A MACHINE LEARNING FRAMEWORK FOR MULTICLASS DETECTION OF DDOS NETWORK ATTACKS

Pan Huiqin, master's degree student, 467311

Belarusian State University of Informatics and Radioelectronics
Minsk, Republic of Belarus

Nasonova N.V. – PhD in Technical Sciences, Associate Professor

Annotation. This paper studies machine-learning-based multiclass detection of DDoS traffic using the CIC-DDoS2019 dataset. After preprocessing, four models are compared: SVM, Random Forest, XGBoost, and MLP. The results show that Random Forest achieves the best overall performance, with near-perfect accuracy and macro-F1. A practical deployment scheme at network boundaries is also outlined for near real-time detection.

Keywords. DDoS-attacks, machine learning, supervised learning.

DDoS attacks pose a serious threat to the availability and reliability of network services by sending a large amount of traffic to the target. As attackers continuously advance their techniques and botnet sizes expand, distinguishing malicious traffic from normal network behavior becomes increasingly challenging. Machine learning is well-suited for this task because it can automatically learn discriminative patterns from traffic data. By modeling complex, high-dimensional network behaviors, supervised learning methods can improve detection accuracy and generalize to unseen or evolving types of DDoS attacks.

Support Vector Machine (SVM) is a supervised learning algorithm based on statistical learning theory, introduced by Vapnik and others in the 1990s. Its main idea is to find an optimal separating hyperplane that maximizes the margin between classes, improving classification performance. Key mechanisms include: 1) the kernel trick, which maps nonlinearly separable data into a higher-dimensional space using kernels such as linear, polynomial, and RBF; 2) support vectors, which are the critical samples defining the decision boundary; and 3) the soft margin, which introduces slack variables to handle noise and outliers.

Random Forest is an ensemble classification algorithm that improves accuracy and robustness by combining multiple decision trees. Its main mechanism relies on two sources of randomness: first, each tree is trained on a Bootstrap sample drawn with replacement from the training set; second, at each split, only a random subset of features is considered. This increases diversity among trees and allows the model to learn different views of the data. For prediction, each tree makes an independent decision, and the final class is determined by majority voting.

XGBoost (Extreme Gradient Boosting) is an ensemble learning algorithm based on Gradient Boosting Decision Trees (GBDT). It builds trees sequentially, where each new tree learns the residual errors of the previous ones, gradually improving the overall prediction. XGBoost optimizes the objective function using a second-order Taylor expansion and includes regularization terms to reduce overfitting. It also supports parallel computation, handles missing values automatically, and offers high flexibility and scalability, which often leads to very strong performance in classification and regression tasks.

The training is based on the CIC-DDoS2019 dataset, which contains approximately 485,000 samples with 87-dimensional flow features, covering 1 type of normal traffic and 10 types of DDoS attacks. After filling in missing values, handling outliers, encoding categorical labels, and Z-score standardization, the data is split into training and test sets in an approximately 80:20 stratified manner. All preprocessing parameters are fitted only on the training set and then applied to the test set to avoid data leakage.

In terms of training strategy, traditional machine learning optimizes hyperparameters using grid search and cross-validation, while the deep model MLP uses the Adam optimizer and categorical cross-entropy loss, combined with early stopping, learning rate decay, as well as batch normalization and Dropout to control overfitting. Finally, each model is evaluated on the test set using metrics such as accuracy, precision, recall, and macro-average F1, and the training process and results can be visualized using learning curves, confusion matrices, and feature importance plots.

The performance of the random forest model on the test set is shown in Table 1. As seen from the results in the table, the four evaluation metrics are all very close to 1 overall, indicating that the model has good overall discriminative ability in the multi-classification task across 11 categories (normal traffic + 10 types of DDoS attacks). Specifically, a high precision indicates that when the model identifies an attack, the vast majority of samples indeed belong to the corresponding attack type, thereby reducing the risk of false alarms; at the same time, a high recall suggests that the model can cover most of the actual attack samples, with relatively few missed detections, allowing potential threats to be discovered more promptly. The balanced performance of precision and recall (macro-average F1-score close to 1) further demonstrates that the model's ability to recognize different attack categories is relatively even, rather than performing well only on a few categories.

Compared with SVM, XGBoost, and deep learning models (such as MLP), random forest achieves the most balanced overall performance. Therefore, it can not only achieve stable classification results on known

data distributions, but is also more suitable as the preferred model for scenarios with unknown traffic. In real network environments, attack behavior may have variations or distribution shifts, and models that can maintain both high precision and recall are often more reliable.

Table 1 – Random Forest Algorithm Parameters

Metric	Accuracy	Precision (Macro Avg)	Recall (Macro Avg)	F1-Score (Macro Avg)
Random Forest	0,998960	0,997982	0,996922	0,997455

Deploy the trained random forest detection module at network boundaries or traffic aggregation points, such as the firewall, the interior of a gateway, or in conjunction with existing NetFlow, sFlow, or mirrored traffic exports. At the collection points, extract flow-level features from the traffic (using the same 87 dimensions or selected features as in the training phase), then perform the same standardization and encoding on the features as used during training, and feed them into the random forest model to obtain the class (normal or specific attack type) and confidence level.

The detection module only performs classification and alerting, and can be decoupled from blocking and rate limiting: it sends the 'predicted category + confidence level + flow identifier' to downstream systems via message queue or interface, and the blocking module generates firewall rules, ACLs, or rate-limiting policies based on the strategies and deploys them to boundary devices. High-confidence attacks can automatically trigger blocking, medium-confidence ones can be rate-limited or have enhanced monitoring, and low-confidence ones can be recorded only for analysis.

Random forests have low inference latency, making them suitable for near real-time detection; during deployment, fix the optimal hyperparameters and preprocessing steps obtained during the training phase, and periodically retrain or validate with newly labeled traffic to cope with concept drift and new attack types, ensuring continuous effectiveness in the IDS.

List of references

1. Khemapatapan C .2- STAGE SOFT DEFENDING SCHEME AGAINST DDOS ATTACK OVER SDN BASED ON NB AND SVM[J]. 2018.

2.Malliga S , Kogilavani S V , Sowmya R .Deep discover: Deep learning models for detecting distributed denial of service (DDOS) attacks[J].AIP Conference Proceedings, 2022, 2393(1):7.DOI:10.1063/5.0074445.

3.Berqia A , Bouijij H .Predicting DoS/DDoS Attacks Using Deep Learning Models[J].2024 6th International Symposium on Advanced Electrical and Communication Technologies (ISAECT), 2024:1-7.DOI:10.1109/isaect64333.2024.10799580.

4.Narender M , Y. N .Deep Regularization Mechanism for Combating Class Imbalance Problem in Intrusion Detection System for Defending DDoS Attack in SDN[J].Journal of Computer Science, 2023.DOI:10.3844/jcssp.2023.334.344.

5. Draper-Gil, G., Lashkari, A. H., Mamun, M., & Ghorbani, A. A. (2016). Characterization of Encrypted and VPN Traffic Using Time-Related Features. In Proceedings of the 2nd International Conference on Information Systems Security and Privacy (ICISSP), 407-414. Rome, Italy.

6.Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2019). Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. In Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP), 108-116.