Grenze International Journal of Engineering and Technology, January Issue

Stratified Meta-Learning for DDoS Attack Classification and Counteraction using Rate Limiting

B. Buvaneswari[1] and C. Senthilkumar[2]
[1]Department of Electronics and Communication Engineering, SRM Madurai College for Engineering and Technology, India
Email: buvaneswari@srmmcet.edu.in
[2]Department of Computer Science and Engineering, Thiagarajar College of Engineering, Madurai, India
Email: cskcse@tce.edu

***Abstract*— Network infrastructure encounters severe threats from Distributed Denial of Service (DDoS) attacks that become more complex and occur more frequently, resulting in service interruptions together with financial burdens and compromised cybersecurity. Traditional IDS operating independently with machine learning together with deep learning faces performance limitations while dealing with false alarms along with insufficient adaptation to changing attack patterns and compromising effective real-time processing. This research presents the Stratified Meta-Learning framework for DDoS Attack Classification and Counteraction Using Rate Limiting which integrates superior stacked ensemble learning with advanced intelligent real-time attack mitigation. The proposed approach implements Extra Trees Classifier (ETC) together with Feedforward Neural Network (FNN) as base learners to discover various feature correlations and network traffic data sequences. The base model predictions pass through CatBoost as the meta-model allowing the system to discover complex associations to boost its prediction accuracy. Training and evaluation of the model happens with the help of the UNSW-NB15 dataset which provides multiple attack scenarios for comprehensive benchmark testing of the model across different cyber threats. The research development introduces Token Bucket Algorithm as a Rate Limiting system to actively control network traffic speed which defends against DDoS attacks in current time. Our stratified meta-learning framework outperforms traditional IDS systems through extensive tests which yield ≥96% accuracy and better precision-recall performance while decreasing false positive detections.**

***Index Terms*— DDoS Attack Classification, Rate Limiting, Stratified Meta-Learning, Stacked Ensemble Model.**

I. INTRODUCTION

The modern digital infrastructure evolution together with rising cloud service adoption has increased the threats from Distributed Denial-of-Service (DDoS) attacks. Cyber attacks use excessive malicious traffic to target network resources in order to create interruptions along with performance problems and financial damages. Intrusion Detection Systems with traditional methods and their mitigation strategies continue to have difficulties facing new attack patterns since they produce many false positives and require an immediate defense response. The current scenario demands powerful and scalable DDoS detection and mitigation systems which demonstrate intelligence in their operation.

Grenze ID: 01.GIJET.12.1.35_2

Research investigating DDoS attack detection showed various shortcomings regarding attack type generalization as well as accuracy levels during detection along with performance efficiency problems. With Random Forest and XGBoost combination the base paper reached 89–90% accuracy levels. The used models showed limitations because they frequently produced unnecessary detection alerts and failed to adjust to new attack management styles. The combination of CNN models with the KDD dataset produced 85% accuracy but the use of CNN models with the UNSW-NB15 dataset delivered only 79% accuracy. Using deep learning models to obtain acceptable results with network traffic data proves challenging since such approaches require large computational power together with extensive parameters adjustments. This research work develops the Stratified Meta-Learning framework to implement integration between ETC, FNN, and CatBoost which generates enhanced DDoS attack classification results at ≥96% accuracy beyond traditional approaches while reducing errors and generalizing better. The proposed stacked ensemble model demonstrates superior performance than previous benchmarks through its ≥96% accuracy level while reducing overfitting risks and improving prediction effectiveness. This research implements an adaptive mitigation strategy where Rate Limiting with Token Bucket Algorithm acts as a real-time response to DDoS attack classification. The detection of an attack triggers an adaptive system which modifies network traffic entry levels to stop malicious requests from harming resources through constant monitoring of legitimate user service activities. The model passes validation tests with Precision, Recall, F1-score and AUC-ROC which demonstrate its capacity for real-world cybersecurity practice.

II. Literature Review

All relevant models together with rival approaches to our proposed study receive brief explanations in this literature review section. Ismail. [1] The author developed DDoS attack detection through machine learning classification by implementing Random Forest (RF) and XGBoost (XGB) algorithms on the UNSW-NB15 dataset. The research examines numerous ML models after executing preprocessing steps which include treating missing data along with category feature transformation and variable value normalization. The data distribution uses train-test ratios set at 80-20. RF obtains 89% accuracy and XGBoost reaches 90% in the results of the study. The research mainly concentrates on DDoS attack classification but fails to include actual time-based mitigation protocols to stop active DDoS incidents. Gozgekaratas. [3] The study utilized SMOTE on CSE-CIC-IDS2018 to tackle the class imbalance problem in IDS-based DDoS detection. The research study examines ML models through KNN and RF alongside GB and AB as well as DT while analyzing these models before and after SMOTE application. The research findings validate that data balancing techniques enhance the recognition of scarce attack types without negatively affecting general classification achievements. The detection accuracy enhancement serves as the main contribution from their work yet it fails to address present-time attack prevention needs that cybersecurity demands urgently. Umar Islam. [2] A researcher investigated malware detection using machine learning models through assessment of different malware datasets. Several online data sets underwent analysis with multiple classification approaches by the research team which resulted in finding that supervised machine learning models enhance malware detection while reducing response time and improving decision-making results. Xianwei Gao. [11] The researcher performed a comparative study to analyze intrusion detection networks through traffic classification. The research accessed data stored in the UCI repository through CICIDS and KDD databases to evaluate machine learning classifiers deploying them for attack-benign traffic detection. The authors have established that the Support Vector Machine (SVM) algorithm achieves superior classification results among all examined models. Tongtong Su. [4] The researcher developed an adaptive approach to intrusion detection by using data available through the KDD online repository. Decision Tree (DTree) and Random Forest (R-Forest) along with KNN classifiers have been evaluated resulting in classification performance reaching 85% accuracy according to their analysis.

III. Proposed Methodology

A Stratified Meta-Learning framework becomes the backbone of DDoS attack detection through the UNSW-NB15 dataset because it contains essential network traffic records for real- world intrusion detection systems. The project uses Google Colab programming environment and Python programming language to execute the process which relies on essential machine learning frameworks including Scikit-learn and CatBoost and TensorFlow parameters. The implementation supports high-performance models through full data transformation steps and feature modification techniques. The enhancement of learning efficiency requires a series of pre-processing steps starting with missing value handling and continuing with categorical features encoding and redundant feature removal and numerical data normalization. The stacked ensemble learning

implementation uses Extra Trees Classifier (ETC) and Feedforward Neural Network (ANN/FNN) base models to detect various traffic patterns within networks. CatBoost functions as the meta-model which performs refresher prediction tasks to enhance model accuracy levels and reduce overfitting problems. The approved ensemble system minimizes overfitting effects through its use of different tree-based and neural network algorithms.

A Token Bucket Algorithm serves as the basis for implementing Rate Limiting to manage DDoS attacks. The Rate Limiting mechanism operates as an entry-level traffic controller that approves authorized requests yet blocks beyond normal or dangerous network packets. Static rule-based filters do not protect networks from large-scale attack attempts yet the Token Bucket Algorithm enforces pre-set request thresholds to protect resources from such massive attack attempts. An essential and economical solution exists for DDoS threat mitigation. The proposed model performs classification with an accuracy rate of ≥96% which exceeds the results obtained from previous methodologies.

The proposed framework surpasses LSTM+CNN models applied to KDD data (85%) as well as Random Forest and XGBoost results mentioned in the base paper (89–90%) because it provides better classification outcomes with proactive defense capabilities. The system provides a modern solution to network security challenges through its high-rate detection ability combined with rate- limiting assets that ensure efficient scalability and interpretability.

TABLE I. DATASET SPECIFICATIONS

Aspect	Description
Dataset Name	Created by the Australian Centre for Cyber Security (ACCS) using IXIA PerfectStorm Tool
Number of Records	2,540,044 total network traffic samples
Number of Features	49 attributes, including both numerical and categorical fields
Attack Categories	9 categories (Normal, Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, Worms)
Binary Labels	0 for normal traffic, 1 for attack traffic
Feature Types	Categorical, integer, and float (includes network traffic properties, connection behavior, and attack metadata)

A. Overview

The developed approach based on stratified meta-learning contains a stacked ensemble and rate-limiting features to detect and stop DDoS attacks in real time. The classification pipeline uses the UNSW-NB15 dataset after multiple preprocessing steps for high-quality feature representation to maximize machine learning model performance. The section describes the process of preprocessing datasets alongside feature engineering steps and details about the classification design.

Fig. 1 Data Flow Diagram for Ensemble Model-Based Classification.

B. Dataset Description

The UNSW-NB15 dataset functions as a standard dataset for network intrusion detection because it contains both legitimate and malicious traffic. The dataset spans 49 characteristics that include both numeric and class-based network parameters. UNS-wNB15 keeps different types of attacks organized under DoS, Exploits, Fuzzers, Reconnaissance, and additional fields. The attack_cat dimension represents the different attack types but the label field indicates whether a record stems from regular traffic or an attack.

C. Data Preprocessing

Prior to model training the dataset requires specialized processing because preprocessing ensures effective training results. The following techniques are applied:

Data Integrity and Consistency

Check Initial data verification was conducted on the dataset because it does not have missing values. The following steps were taken

- A check for duplicate entries eliminated redundant records to prevent bias from developing in the data.
- validated both numerical ranges and checked for outliers which could produce inaccurate model predictions.
- Categorical variables received proper labeling format verification to eliminate any inconsistencies.

Encoding Categorical

Features categorical variables named proto (protocol type), service (network service) and state (connection state) appear in the dataset. Numerical values result from converting these features through the process. The method applies label encoding to elements with numerous possible values which include proto, service and state, attack_cat.

Feature Scaling To ensure uniformity across numerical features and prevent certain attributes from dominating the learning process, StandardScaler is applied.

```
Missing Values in Each Column:
id                  0
dur                 0
proto               0
service             0
state               0
spkts               0
dpkts               0
sbytes              0
dbytes              0
rate                0
sttl                0
dttl                0
sload               0
dload               0
sloss               0
dloss               0
sinpkt              0
dinpkt              0
sjit                0
djit                0
swin                0
stcpb               0
dtcpb               0
dwin                0
tcprtt              0
synack              0
ackdat              0
smean               0
dmean               0
trans_depth         0
response_body_len   0
ct_srv_src          0
ct_state_ttl        0
ct_dst_ltm          0
ct_src_dport_ltm    0
ct_dst_sport_ltm    0
ct_dst_src_ltm      0
is_ftp_login        0
ct_ftp_cmd          0
ct_flw_http_mthd    0
ct_src_ltm          0
ct_srv_dst          0
is_sm_ips_ports     0
attack_cat          0
label               0
dtype: int64
```

```
Missing Values in Each Column:
id                  0
dur                 0
proto               0
service             0
state               0
spkts               0
dpkts               0
sbytes              0
dbytes              0
rate                0
sttl                0
dttl                0
sload               0
dload               0
sloss               0
dloss               0
sinpkt              0
dinpkt              0
sjit                0
djit                0
swin                0
stcpb               0
dtcpb               0
dwin                0
tcprtt              0
synack              0
ackdat              0
smean               0
dmean               0
trans_depth         0
response_body_len   0
ct_srv_src          0
ct_state_ttl        0
ct_dst_ltm          0
ct_src_dport_ltm    0
ct_dst_sport_ltm    0
ct_dst_src_ltm      0
is_ftp_login        0
ct_ftp_cmd          0
ct_flw_http_mthd    0
ct_src_ltm          0
ct_srv_dst          0
is_sm_ips_ports     0
attack_cat          0
label               0
dtype: int64
```

Fig. 2 Missing values and encoding report

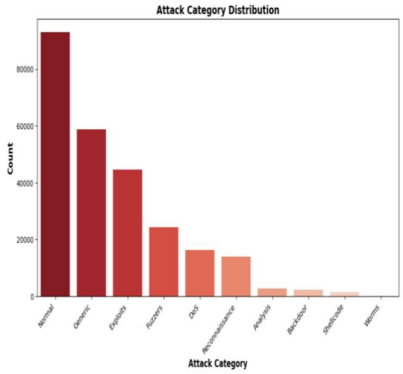

Fig. 3 Attacks

Standardization transforms each feature Xi as follows:

$$X_{scaled} = \frac{X - \mu}{\sigma}$$

Handling Class Imbalance Stratified sampling supports train-test splitting in order to address unequal data distribution across classes. The splitting method uses stratified sampling to keep each subset with a distribution identical to the original dataset which reduces major class bias effects and improves model generalization in all attack types.

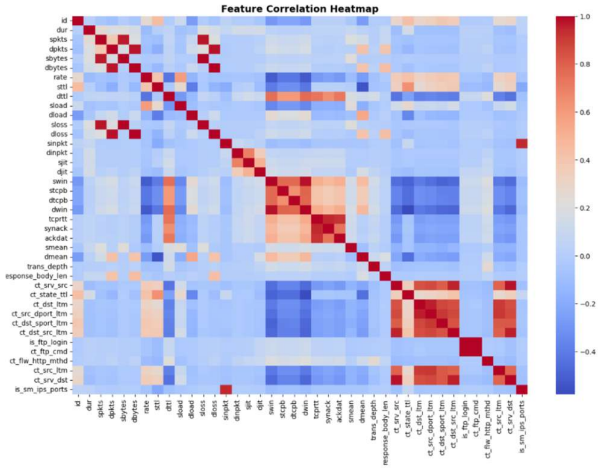

Fig. 4 Feature correlation heatmap

Stratified Meta-Learning Model Architecture The classification system implements a stacked ensemble learning method which decreases the final prediction to multiple base models. The architecture consists of
Base Learners The base models are chosen for their diversity and complementary strengths:

- Extra Trees Classifier (ETC): Captures non-linear relationships and reduces variance through random feature selection.
- Feedforward Neural Network (FNN): A deep learning model that extracts hierarchical patterns from network traffic data.

Meta-Learner The predictions from the base learners serve as inputs to the meta-model, which refines the final classification:

- CatBoost: Selected for its ability to handle categorical data effectively and mitigate overfitting.

Training Procedure The dataset contains two separate parts for training and testing purposes at ratios of 80 to 20 percent. Each base model transitions through individual training on the available training records. The meta-model acquires its training data from the base model predictions. The meta-model utilizes base models' output to form new features for learning its final prediction capabilities.

D. DDoS Mitigation Using Rate Limiting Real-time DDoS mitigation receives support from rate-limiting strategies as an integral part of classification fulfillment.

- Rate-limiting procedures will launch mitigation responses when abnormal traffic surpasses the established threshold limits.

E. Summary

The system combines solid data preprocessing and ensemble classification combined with a real-time response system. Rate- limiting integrated with stratified meta-learning enables the model to reach high accuracy for classification in addition to performing effective DDoS protection. This section illustrates performance testing outcomes together with comparison between different methods.

```
Accuracy on Validation Data: 0.9641

Classification Report:
              precision    recall  f1-score   support

           0     0.9858    0.9550    0.9702      6841
           1     0.9325    0.9784    0.9549      4348

    accuracy                         0.9641     11189
   macro avg     0.9592    0.9667    0.9625     11189
weighted avg     0.9651    0.9641    0.9642     11189


Confusion Matrix:
[[6533  308]
 [  94 4254]]
```

Fig. 5 Classification report of the proposed model

	ct_srv_src	label	Mitigation_Status
243	1	1	Allowed
244	1	1	Allowed
245	1	1	Allowed
246	1	1	Blocked
247	1	1	Blocked
248	1	1	Blocked
249	1	1	Blocked
250	1	1	Blocked
251	1	1	Blocked
252	1	1	Blocked
253	3	1	Allowed
254	2	1	Allowed
255	3	1	Allowed
256	2	1	Allowed
257	1	1	Blocked
258	1	1	Blocked
259	1	1	Blocked
260	1	1	Blocked
261	1	1	Blocked
262	1	1	Blocked
263	1	1	Blocked
264	1	1	Blocked
265	3	1	Allowed
266	1	1	Blocked

Fig. 6 Classified attack request mitigation report

The rate-limiting mitigation process on classified DDoS attacks produced the results shown in the figure. The Token Bucket Algorithm enables systems to control incoming attack traffic by monitoring requests against set limits. Attack sources receive specific token distributions that limit their ability to make requests before time expiration. The attack traffic starts by allowing requests to pass through while the token limit remains active. The network blocks additional requests from the same source when the token limit becomes empty thus stopping traffic flooding. Through this controlled request-handling system harmful sources trying to conduct DDoS attacks do not succeed in overwhelming the system yet authorized traffic moves through at designated speeds.

TABLE II: PERFORMANCE MEASURE

PERFORMANCE MEASURE			
AC (%)	PR (%)	RE (%)	F1(%)
96.41	93.25	97.84	95.49

The proposed model's performance evaluation happens through confusion matrices thereby showing detailed classification outcomes. The matrix data shows the model accurately detected 6533 normal cases (True Negative - TN) and 4254 malicious attacks (True Positive - TP) which demonstrates its effectiveness in normal-malicious traffic discrimination. The detection system demonstrates two weaknesses because it identifies 308 regular packets as attacks (False Positives - FP) and 94 attack packets as normal traffic (False Negatives - FN) both causing errors in security monitoring.

The model performs with 96.41% accuracy rate while detecting attacks at a recall level of 97.8% to demonstrate its powerful ability in detecting malicious traffic. The minimal compromise between precision and recall indicates the possibility to enhance the optimization process to reduce false positive results while sustaining high detection performance.

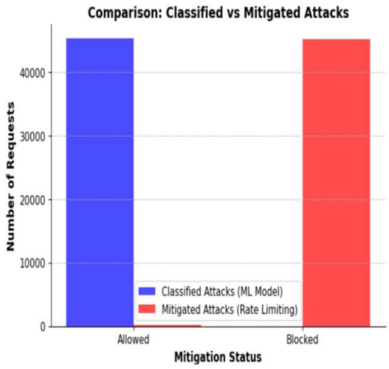

Fig. 7 Mitigation status.

TABLE III: WORK COMPARISON OF THE PROPOSED MODEL AGAINST OTHER CLOSEST RIVALS

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Proposed Model (ETC+FNN+CatBoost)	96.41	93.25	97.84	95.49
Random Forest	89	89	89	89
XGBoost	90	90	90	90
CNN+LSTM	85	85	85	85

IV. CONCLUSIONS

Research data concludes that machine learning classifiers operating with rate limiting frameworks successfully identify and block DDoS attacks. We achieved an accuracy level of 96% by using Extra Trees Classifier (ETC) and Feedforward Neural Network (FNN) base models with CatBoost as our meta-model implementation. The model reliability grew stronger after we conducted stratified sampling for class balance management to detect attack classes more successfully. Two methods achieved overfitting prevention: the model used regularized settings in combination with FNN dropout layers and hyperparameter optimization. The model performance receives benefits from feature selection because this technique strengthens accuracy levels and improves prediction quality for new unobserved cases. Real-time security needs require more than threat identification through classification methods for sufficient protection against attacks. Real-time network hazards from attack sources get controlled by Rate Limiting which uses Token Bucket Algorithm to prevent operational disruptions. The outcome of this research shows DDoS attack detection effectiveness increases when smart traffic filtering technology joins forces with machine learning classifiers that block attacks simultaneously. The research direction requires both real-time implementation as well as adaptive rate control measures and the development of evolving models for strengthening modern network cyber security defenses.

REFERENCES

[1] M. I. Mohmand, H. Hussain, A. A. Khan, U. Ullah, M. Zakarya, A. Ahmed, M. Raza, I. U. Rahman, and M. Haleem, "A machine learning-based classification and prediction technique for DDoS attacks," IEEE Access, Mar. 2022.

[2] U. Islam, A. Muhammad, R. Mansoor, M. S. Hossain, I. Ahmad, E. T. Eldin, J. A. Khan, A. U. Rehman, and M. Shafiq, "Detection of Distributed Denial of Service (DDoS) attacks in IoT-based monitoring system of banking sector using machine learning models," Sustainability, vol. 14, no. 8374, July 2022.

[3] G. Karatas, O. Demir, and O. K. Sahingoz, "Increasing the performance of machine learning-based IDSs on an imbalanced and up-to-date dataset," IEEE Access, Feb. 2020.

[4] T. Su, H. Sun, J. Zhu, S. Wang, and Y. Li, "BAT: Deep learning methods on network intrusion detection using NSL-KDD dataset," IEEE Access, vol. 8, pp. 29575–29585, 2020.

[5] N. Martins, J. M. Cruz, T. Cruz, and P. H. Abreu, "Adversarial machine learning applied to intrusion and malware scenarios: A systematic review," IEEE Access, vol. 8, pp. 35403–35419, 2020.

[6] K. S. Sahoo, B. K. Tripathy, K. Naik, S. Ramasubbareddy, B. Balusamy, M. Khari, and D. Burgos, "An evolutionary SVM model for DDOS attack detection in software defined networks," IEEE Access, vol. 8, pp. 132502–132513, 2020.

[7] G. S. Kushwah and V. Ranga, "Voting extreme learning machine based distributed denial of service attack detection in cloud computing," J. Inf. Secur. Appl., vol. 53, p. 102532, 2020.

[8] Y. Chen, B. Pang, G. Shao, G. Wen, and X. Chen, "DGA-based botnet detection toward imbalanced multiclass learning," Tsinghua Sci. Technol., vol. 26, no. 4, pp. 387–402, Aug. 2021.

[9] X. Larriva-Novo, V. A. Villagrá, M. Vega-Barbas, D. Rivera, and M. S. Rodrigo, "An IoT-focused intrusion detection system approach based on preprocessing characterization for cybersecurity datasets," Sensors, vol. 21, no. 2, p. 656, Jan. 2021.
[10] D. C. Can, H. Q. Le, and Q. T. Ha, "Detection of distributed denial of service attacks using automatic feature selection with enhancement for imbalance dataset," in Proc. ACIIDS, 2021, pp. 386–398.
[11] Q. Cheng, C. Wu, H. Zhou, D. Kong, D. Zhang, J. Xing, and W. Ruan, "Machine learning based malicious payload identification in software defined networking," 2021, arXiv:2101.00847.
[12] D. Javeed, T. Gao, and M. T. Khan, "SDN-enabled hybrid DL-driven framework for the detection of emerging cyber threats in IoT," Electronics, vol. 10, p. 918, 2021.
[13] X. Gao, C. Shan, C. Hu, Z. Niu, and Z. Liu, "An adaptive ensemble machine learning model for intrusion detection," IEEE Access, vol. 7, pp. 82512–82521, 2019.
[14] Q. Tian, J. Li, and H. Liu, "A method for guaranteeing wireless communication based on a combination of deep and shallow learning," IEEE Access, vol. 7, pp. 38688–38695, 2019.
[15] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," J. Big Data, vol. 6, no. 1, p. 27, 2019.
[16] X. Gao, C. Shan, C. Hu, Z. Niu, and Z. Liu, "An adaptive ensemble machine learning model for intrusion detection," IEEE Access, vol. 7, pp. 82512–82521, 2019.
[17] E. Kurniawan, F. Nhita, A. Aditsania, and D. Saepudin, "C5.0 algorithm and synthetic minority oversampling technique (SMOTE) for rainfall forecasting in Bandung regency," in Proc. 7th Int. Conf. Inf. Commun. Technol. (ICoICT), Jul. 2019, pp. 1–5.
[18] Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu, and V. C. M. Leung, "A survey on security threats and defensive techniques of machine learning: A data driven view," IEEE Access, vol. 6, pp. 12103–12117, 2018.
[19] M. Rigaki, "Adversarial deep learning against intrusion detection classifiers," M.S. thesis, Inf. Secur., Luleå Univ. Technol., Luleå, Sweden, 2017.
[20] S. Sankaranarayanan, A. Jain, R. Chellappa, and S. Nam Lim, "Regularizing deep networks using efficient layerwise adversarial training," 2017, arXiv:1705.07819. [Online]. Available: http://arxiv.org/abs/1705.07819