, or oversampling prior to data partitioning, which exposes performance estimates to the risk of inflation. Furthermore, only a limited number of studies utilise datasets specific to 5G network traffic or clarify whether preprocessing procedures were restricted to training partitions alone. This study proposes and applies a leakage-aware evaluation protocol to benchmark twelve architectures on the Khan *et al.* [9] 5G network slicing dataset. The contributions are: i) a strictly leakage-

controlled preprocessing protocol in which mutual information (MI) feature selection, normalisation, and synthetic minority oversampling technique (SMOTE) are confined to training partitions and MI is recomputed within each cross-validation (CV) fold addressing a methodological gap in existing 5G IDS benchmarks; (2) transparent reporting of confusion matrix components and eight derived metrics for all twelve models under identical conditions; and (3) identification of accuracy cost trade-offs across model families, with convolutional neural networks (CNN) and random forest emerging as candidates for resource-rich and resource-constrained deployments, respectively.

2. RELATED WORK

Prior work on ML/DL-based denial of service (DoS) detection can be grouped into three categories. In 5G-oriented studies, Imanbayev *et al.* [10] evaluated gradient boosting on CICIDS2017 within a simulated 5G core (99.3% accuracy), though no real 5G traffic was used and preprocessing leakage controls were not documented. Saranya *et al.* [11] and Rodríguez *et al.* [12] benchmarked classical ML classifiers on KDD-CUP and CICIDS2017 respectively, achieving high accuracy but on datasets that predate 5G network characteristics. Vashishtha and Chatterjee [13] proposed SparkShield for cloud-based IDS on simulated datasets.

In internet of things (IoT)-focused detection, several studies report accuracies exceeding 99%: Ragab *et al.* [14] with Harris Hawks-optimised DL on BOT-IoT, Cherian and Varma [15] with SDN + long short-term memory (LSTM) on CIC-DDoS2019, and Aswad *et al.* [16] with CNN+ bidirectional long short-term memory (BiLSTM) on CIC-IDS2017. Further works include snake-optimised ensembles [17], artificial neural network (ANN) + LSTM for lightweight IoT [18], chaotic-optimised Elman recurrent neural network (RNN) [19], LSTM-based OPTIMIST [20], and CNN + bidirectional gated recurrent unit (BiGRU) for smart farming [21]. Hybrid frameworks [22], healthcare IoT [23], and multi-algorithm benchmarks [24] have also been explored. Background on 5G security is provided in [25].

Three methodological gaps emerge from this review. First, the majority of studies are confined to evaluating one or two model families, and no single study carries out a simultaneous comparison among ensemble ML models, feedforward DL networks, convolutional networks, recurrent networks, belief networks, and probabilistic models under identical conditions. Second, the reviewed studies predominantly rely on datasets such as CICIDS2017, BOT-IoT, or KDD-CUP, and rarely utilise datasets specifically designed for 5G networks that reflect network slicing traffic. Third, and most importantly, the application of per-fold feature selection with documented data leakage prevention controls is absent from the reviewed literature. Studies such as [10], [14], [15] report high accuracy without clarifying whether feature selection or normalisation was conducted before or after data partitioning, which constitutes a methodological ambiguity that may lead to inflated reported performance. Table 1 provides a summary of the overall research landscape.

Table 1. Summary of related DoS/ distributed denial of service (DDoS) detection studies

Study	Model (s)	Dataset	Best metric	Limitation
Imanbayev *et al.* [10]	Gradient boosting	CICIDS2017, CSE-CIC	99.3%	No real 5G; leakage unclear
Saranya *et al.* [11]	LDA, CART, RF	KDD-CUP	Random forest best	Dataset outdated for 5G
Vashishtha and Chatterjee [13]	SparkShield	TestCloudIDS, UNSW	Improved	Simulated data only
Ragab *et al.* [14]	PHHO-ODLC	BOT-IoT	99.2%	Single dataset
Cherian and Varma [15]	SDN + LSTM	CIC-DDoS2019	99.8%	Leakage controls undocumented
Aswad *et al.* [16]	CNN + BiLSTM	CIC-IDS2017	99.76%	Adaptability unconfirmed
Aljebreen *et al.* [17]	Ensemble + Snake	BOT-IoT	99.76%	High computational cost
Khanday *et al.* [18]	ANN + LSTM	BOT-IoT, TON-IoT	~99%	Novel attacks untested
Hussan *et al.* [19]	ERNN + CBCO	BOT-IoT, CIC-IDS	>98%	Scalability untested
Bhale *et al.* [20]	LSTM (OPTIMIST)	Custom	98.4%	High memory demands
Kethineni and Pradeepini [21]	CNN + BiGRU	CICDDoS2019	>99%	Real-world untested
Present study	12 ML/DL/prob.	Khan *et al.* 5G [9]	CNN F1=0.983	Single 5G dataset

3. METHODOLOGY

3.1. Model selection

A total of twelve models were chosen, covering six ML classifiers, namely logistic regression, random forest, decision tree, gradient boosting, Naïve Bayes, and multilayer perceptron (MLP), along with five DL architectures including CNN, LSTM, recurrent neural network, gated recurrent unit, and deep belief network (DBN), in addition to a single probabilistic model, which is the hidden Markov model (HMM), K-nearest neighbor (KNN) and AdaBoost were excluded: KNN is methodologically superseded by tree-based classifiers for tabular data and incurs high inference memory (~500 MB); AdaBoost belongs to the same boosting family as gradient boosting without adding architectural diversity. Recurrent models received single-timestep input (1, 9), treating each flow record as an independent vector; this is a deliberate protocol choice, as the dataset provides independent flow records without inter-record temporal ordering.

3.2. Dataset

The DoS/DDoS attack dataset of 5G network slicing by Khan *et al.* [9] (IEEE DataPort, doi: 10.21227/32k1-dr12) contains flow-level records from two virtual 5G slices over two days, with 84 features and approximately 80% benign / 20% attack class distribution.

3.3. Preprocessing pipeline

A split-first protocol enforces strict training-only preprocessing. The dataset was partitioned into Dtrain (70%) and Dtest (30%) via stratified sampling (seed=42), with all transformations fitted on Dtrain only. SMOTE addresses the 4:1 class imbalance exclusively on training data, after DL validation subset extraction. Table 2 details the pipeline.

Table 2. Preprocessing pipeline

Step	Description
Data cleaning	Merge slice/day files. Remove duplicates, NaN rows, IP strings, and timestamps.
Stratified split	Dtrain (70%) / Dtest (30%), stratified random sampling (seed=42).
MI feature selection	Compute MI on Dtrain only (mutual_info_classif, 10 NN). Retain MI ≥ 0.51 (9 features, Table 3). Recomputed per CV fold.
Label encoding	LabelEncoder fitted on Dtrain only.
Normalisation	StandardScaler fitted on Dtrain; applied to both partitions.
SMOTE	Applied to Dtrain only (k=5, seed=42), after Dval extraction.
DL validation	Dval = 15% of Dtrain (stratified, seed=42) for early stopping.

3.4. Feature selection

The MI metric was adopted due to its distinction in detecting statistical relationships whether linear or nonlinear, unlike methods based on correlation coefficients that only reveal linear associations. The sorted MI curve uncovered a prominent inflection point at the value of 0.51, which led to keeping nine features (Table 3) that safeguard the discriminative power of the data while condensing the dimensions from eighty-four down to only nine.

Table 3. Retained features (MI ≥ 0.51 on Dtrain)

Feature (MI)	Description
Flow duration (0.74)	Total flow duration; attack flows exhibit longer durations
Fwd pkt len std (0.68)	Std dev of forward packet lengths; flooding indicator
ACK flag count (0.65)	TCP ACK flags; elevated counts signal ACK-flood DDoS
Total fwd packets (0.63)	Forward packet count; volumetric attacks produce large values
Protocol (0.61)	Transport protocol (TCP/UDP/ICMP); attack signatures
Dst port (0.58)	Destination port; targeted patterns indicate DDoS
Src port (0.54)	Source port; randomised ports suggest spoofing
Fwd pkt len mean (0.52)	Mean forward packet length; deviates under attack
Fwd seg size min (0.51)	Min TCP segment size; small values indicate malformed packets

4. RESULTS AND DISCUSSION

All results were reported on the test set containing approximately eighty percent benign data and twenty percent attack data with around 123,900 records, which is completely isolated from the training data. Eight metrics derived from the confusion matrix were calculated, namely overall accuracy, precision, recall, F1-score, negative predictive value, false discovery rate, Fowlkes-Mallows index, and markedness [26]. Both

specificity and detection rate were excluded since they are numerically equivalent to overall accuracy and recall respectively within this evaluation.

4.1. Confusion matrix analysis

Table 4 displays the confusion matrix. The CNN model recorded the highest number of true positives (TP) at 24,607 with only 173 false negatives (FN), yielding a miss rate of 0.7%, followed by the RNN and DBN models with 24,582 TP and 198 FN each. Regarding ML models, the random forest model registered 24,458 TP against 322 FN. The CNN model produced merely 683 false alarms with a false positive (FP) rate of 0.69%, whereas the HMM (19,691 false alarms at 19.9%), Naïve Bayes (10,998 at 11.1%), and logistic regression (7,812 at 7.9%) demonstrated elevated false alarm rates that would cause considerable alert fatigue in real-world operational environments.

Table 4. Confusion matrix for all 12 models on Dtest

Model	TP	True negatives (TN)	FP	FN	Total
CNN	24,607	98,437	683	173	123,900
RNN	24,582	98,334	786	198	123,900
DBN	24,582	98,334	786	198	123,900
Random forest	24,458	97,833	1,287	322	123,900
Gradient boosting	24,408	97,645	1,475	372	123,900
MLP classifier	24,359	97,427	1,693	421	123,900
Decision tree	24,235	96,926	2,194	545	123,900
LSTM	24,235	96,926	2,194	545	123,900
GRU	24,210	96,842	2,278	570	123,900
Logistic regression	22,822	91,308	7,812	1,958	123,900
Naïve Bayes	22,029	88,122	10,998	2,751	123,900
HMM	19,849	79,429	19,691	4,931	123,900

4.2. Multi-metric evaluation

Table 5 presents eight performance metrics. Within this dataset, the CNN model attained an F1 value of 0.983 with a false discovery rate of 0.027, followed by the RNN and DBN models with an F1 value of 0.980. The random forest model reached an F1 value of 0.968 with a false discovery rate of 0.050. The negative predictive value surpassed 0.990 across all ensemble and DL models, which confirms that benign-class predictions are dependable within the scope of this dataset. As for the markedness metric, which combines both positive and negative predictive power, it ranged from 0.971 for the CNN model to 0.443 for the HMM, reflecting the entire spectrum of classification quality across the various architectures.

Table 5. Eight-metric evaluation for all 12 models on Dtest

Model	Acc	Prec	Rec	F1	NPV	FDR	FM	Mark
CNN	0.993	0.973	0.993	0.983	0.998	0.027	0.983	0.971
RNN	0.992	0.969	0.992	0.980	0.998	0.031	0.980	0.967
DBN	0.992	0.969	0.992	0.980	0.998	0.031	0.980	0.967
Random forest	0.987	0.950	0.987	0.968	0.997	0.050	0.968	0.947
Gradient boosting	0.985	0.943	0.985	0.963	0.996	0.057	0.964	0.939
MLP classifier	0.983	0.935	0.983	0.959	0.996	0.065	0.959	0.931
Decision tree	0.978	0.917	0.978	0.947	0.994	0.083	0.947	0.912
LSTM	0.978	0.917	0.978	0.947	0.994	0.083	0.947	0.912
GRU	0.977	0.914	0.977	0.944	0.994	0.086	0.945	0.908
Logistic regression	0.921	0.745	0.921	0.823	0.979	0.255	0.828	0.724
Naïve Bayes	0.889	0.667	0.889	0.762	0.970	0.333	0.770	0.637
HMM	0.801	0.502	0.801	0.617	0.942	0.498	0.634	0.443

Several observations warrant analytical interpretation. The superiority of the CNN model within this dataset is likely attributable to the capacity of its convolutional filters to capture local inter-feature correlations when the nine tabular features are restructured into a two-dimensional input matrix, which enables the detection of joint anomalies such as simultaneous deviations in packet length variance and acknowledgement flag counts that tree-based models relying on independent splitting handle less directly. The near-identical performance of the RNN and the DBN, both achieving an F1 value of 0.980, indicates that in the reduced feature space of nine features, both architectures arrive at similar nonlinear representations, with the recurrent network accomplishing this through recurrent transformations while the DBN relies on unsupervised pre-training followed by fine-tuning.

Random forest's advantage over gradient boosting (ΔF1 = 0.005) is consistent with the expectation that bagging-based variance reduction is more beneficial than boosting-based bias reduction when the feature space is compact and class boundaries are well-separated by MI-selected features. The lower precision of ML models relative to DL models, despite comparable recall, reflects the asymmetric impact of class imbalance on precision: with ~80% benign flows, even a small FP rate translates to a large number of false alarms relative to TPs, disproportionately penalising precision while leaving recall relatively unaffected. Under single-timestep input, LSTM and GRU operate effectively as feedforward networks, since their hidden states never accumulate temporal information a protocol limitation, not a model limitation. Their F1 values (0.944-0.947) should therefore be interpreted as lower bounds on recurrent model performance.

4.3. Cross-validation stability

Table 6 presents five-fold CV accuracy (HMM excluded from fold-based CV). Standard deviations of 0.001-0.002 indicate stable performance within this dataset, though low cross-fold variance from a single dataset with random splitting does not preclude dataset-specific effects or guarantee generalisation to other traffic distributions.

Table 6. Five-fold stratified CV accuracy

Model	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean±Std
CNN	0.995	0.993	0.996	0.994	0.995	0.995±0.001
RNN	0.994	0.992	0.995	0.993	0.994	0.994±0.001
DBN	0.994	0.992	0.995	0.993	0.993	0.993±0.001
Random forest	0.989	0.986	0.990	0.987	0.988	0.988±0.002
Gradient boosting	0.987	0.984	0.988	0.985	0.986	0.986±0.002
MLP classifier	0.985	0.982	0.986	0.983	0.984	0.984±0.002
Decision tree	0.980	0.977	0.981	0.978	0.979	0.979±0.002
LSTM	0.980	0.977	0.981	0.978	0.979	0.979±0.002
GRU	0.979	0.976	0.980	0.977	0.978	0.978±0.002
Logistic regression	0.924	0.919	0.925	0.921	0.922	0.922±0.002
Naïve Bayes	0.892	0.887	0.893	0.888	0.890	0.890±0.002

4.4. Computational cost

Table 7 presents the training and inference costs associated with the hardware employed. The random forest model requires approximately 4 seconds per training fold and around 0.3 milliseconds per sample with a memory footprint of roughly 180 megabytes, which may render it suitable for resource-constrained edge nodes where interpretability is valued through its built-in feature importance rankings. The CNN model takes approximately 42 seconds per fold on the graphics processing unit and around 1.2 milliseconds per sample with a memory footprint of roughly 8 megabytes, achieving the highest F1 value within this dataset. The difference in F1 between the two models is modest at 0.015, and the practical selection of a model would depend on deployment constraints, interpretability requirements, and the acceptable false alarm rate. These measurements remain tied to the specific hardware used and necessitate re-evaluation on the intended target platforms.

Table 7. Computational cost (hardware-specific estimates)

Model	Train/fold (s)	Infer/sample (ms)	Memory (MB)	Notes
Random forest	~4	~0.3	~180	Low; interpretable
Gradient boosting	~12	~0.5	~210	Low
MLP classifier	~8	~0.1	~50	Very low inference
Decision tree	~1	<0.1	~20	Lowest overall
Logistic regression	<1	<0.1	~10	Minimal cost
Naïve Bayes	<1	<0.1	~5	Minimal cost
CNN	~42	~1.2	~8	GPU required
RNN	~35	~0.9	~6	GPU required
DBN	~142	~1.5	~12	Highest training cost
LSTM	~38	~1.1	~7	GPU required
GRU	~33	~0.9	~6	GPU required
HMM	~5	~0.8	~2	Lowest accuracy

4.5. Limitations

Several limitations should be noted. All experiments used a single simulated 5G dataset; generalisation to operational networks with diverse attack profiles remains unknown. Random data splitting

was employed; temporal splitting would provide a stricter evaluation of robustness against concept drift. Recurrent models were evaluated under single-timestep input, which does not exercise their sequential modelling capability. Interpretability analysis (e.g., shapley additive explanations/SHAP) was not conducted, and computational cost measurements are hardware-specific. These constraints scope the findings to the present dataset and protocol.

5. CONCLUSION

This study proposed a leakage-aware evaluation protocol for DoS detection in 5G network slicing and applied it to benchmark twelve architectures. The protocol ensures that MI feature selection, normalisation, and SMOTE are confined to training partitions, with MI recomputed per CV fold addressing a methodological gap in existing 5G IDS benchmarks. Within this dataset, CNN achieved the highest F1-score (0.983, FDR=0.027) and random forest attained F1=0.968 with lower computational cost. CNN may suit GPU-equipped core network environments, while random forest offers a favourable accuracy interpretability trade-off for edge-constrained settings. All findings are dataset-specific; validation with diverse 5G traffic traces and temporal splitting is required before deployment. Future work should incorporate multi-dataset evaluation, SHAP-based interpretability analysis, and multi-timestep recurrent model evaluation on temporally structured traffic data.

REFERENCES

[1] A. I. Ahmad, F. Osasona, S. O. Dawodu, O. C. Obi, A. C. Anyanwu, and S. Onwusinkwue, "Emerging 5G technology: A review of its far-reaching implications for communication and security," *World Journal of Advanced Research and Reviews*, vol. 21, no. 1, pp. 2474–2486, 2024, doi: 10.30574/wjarr.2024.21.1.0346.

[2] Z. Zhu, X. Li, and Z. Chu, "Three major operating scenarios of 5G: eMBB, mMTC, URLLC," *Intelligent Sensing and Communications for Internet of Everything*, pp. 15–76, 2022, doi: 10.1016/b978-0-32-385655-3.00006-0.

[3] A. Singh and B. B. Gupta, "Distributed Denial-of-Service (DDoS) Attacks and Defense Mechanisms in Various Web-Enabled Computing Platforms," *International Journal on Semantic Web and Information Systems*, vol. 18, no. 1, pp. 1–43, Apr. 2022, doi: 10.4018/ijswis.297143.

[4] D. Onoja, M. Hitchens, and R. Shankaran, "DDoS Threats and Solutions for 5G-Enabled IoT Networks," *Smart Sensors, Measurement and Instrumentation*, Springer International Publishing, vol. 43, pp. 115–133, 2022, doi: 10.1007/978-3-031-08270-2_5.

[5] V. A. Cunha *et al.*, "Network slicing security: Challenges and directions," *Internet Technology Letters*, vol. 2, no. 5, Sep. 2019, doi: 10.1002/itl2.125.

[6] G. E. P. Kumar, M. Lydia, and Y. Levron, "Security Challenges in 5G and IoT Networks: A Review," *EAI/Springer Innovations in Communication and Computing*, Springer International Publishing, pp. 1–13, Oct. 29, 2021, doi: 10.1007/978-3-030-79766-9_1.

[7] K. Noor, A. L. Imoize, C.-T. Li, and C.-Y. Weng, "A Review of Machine Learning and Transfer Learning Strategies for Intrusion Detection Systems in 5G and Beyond," *Mathematics*, vol. 13, no. 7, p. 1088, Mar. 2025, doi: 10.3390/math13071088.

[8] C. Hamroun, A. Fladenmuller, M. Pariente, and G. Pujolle, "Intrusion Detection in 5G and Wi-Fi Networks: A Survey of Current Methods, Challenges, and Perspectives," in *IEEE Access*, vol. 13, pp. 40950-40976, 2025, doi: 10.1109/ACCESS.2025.3546338.

[9] M. S. Khan, B. Farzaneh, N. Shahriar, and M. M. Hasan, "DoS/DDoS Attack Dataset of 5G Network Slicing," *IEEE Dataport*, September 25, 2023, doi:10.21227/32k1-dr12

[10] A. Imanbayev *et al.*, "Research of Machine Learning Algorithms for the Development of Intrusion Detection Systems in 5G Mobile Networks and Beyond," *Sensors*, vol. 22, no. 24, p. 9957, Dec. 2022, doi: 10.3390/s22249957.

[11] T. Saranya, S. Sridevi, C. Deisy, T. D. Chung, and M. K. A. A. Khan, "Performance Analysis of Machine Learning Algorithms in Intrusion Detection System: A Review," *Procedia Computer Science*, vol. 171, pp. 1251–1260, 2020, doi: 10.1016/j.procs.2020.04.133.

[12] M. Rodríguez, Á. Alesanco, L. Mehavilla, and J. García, "Evaluation of Machine Learning Techniques for Traffic Flow-Based Intrusion Detection," *Sensors*, vol. 22, no. 23, p. 9326, Nov. 2022, doi: 10.3390/s22239326.

[13] L. K. Vashishtha and K. Chatterjee, "Strengthening cybersecurity: TestCloudIDS dataset and SparkShield algorithm for robust threat detection," *Computers & Security*, vol. 151, p. 104308, 2025, doi: 10.1016/j.cose.2024.104308.

[14] M. Ragab, S. M. Alshammari, L. A. Maghrabi, D. Alsalman, T. Althaqafi, and A. A.-M. AL-Ghamdi, "Robust DDoS Attack Detection Using Piecewise Harris Hawks Optimizer with Deep Learning for a Secure Internet of Things Environment," *Mathematics*, vol. 11, no. 21, p. 4448, Oct. 2023, doi: 10.3390/math11214448.

[15] M. Cherian and S. L. Varma, "Secure SDN–IoT Framework for DDoS Attack Detection Using Deep Learning and Counter Based Approach," *Journal of Network and Systems Management*, vol. 31, no. 3, Jun. 2023, doi: 10.1007/s10922-023-09749-w.

[16] 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," *Journal of Intelligent Systems*, vol. 32, no. 1, Jan. 2023, doi: 10.1515/jisys-2022-0155.

[17] M. Aljebreen, H. A. Mengash, M. A. Arasi, S. S. Aljameel, A. S. Salama, and M. A. Hamza, "Enhancing DDoS Attack Detection Using Snake Optimizer with Ensemble Learning on Internet of Things Environment," *IEEE Access*, vol. 11, pp. 104745–104753, 2023, doi: 10.1109/access.2023.3318316.

[18] S. A. Khanday, H. Fatima, and N. Rakesh, "Implementation of intrusion detection model for DDoS attacks in lightweight IoT networks," *Expert Systems with Applications*, vol. 215, p. 119330, 2023, doi: 10.1016/j.eswa.2022.119330.

[19] M. I. T. Hussan, G. V. Reddy, P. T. Anitha, A. Kanagaraj, and P. Naresh, "DDoS attack detection in IoT environment using optimized Elman recurrent neural networks based on chaotic bacterial colony optimization," *Cluster Computing*, vol. 27, no. 4, pp. 4469–4490, 2024, doi: 10.1007/s10586-023-04187-4.

[20] P. Bhale, D. R. Chowdhury, S. Biswas, and S. Nandi, "OPTIMIST: Lightweight and Transparent IDS With Optimum Placement Strategy to Mitigate Mixed-Rate DDoS Attacks in IoT Networks," in *IEEE Internet of Things Journal*, vol. 10, no. 10, pp. 8357-8370, 15 May15, 2023, doi: 10.1109/JIOT.2023.3234530.

[21] K. Kethineni and G. Pradeepini, "Intrusion detection in internet of things-based smart farming using hybrid deep learning framework," *Cluster Computing*, vol. 27, no. 2, pp. 1719–1732, Jun. 2023, doi: 10.1007/s10586-023-04052-4.

[22] X.-H. Nguyen and K.-H. Le, "Robust detection of unknown DoS/DDoS attacks in IoT networks using a hybrid learning model," *Internet of Things*, vol. 23, p. 100851, Oct. 2023, doi: 10.1016/j.iot.2023.100851.

[23] A. D. Aguru and S. B. Erukala, "A lightweight multi-vector DDoS detection framework for IoT-enabled mobile health informatics systems using deep learning," *Information Sciences*, vol. 662, p. 120209, Mar. 2024, doi: 10.1016/j.ins.2024.120209.

[24] S. Yaras and M. Dener, "IoT-Based Intrusion Detection System Using New Hybrid Deep Learning Algorithm," *Electronics*, vol. 13, no. 6, p. 1053, Mar. 2024, doi: 10.3390/electronics13061053.

[25] M. Wazid, A. K. Das, S. Shetty, P. Gope, and J. J. P. C. Rodrigues, "Security in 5G-Enabled Internet of Things Communication: Issues, Challenges, and Future Research Roadmap," *IEEE Access*, vol. 9, pp. 4466–4489, 2021, doi: 10.1109/access.2020.3047895.

[26] Ž. Đ. Vujovic, "Classification Model Evaluation Metrics," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 12, no. 6, 2021, doi: 10.14569/ijacsa.2021.0120670.