Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 8, pp. 3085–3093
DOI : 10.47974/JDMSC-2693

Discrete mathematical modeling–driven machine learning framework for secure encrypted DDoS attack detection in cloud and edge computing

Manmohan Sharma [@]
Department of Computer Science & Engineering
Faculty of Science, Technology and Architecture (FoSTA)
Manipal University Jaipur
Jaipur 303007
Rajasthan
India

Anudeep Arora [†]
Department of Management
New Delhi Institute of Management (NDIM)
Guru Gobind Singh Indraprastha University
Tughlakabad
New Delhi 110062
India

Priya Mathur [§]
Department of Mathematics
Poornima Institute of Engineering & Technology
Jaipur 302022
Rajasthan
India

Saloni Bhushan [‡]
Department of Computer Application
School of Humanities & Sciences
D Y Patil Deemed to be University
Nerul
Navi Mumbai 400706
Maharashtra
India

[@] *E-mail:* manmohan.sharma@jaipur.manipal.edu
[†] *E-mail:* arora.anudeep85@gmail.com
[§] *E-mail:* priya.mathur@poornima.org
[‡] *E-mail:* drsaloni.bhushan@gmail.com

Amit Kumar Gupta *
Department of Computer Science & Engineering
Faculty of Science, Technology and Architecture (FoSTA)
Manipal University Jaipur
Jaipur 303007
Rajasthan
India

Abstract

Distributed denial-of-service (DDoS) attacks causes significant threats in modern cloud and edge computing because of complex traffic patterns that intricate attack detection. Traditional methods have limitations in distinguishing attack strategies and extracting relevant information. This study retains machine learning and deep learning techniques to find DDoS attacks using the CICDDoS-2019 dataset, which has pre-processed to include 431,371 records and 78 features. The preprocessing intricate handling duplicates, missing values, and class imbalances using SMOTE. Various classifiers, including Logistic Regression, Decision Tree, Random Forest, XGBoost, and LightGBM, alongside One-Class SVM and Autoencoder-Bidirectional LSTM (AE–BiLSTM) models, were utilized. Assessments of metrics such as accuracy, precision, recall, F1-score, and ROC–AUC were employed to assess the models. Results indicated that ensemble methods like XGBoost and LightGBM achieved perfect ROC–AUC scores of 1.00, while AE–BiLSTM shows strong anomaly detection with a ROC–AUC of 0.9979. This framework gives a practical solution for improving network security within cloud and edge computing environments.

Subject Classification: *Primary 94A60, Secondary 94A62.*

Keywords: *DDoS attack, Security, Machine learning, Deep learning, Anomaly detection, Edge computing.*

1. Introduction

The quick growth of cloud and edge computing infrastructures is preminent to increased cyberthreats, particularly DDoS attacks, which exploit network heterogeneity and overwhelm services with vicious traffic. Traditional signature-based intrusion detection systems stumble to identify these attacks due to their size and generation pattern. Recent studies indicate that both supervised and unsupervised learning methods can achieve dominant detection accuracy on network traffic flow datasets, with the CICDDoS-2019 dataset proving valuable due to its diverse attack scenarios [1], [2]. However, issues such as class imbalance and feature redundancy in publicly available datasets can bias machine learning models. While supervised methods perform well with labeled data, they may struggle with unseen attack patterns. Anomaly detection methods can learn normal traffic behavior, enabling the identification of minor deviations, thus enhancing intrusion detection. This study proposes a robust

* *E-mail:* **amit.gupta@jaipur.manipal.edu (Corresponding Author)**

intrusion detection framework that integrates classical machine learning classifiers with anomaly-based deep learning models using a cleaned and balanced version of the CICDDoS-2019 dataset, effectively addressing data imbalance and feature dependency challenges for cloud and edge security monitoring.

2. Related Work

The study explores various machine learning methods for intrusion detection, highlighting the effectiveness of decision trees, support vector machines, and logistic regression in identifying anomalous network behavior [3]. It emphasizes the benefits of Random Forests in preventing overfitting and improving generalization [4, 5, 6]. Boosting techniques like XGBoost and LightGBM are noted for enhancing detection accuracy to approximately 99%, especially with CICDDoS-derived datasets [7, 8]. Addressing class imbalance, it advocates for Synthetic Minority Over-sampling Technique (SMOTE) as a preferred method [9]. One-Class SVM is effective for lower traffic volumes, but its performance declines in high-dimensional data [10]. The integration of autoencoders with recurrent neural networks further improves detection capabilities [11, 12, 13, 14, 15]. The research underlines the necessity for threshold-independent metrics and comprehensive error analyses, presenting an integrated framework to evaluate both supervised and anomaly-based models in DDoS detection.

3. Methodology

This section describes the methodology for developing and evaluating a machine learning-based intrusion detection framework. It includes dataset description, preprocessing steps, balancing of minority classes using SMOTE, feature analysis, and the implementation of supervised and anomaly-based learning methods. The methodology addresses data imbalance, feature dependency, and the heterogeneous nature of DDoS traffic. It proposes a hybridization of classical and deep learning techniques to enhance anomaly detection in cloud and edge computing environments, ensuring reproducibility, robustness, and fair performance assessment under real-time network conditions.

3.1 *Dataset Description and Preprocessing*

The research utilizes a publicly available distributed denial-of-service (DDoS) traffic dataset from Kaggle, derived from the CICDDoS-2019 benchmark. The original dataset consists of multiple files corresponding to various DDoS traffic patterns, which were combined into a single dataset containing 431,371 network flow records with 78 features, saved in CSV format. Duplicate entries and records with missing values were removed to ensure data reliability, along with certain less-correlated features to reduce noise. The dataset was converted into a binary classification problem by grouping all attack categories under a single Attack label and categorizing normal traffic as Benign. The resulting dataset presented a class imbalance with 323,143 attack samples

and 94,704 benign samples, leading to an imbalance ratio of 0.2931, which may negatively impact model performance.

3.2 *Correlation Analysis and Feature Dependency*

To understand the interdependencies in the dataset, a Pearson correlation matrix was created as shown in Figure 1, revealing pairs of features with high correlations (e.g., Flow Duration with Fwd IAT Total and Packet Length Min with Fwd Packet Length Min), indicating strong linear relationships in flow-based network datasets. The analysis suggests retaining these features to maintain domain-specific information, enabling ensemble and deep learning models to handle redundancy through internal feature weighting. This approach supports the research's interest in using tree-based and neural models that are robust to multicollinearity in high-dimensional trafficdata.

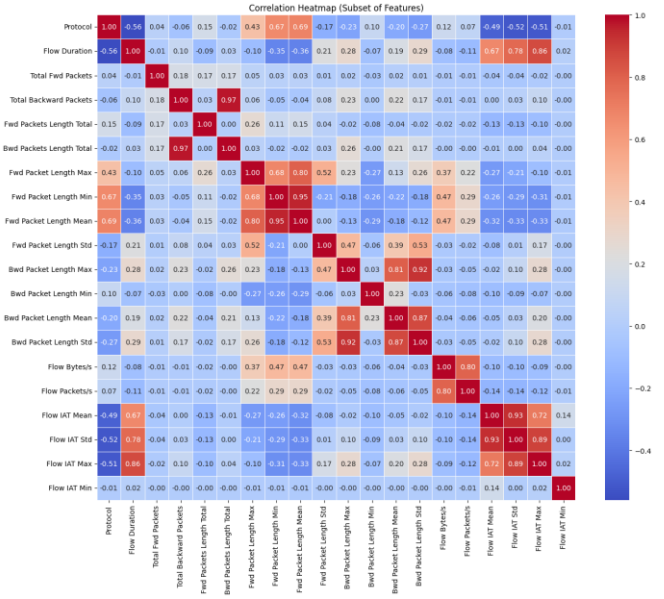

Figure 1
Correlation matrix for features of CICDDoS-2019 dataset

3.3 *Class Balancing Using SMOTE*

SMOTE mitigates class imbalance via synthetic minority sampling. Samples are interpolated between nearest minority neighbors [8]. Given a minority sample x_i and one of its nearest neighbors x_j, a synthetic instance is created as:

$$x_{\text{new}} = x_i + \lambda(x_j - x_i), \lambda \in (0,1) \quad (1)$$

After applying SMOTE, the dataset achieved perfect class balance, with 323,143 samples each for Attack and Benign classes. The Figure 2 shows

misbalancing before applying the SMOTE and balancing after the applying SMOTE of the data sample.

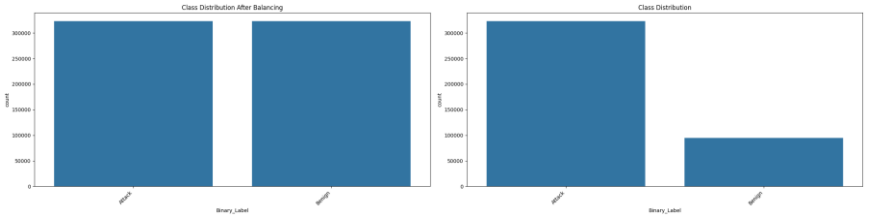

Figure 2
Dataset balancing using SMOTE

3.4 *Feature Scaling and Data Partitioning*

All numeric features were scaled using z-score normalization:

$$x' = \frac{x-\mu}{\sigma} \tag{2}$$

μ and σ represent each feature's mean and standard deviation. Data were split using stratified sampling to preserve class balance.

3.5 *Machine Learning Models*

3.5.1 Logistic Regression

Logistic Regression served as the baseline linear classifier. It estimates attack probability using the sigmoid function:

$$P(y = 1 \mid x) = \frac{1}{1+e^{-(\mathbf{w}^{\mathrm{T}}\mathbf{x}+b)}} \tag{3}$$

Parameters are optimized via binary cross-entropy minimization. Logistic Regression offers interpretability and a performance baseline.

3.5.2 Decision Tree

The Decision Tree recursively partitions the feature space. Splits are selected by minimizing Gini impurity.:

$$G = 1 - \sum_{k=1}^{K} p_k^2 \tag{4}$$

where p_k represents the proportion of samples belonging to class k. Decision Trees capture nonlinear relationships and decision boundaries in network traffic data .

3.5.3 Random Forest

Random Forest ensembles multiple bootstrap-trained decision trees. Final predictions are produced through majority voting:

$$\hat{y} = \text{mode}\{h_1(x), h_2(x), \ldots, h_T(x)\} \tag{5}$$

This ensemble reduces variance and improves generalization. It is effective for intrusion detection tasks [4].

3.5.4 XGBoost and LightGBM

Gradient boosting models enhanced predictive performance. They build additive trees via regularized loss minimization.:

$$\mathcal{L} = \sum_{i=1}^{n} \ell(y_i, \hat{y}_i) + \sum_k \Omega(f_k) \quad (6)$$

where $\ell(\cdot)$ is the loss function and $\Omega(\cdot)$ penalizes model complexity. LightGBM introduces histogram-based splitting to improve computational efficiency on large datasets [5-6].

3.6 *Anomaly Detection Models*

3.6.1 One-Class Support Vector Machine

One-Class SVM modeled normal traffic behavior. It learns a boundary enclosing most benign samples.:

$$\min_{\mathbf{w},\xi,\rho} \frac{1}{2} \| \mathbf{w} \|^2 + \frac{1}{vn} \sum_{i=1}^{n} \xi_i - \rho \quad (7)$$

subject to $\mathbf{w}^\top \phi(x_i) \geq \rho - \xi_i$. Samples falling outside this boundary are classified as anomalies [9].

4. Results and Discussion

This section discusses the experimental results and performance is measured through standard classification metrics and ROC–AUC analysis to fully assess detection capabilities across different decision thresholds.

4.1 *Overall Classification Performance*

Table 1 presents the performance metrics. Ensemble learning outperforms linear and single-tree classifiers in network traffic data analysis. Logistic Regression achieves 99.67% accuracy with high recall (0.9976) but lower precision, indicating some false alarms. The Decision Tree model exhibits 99.94% accuracy with balanced precision and recall, though its sensitivity to data variations may affect generalization.

4.2 *Performance of Ensemble Learning Models*

Random Forest, XGBoost, and LightGBM demonstrate high performance in classification tasks. Random Forest achieves an accuracy of 99.94% through ensemble averaging, which enhances robustness. XGBoost and LightGBM attain even higher accuracies of 99.96% and 99.97%, respectively, while also exhibiting the lowest error metrics (MSE and RMSE) and highest R^2 scores, indicating their predictive stability. Their performance benefits from iterative boosting mechanisms that refine decision boundaries. Among these, LightGBM stands out for its combination of accuracy and computational efficiency.

4.3 *Anomaly Detection Models*

The One-Class SVM achieves an accuracy of 91.85% and an R^2 value of 0.67, but its lower recall indicates it misses some attack instances, highlighting limitations of boundary-based unsupervised methods in high-dimensional spaces. In contrast, the Autoencoder–BiLSTM (AE–BiLSTM) model shows superior anomaly detection with 98.45% accuracy and 99.91% recall, crucial for reducing the risk of unnoticed attacks.

4.4 *ROC–AUC Analysis*

To validate detection robustness, ROC curves were plotted for various models shown in Figure 3, calculating their AUC values. The Logistic Regression model achieved an AUC of 0.992, while the Decision Tree reached 0.998. Ensemble models like Random Forest, XGBoost, and LightGBM attained perfect AUC values of 1.00, showcasing their effectiveness in intrusion detection. The AE–BiLSTM model recorded an AUC of 0.9979, indicating strong anomaly detection capabilities. Ensemble models, especially LightGBM and XGBoost, provide superior detection accuracy, while the AE–BiLSTM excels in adaptive security monitoring, making it fit for dynamic environments. The findings confirm that the proposed framework demonstrates high ROC–AUC values and robustness for network traffic analysis.

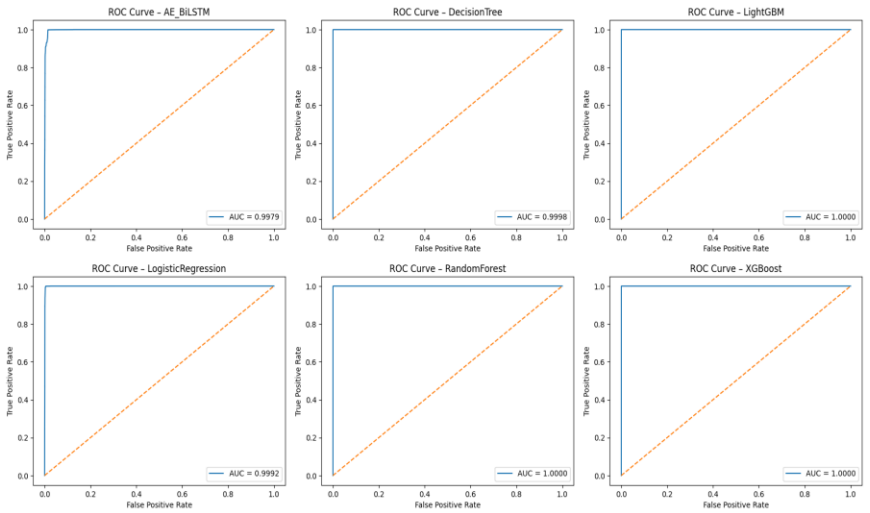

Figure 3
ROC Curve for each Evaluated Model

Table 1
Results Summary for each Evaluated Model

Model	Accuracy	Precision	Recall	F1	MSE	MAE	RMSE	R2
Logistic Regression	0.9967	0.9959	0.9976	0.9967	0.0033	0.0033	0.0571	0.9870
Decision Tree	0.9994	0.9994	0.9994	0.9994	0.0006	0.0006	0.0253	0.9974
RandomForest	0.9994	0.9991	0.9998	0.9994	0.0006	0.0006	0.0241	0.9977
XGBoost	0.9996	0.9995	0.9997	0.9996	0.0004	0.0004	0.0193	0.9985
LightGBM	0.9997	0.9995	0.9998	0.9997	0.0003	0.0003	0.0183	0.9987
OneClass-SVM	0.9185	0.9458	0.8878	0.9159	0.0815	0.0815	0.2855	0.6739
AE_BiLSTM	0.9845	0.9708	0.9991	0.9848	0.0155	0.0155	0.1243	0.9382

5. Conclusion

This research identifies intrusion in real-time traffic data within cloud edge computing using a machine learning framework. A publicly available DDoS dataset from Kaggle, which includes various attack types, was combined into a binary classifier format. Data preprocessing involved cleaning, removing duplicates and null values, and class balancing via SMOTE to ensure reliable model training. Experimental analysis showed that both supervised and anomaly-based learning methods effectively differentiate between attack and non-attack traffic. Ensemble classifiers, LightGBM and XGBoost, achieved 99% accuracy in detecting anomalies, while Random Forests also performed well. Logistic Regression and Decision Tree models provided decent interpretability but were less accurate than boosted ensembles. The proposed Autoencoder–BiLSTM framework excelled in anomaly detection, boasting high recall and ROC–AUC values due to its ability to learn traffic patterns. In contrast, One-Class SVM had lower accuracy, highlighting the limitations of boundary-based methods in complex traffic scenarios. The study concludes that combining ensemble and deep learning techniques offers a robust security solution for cloud and edge computing. Future work may expand the framework to multi-class classification, lightweight models for real-time deployment, and explainable AI to enhance transparency in intrusion detection.

References

[1] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1153–1176 (2016).

[2] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. Int. Conf. Information Systems Security and Privacy* (ICISSP), pp. 108–116 (2018).

[3] W. Lee and S. J. Stolfo, "A framework for constructing features and models for intrusion detection systems," *ACM Trans. Inf. Syst. Secur.*, vol. 3, no. 4, pp. 227–261 (Nov 2000).

[4] L. Breiman, "Random forests," *Mach. Lear*n., vol. 45, no. 1, pp. 5–32 (2001).

[5] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, San Francisco, CA, USA, pp. 785–794 (2016), doi: 10.1145/2939672.2939785.

[6] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree,"

in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, pp. 3146–3154 (2017).

[7] M. Hasan, M. M. Islam, M. A. Razzaque, and M. M. Hassan, "DDoS attack detection using ensemble machine learning," *Comput. Secur.*, vol. 113, Art. no. 102549 (2022).

[8] 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).

[9] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Comput.*, vol. 13, no. 7, pp. 1443–1471 (2001).

[10] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507 (2006).

[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780 (1997).

[12] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874 (2006).

[13] A. Noonia, D. Thakral, P. Mathur, F. Sheth, H. Shaikh, and A. K. Gupta, "A discrete mathematical model and cryptography for secure medical image analysis: Encrypted chest X-ray classification," *J. Discrete Math. Sci. Cryptogr.*, vol. 28, no. 5-A, pp. 1473–1486 (2025).

[14] D. Goyal, A. Kumar, P. Mathur, F. Sheth, and A. K. Gupta, "Robust cybersecurity through discrete and large language models for effective phishing attack detection," *J. Discrete Math. Sci. Cryptogr.*, vol. 28, no. 5-A, pp. 1621–1638 (2025).

[15] D. Goyal, F. Sheth, P. Mathur, and A. K. Gupta, "Discrete mathematical models for enhancing cybersecurity: A mathematical and statistical analysis of machine learning approaches in phishing attack detection," *J. Discrete Math. Sci. Cryptogr.*, vol. 27, no. 2-B, pp. 569–599 (2024).

Received December, 2025