Detection of Denial-of-Service Attacks in Mobile Ad Hoc Networks Using Machine Learning Classifiers

Varanasi Sai Srinivasa Karthik, Pravallika Ghantasala, Mitta Sreenidhi Reddy, Narra Rajeswari, Arshad Ahmad Khan Mohammad*

Department of CSE (Cybersecurity), GITAM School of Technology, Hyderabad

*Corresponding author: amohamma2@gitam.edu

Abstract

Mobile Ad Hoc Networks present unique security challenges due to their decentralized architecture and dynamic topology. We address the problem of detecting Denial-of-Service attacks using machine learning classifiers trained on network flow features. Our dataset comprises 4,207 samples with 21 features extracted from simulated MANET traffic, categorized into three classes: normal operation, legitimate congestion, and attack traffic. We evaluate Random Forest, XGBoost, Support Vector Machine, and K-Nearest Neighbors using stratified five-fold cross-validation with proper preprocessing to prevent data leakage. XGBoost achieves the highest multiclass accuracy of 94.7% with a Cohen's Kappa of 0.921, while binary classification reaches 96.7% accuracy with ROC-AUC of 0.994. Statistical testing confirms that XGBoost significantly outperforms Random Forest (p = 0.009). Feature importance analysis identifies queue length, buffer utilization, and packet forwarding consistency as the most discriminative metrics. We discuss practical implications for MANET security and acknowledge limitations of simulation-based evaluation.

Keywords: Mobile Ad Hoc Networks, Intrusion Detection, Machine Learning, Denial-of-Service, Network Security

1. Introduction

Mobile Ad Hoc Networks differ fundamentally from conventional wireless networks in their lack of fixed infrastructure. Nodes in a MANET communicate directly with one another and cooperatively route packets through intermediate hosts. This architecture enables rapid deployment in scenarios where traditional infrastructure is unavailable or impractical, including military operations, disaster response, and vehicular networks. However, the same characteristics that provide flexibility also create substantial security vulnerabilities.

Denial-of-Service attacks pose a particularly serious threat to MANETs. An attacker can disrupt network operations by flooding the medium with spurious packets, overwhelming node buffers, or corrupting routing tables. The open wireless channel and cooperative routing assumptions make detection difficult because malicious behavior can mimic legitimate congestion patterns. Conventional signature-based intrusion detection systems fail in this environment because they cannot adapt to the network's changing topology or identify novel attack variants.

Machine learning offers an alternative approach that learns patterns from network behavior rather than relying on predefined attack signatures. Classifiers can be trained to distinguish between normal traffic, legitimate congestion, and attack scenarios based on features extracted from network flows. This paper evaluates several machine learning algorithms for this task and identifies which network metrics are most useful for detection.

Our work makes four contributions. First, we compile a dataset of network flow records from simulated MANET environments with realistic feature distributions. Second, we evaluate four classifiers with proper methodology including stratified cross-validation and preprocessing within each fold to avoid information leakage. Third, we perform statistical significance testing to confirm performance differences between models. Fourth, we analyze feature importance to identify which metrics are most useful for practitioners deploying intrusion detection systems.

2. Related Work

Research on intrusion detection in mobile networks has followed two main directions: anomaly-based methods that learn normal behavior patterns and misuse-based systems that match known attack signatures. Nadeem and Howarth provided an extensive survey of techniques developed for MANETs, noting that anomaly detection generally performs better because it can identify previously unseen attacks. However, anomaly detection suffers from higher false positive rates when legitimate network congestion triggers alerts.

Several studies have applied machine learning to network intrusion detection. Zhang and colleagues used Random Forest on the KDD Cup dataset and achieved detection rates above 90% for most attack categories. Their analysis showed that ensemble methods generally outperform single classifiers because they can capture complex decision boundaries. More recent work has examined gradient boosting methods, with Chen and Guestrin demonstrating that XGBoost provides both accuracy and computational efficiency advantages over Random Forest on tabular data.

MANET-specific intrusion detection has received less attention. Kurosawa and colleagues proposed an SVM-based system for detecting black hole attacks in AODV networks, reporting 95% detection accuracy on their simulated dataset. Huang and colleagues combined fuzzy logic with neural networks for multiclass attack detection but did not compare their method against standard classifiers. A limitation of much existing work is the use of single train-test splits rather than cross-validation, which can produce overly optimistic performance estimates.

Our work addresses gaps in existing literature by providing a rigorous comparison of multiple classifiers on MANET-specific features with proper cross-validation methodology. We also distinguish between binary detection (attack versus no attack) and three-class classification that separates attacks from legitimate congestion, which is important for reducing false positives in operational systems.

3. Dataset and Features

We generated a dataset of 4,207 network flow records from simulated MANET environments. The simulation parameters match typical MANET deployment scenarios: 30 nodes distributed in a 1000 by 1000 meter area using the AODV routing protocol with Random Waypoint mobility at speeds between 1 and 10 meters per second. Traffic consists of constant bit rate UDP flows with 512 byte packets sent at 4 packets per second. Each simulation runs for 300 seconds.

The dataset contains three classes. Smooth samples (1,430 instances, 34%) represent normal network operation with stable routes and low congestion. Non-Malicious samples (1,388 instances, 33%) represent legitimate congestion scenarios where increased traffic or mobility causes degraded performance without any attack. Malicious samples (1,389 instances, 33%) represent DoS flooding attacks where a compromised node generates excessive traffic to overwhelm neighboring nodes.

We extract 21 features from each flow, organized into four categories. Queue metrics include queue length and buffer utilization, which directly measure network congestion. Packet metrics include delivery ratio, drop rate, and forwarding consistency, which indicate routing efficiency. Timing metrics include response time and propagation delay, which measure network responsiveness. Behavioral metrics include route stability, trust values, collision rates, and protocol error counts, which capture anomalous behavior patterns. We selected these features because they are available at the network layer and do not require deep packet inspection.

4. Methodology

4.1 Classifiers

We evaluate four classification algorithms representing different learning paradigms. Random Forest is an ensemble of decision trees that reduces variance through bagging and random feature selection at each split. We use 200 trees with maximum depth of 15 and balanced class weights to handle slight class imbalance. XGBoost implements gradient boosted trees with regularization to prevent overfitting. We use 200 boosting rounds with maximum depth of 8 and learning rate of 0.1. Support Vector Machine finds optimal hyperplanes separating classes in a transformed feature space. We use the RBF kernel with C=10 and balanced class weights. K-Nearest Neighbors classifies samples based on the majority class among their closest neighbors in feature space. We use 7 neighbors with distance weighting.

4.2 Evaluation Protocol

We use stratified five-fold cross-validation to estimate classifier performance. This procedure divides the data into five equal parts while maintaining class proportions in each fold. Each fold serves as the test set once while the remaining four folds form the training set. This yields five performance estimates whose mean and standard deviation characterize expected accuracy on new data.

A critical aspect of our methodology is performing feature scaling within each fold rather than on the full dataset before splitting. This prevents information leakage where statistics from test samples contaminate the training process. Within each fold, we fit a StandardScaler on training data and apply the same transformation to test data. This ensures that test performance reflects what would be observed on truly unseen samples.

We report accuracy, precision, recall, and F1-score using weighted averaging across classes. We also compute Cohen's Kappa, which adjusts for chance agreement, and Matthews Correlation Coefficient, which is robust to class imbalance. For binary classification, we additionally report ROC-AUC and average precision scores. We use paired t-tests to assess whether performance differences between classifiers are statistically significant.

5. Results

5.1 Multiclass Classification

Table 1 presents cross-validation results for three-class classification distinguishing Smooth, Non-Malicious, and Malicious traffic. XGBoost achieves the highest accuracy at 94.7% with standard deviation of 0.5% across folds. Random Forest follows at 94.2% and SVM at 93.9%. KNN performs notably worse at 91.8% accuracy, likely due to the curse of dimensionality with 21 features. Cohen's

Kappa values above 0.9 for the tree-based methods indicate excellent agreement beyond chance.

Model	Accuracy (%)	F1 (%)	Kappa	MCC
XGBoost	94.7 ± 0.5	94.8 ± 0.5	0.921	0.921
Random Forest	94.2 ± 0.3	94.2 ± 0.3	0.913	0.913
SVM	93.9 ± 0.4	93.9 ± 0.4	0.908	0.908
KNN	91.8 ± 0.9	91.8 ± 0.9	0.878	0.881

Table 1: Multiclass classification results (5-fold CV)

5.2 Binary Classification

For binary classification where Smooth and Non-Malicious are combined into a No-Attack class, performance improves substantially. Table 2 shows that XGBoost achieves 96.7% accuracy with ROC-AUC of 0.994, indicating excellent discrimination between attack and benign traffic. All classifiers except KNN achieve ROC-AUC above 0.99. These results suggest that distinguishing attacks from non-attacks is considerably easier than the three-class problem where legitimate congestion must be separated from attacks.

Model	Accuracy (%)	F1 (%)	ROC-AUC	Avg Prec
XGBoost	96.7 ± 0.6	96.7 ± 0.6	0.994	0.989
Random Forest	96.3 ± 0.3	96.3 ± 0.3	0.994	0.987
SVM	96.1 ± 0.7	96.1 ± 0.7	0.993	0.984
KNN	94.1 ± 0.9	94.2 ± 0.9	0.985	0.951

Table 2: Binary classification results (5-fold CV)

5.3 Statistical Significance

We performed paired t-tests comparing the fold-wise accuracies of different classifiers. XGBoost significantly outperforms Random Forest on the multiclass task ($t = 4.71$, $p = 0.009$) despite the relatively small absolute difference of 0.5 percentage points. This confirms that the performance advantage of gradient boosting over bagging is consistent across different data subsets rather than an artifact of a particular split. The differences between XGBoost and SVM, and between Random Forest and SVM, are also statistically significant at the 0.05 level.

5.4 Feature Importance

Feature importance scores from Random Forest and XGBoost identify which network metrics are most useful for classification. Both models rank queue length as the most important feature, followed by buffer utilization and packet forwarding consistency. These metrics directly measure the network congestion that DoS attacks create. Route stability and CPU utilization also rank highly, reflecting the resource exhaustion and routing disruption caused by flooding attacks. Interestingly, malformed packet counts and protocol errors rank lower despite being intuitive attack indicators, suggesting that volume-based features are more discriminative than anomaly flags in our dataset.

6. Discussion

Our results demonstrate that machine learning classifiers can effectively detect DoS attacks in MANETs when trained on appropriate network features. The 94.7% multiclass accuracy achieved by XGBoost compares favorably with results reported in existing literature, while our rigorous cross-validation methodology provides more reliable performance estimates than single split evaluations.

The performance gap between binary and multiclass classification has practical implications. Systems that only need to distinguish attacks from non-attacks can achieve nearly 97% accuracy, which may be acceptable for many applications. However, the 5-6% of errors will include both missed attacks (false negatives) and false alarms on legitimate congestion (false positives). The three-class formulation allows operators to separately tune thresholds for attack detection and congestion identification, potentially reducing unnecessary responses to benign conditions.

The feature importance analysis suggests that network operators should prioritize monitoring queue-related metrics for DoS detection. Queue length and buffer utilization are straightforward to measure at routers without specialized hardware, making them practical for deployment. The lower importance of behavioral features like trust values indicates that simple resource monitoring may be sufficient for detecting flooding attacks, though more sophisticated attacks might require additional indicators.

The statistical significance of performance differences between classifiers informs model selection. XGBoost's advantage over Random Forest justifies the additional complexity of gradient boosting for applications where accuracy is paramount. However, Random Forest may be preferable when interpretability or training speed are priorities, since its performance is only marginally lower.

7. Limitations and Future Work

Several limitations of this work should be acknowledged. First, our dataset comes from simulated network environments rather than real-world traffic. While simulation allows controlled experimentation, actual networks exhibit traffic patterns and attack behaviors that may differ from our synthetic data. Validating these results on traffic from real MANET testbeds is an important next step.

Second, we focus exclusively on DoS flooding attacks. MANETs face other threats including black hole attacks, wormhole attacks, and Sybil attacks that our classifiers may not detect. Extending the approach to multiple attack types would increase practical utility but also classification complexity.

Third, our dataset size of 4,207 samples is relatively modest by machine learning standards. Larger datasets might improve classifier performance, particularly for the Non-Malicious class that overlaps substantially with both other classes. Additionally, we have not explored deep learning methods that might extract more discriminative features automatically from raw network data.

Future work should address these limitations while also investigating deployment considerations such as computational requirements for resource-constrained nodes and adaptation to concept drift as network conditions change. Online learning methods that update models incrementally may be necessary for practical systems.

8. Conclusion

This paper evaluated machine learning classifiers for detecting DoS attacks in Mobile Ad Hoc Networks. Using a dataset of 4,207 network flow samples with 21 features, we compared Random

Forest, XGBoost, SVM, and KNN under stratified five-fold cross-validation with proper preprocessing to prevent data leakage. XGBoost achieved the best multiclass accuracy of 94.7% with Cohen's Kappa of 0.921, significantly outperforming other classifiers according to statistical testing. Binary classification reached 96.7% accuracy with ROC-AUC of 0.994. Feature importance analysis identified queue length, buffer utilization, and packet forwarding consistency as the most discriminative metrics. While limitations exist regarding the use of simulated data and focus on a single attack type, our results demonstrate the viability of machine learning for MANET intrusion detection and provide guidance on feature and model selection. Code and data are available at https://github.com/vssk18/manet-ids to support reproducibility.

References

[1] A. Nadeem and M.P. Howarth, A survey of MANET intrusion detection and prevention approaches for network layer attacks, IEEE Communications Surveys and Tutorials, vol. 15, no. 4, pp. 2027-2045, 2013.

[2] J. Zhang, M. Zulkernine, and A. Haque, Random-forests-based network intrusion detection systems, IEEE Transactions on Systems, Man, and Cybernetics Part C, vol. 38, no. 5, pp. 649-659, 2008.

[3] T. Chen and C. Guestrin, XGBoost: A scalable tree boosting system, Proceedings of KDD, pp. 785-794, 2016.

[4] S. Kurosawa et al., Detecting blackhole attack on AODV-based mobile ad hoc networks by dynamic learning method, International Journal of Network Security, vol. 5, no. 3, pp. 338-346, 2007.

[5] C. Perkins and E. Royer, Ad-hoc on-demand distance vector routing, Proceedings of IEEE WMCSA, pp. 90-100, 1999.

[6] L. Breiman, Random forests, Machine Learning, vol. 45, no. 1, pp. 5-32, 2001.

[7] V. Vapnik, The Nature of Statistical Learning Theory, Springer, 1995.