Information Dynamics and Applications
https://www.acadlore.com/journals/IDA

Leveraging Artificial Intelligence for Blackhole Attack Detection in MANETs: A Comparative Study

Zainab Bashar Ibrahim*, Mayada Faris Ghanim

Computer Engineering Department, University of Mosul, 41001 Mosul, Iraq

* Correspondence: Zainab Bashar Ibrahim (Zainab.en1384@student.uomosul.edu.iq)

Received: 11-08-2024 **Revised:** 11-29-2024 **Accepted:** 12-16-2024

Citation: Z. B. Ibrahim and M. F. Ghanim, "Leveraging artificial intelligence for blackhole attack detection in MANETs: A comparative study," *Inf. Dyn. Appl.*, vol. 3, no. 4, pp. 245–257, 2024. https://doi.org/10.56578/ida030404.

Abstract: Blackhole attacks represent a significant threat to the security of communication networks, particularly in emerging network architectures such as Mobile Ad Hoc Networks (MANETs). These attacks, characterized by their ability to obscure malicious behavior, evade conventional detection methods due to their loosely defined signatures and their ability to bypass traditional filtering mechanisms. This study investigates the application of machine learning techniques, specifically Support Vector Machine (SVM), Convolutional Neural Network (CNN), and Decision Tree (DT), for the detection and mitigation of blackhole attacks in MANETs. Simulations conducted in MATLAB 2023a examined network configurations with node densities of 50, 100, 250, and 500 nodes to assess the performance of these classifiers in comparison to conventional detection approaches. The results demonstrated that both SVM and CNN achieved near-perfect detection accuracy of 100% across all network configurations, outperforming traditional methods. SVM was chosen due to its efficacy in handling high-dimensional data, CNN for its ability to learn complex, nonlinear hierarchical features, and DT for its interpretability. The findings underscore the potential of these machine learning models in enhancing the precision of blackhole attack detection, thereby improving network security. Future research is recommended to explore the scalability and training efficiency of these models, particularly through the integration of advanced techniques such as model fusion and deep learning architectures. This study contributes to the growing body of literature on radar wave radio (RWR)-based and machine learning-based attack detection and highlights the potential of artificial intelligence (AI) solutions in transforming traditional emitter identification methods, offering significant improvements to network protection systems.

Keywords: MANET; AODV; Blackhole; AI; Security; Machine learning

1 Introduction

MANETs are wireless networks without fixed infrastructure and do not have centralized packet routing administration [1, 2]. Because of these unique characteristics, MANETs are vulnerable to a range of security threats, such as blackhole attacks [3–5], where hostile nodes discard packets on purpose, seriously disrupting connectivity. Maintaining the availability, confidentiality, and integrity of network services and data depends on MANET security [6]. When two nodes are within range of one another, single-hop direct communication takes place; otherwise, multi-hop communication takes place via intermediary nodes. Routing protocols for MANETs are often separated into three categories: proactive, reactive, and hybrid [7]. For this reason, the most used reactive protocol for packet transmission is the classic Ad-hoc On-demand Distance Vector (AODV) routing protocol. Since AODV incorporates both route identification and repair methods, it is widely used in MANETs for the distribution of multimedia and emergency information. However, because of wireless communication and finite energy resources, security remains an important factor. As a result, several types of attacks, such as blackhole, wormhole, grayhole, etc., can significantly impair the performance of MANETs [3]. By enabling hostile nodes to propagate erroneous pathways to the source node as viable routes, black hole attacks take advantage of the AODV route discovery process. They indicate a new path to the target by sending an route reply (RREP) with a destination sequence number greater than the route request (RREQ) message. The most prevalent kind of attack on MANETs is this one because routing algorithms frequently deliver packets by mistakenly trusting their nearby nodes, which causes packet loss. It's challenging to distinguish between a packet loss caused by a blackhole assault and regular network activity [8, 9].

Network security is one area where AI has shown to be revolutionary. The increasing sophistication and frequency of cyberattacks need the use of advanced defense-oriented technologies. Traditional security methods sometimes fail to detect new and evolving threats because they rely on pre-established rules and signatures. AI offers many solutions for these issues because of its ability to learn from the data and adapt to new patterns [10]. Recent studies have indicated that AI techniques are not only capable of detecting cyberattacks but can also mitigate them in real time, thereby enhancing the resilience of network infrastructures. Furthermore, the incorporation of AI into current security frameworks has resulted in considerable improvements in threat detection accuracy and reaction times [10, 11].

1.1 Blackhole Attack

Blackhole attack is one type of attack that is common in MANETs. The attacker node claims that it is the shortest and best path for the destination to drop all the packet data and then it decides to send them to the destination or delete them, which creates a "black hole" where the data just disappears [12–14]. The network could stop and the performance could reduce because of the loss of the data packets. This type of attack takes advantage of how routing systems find routes like in the AODV protocol [13, 14]. When the source node doesn't have the data about the destination and the path, it sends a RREQ message to all the nodes in AODV. The RREP message is sent by the attacker node to the source that claims it is the shortest and has a lower hop count to show that is the best path. The source node believes this and sends its data based on that information [13, 14]. Blackhole attacks can cause damage to the networks, which is why it is important to understand and stop them. These attacks can cause data loss, more delay, and lower network performance. MANETs are used in areas like disaster recovery, military missions, and vehicle communication, and it's really important to protect and secure these networks [14, 15]. Blackhole attacks are still important today, despite being a well-known attack vector. To deal with the growing complexity of these attacks, researchers have been creating new methods. Many improvements to the AODV protocol and other techniques have been proposed in recent studies to make MANETs more secure against blackhole attacks [15, 16].

1.2 Key AI Techniques in Network Security

A variety of AI approaches have been used for network security as follows:

- SVM: A supervised learning model that classifies data by finding the best hyperplane to separate different classes. In network security, SVM uses features extracted from network traffic to classify whether the behavior is normal (benign) or harmful (malicious) [17]. SVM is a useful tool in detecting network threats and other security risks due to its ability to handle high-dimensional data and its efficacy in finding patterns [18, 19].
- CNN: A type of deep learning model that focuses on pattern recognition. CNNs may detect anomalies and intrusions in network security by examining traffic patterns as features and identifying complex attack signatures [20]. Recent developments have demonstrated CNNs' ability to analyze real-time network traffic data efficiently, increasing the security infrastructure's overall efficacy and boosting the detection accuracy of anomalous activity.
- DT: DTs are used to create models that predict the values of target variables by starting with different input variables. Depending on the characteristics of the data, DTs may be used in network security to classify network traffic as benign or malicious by following a tree-like structure of decision rules [21].

1.3 Problem Statement

The increased reliance on networked systems and the rapid development of technology itself have shifted cybersecurity to the fore. Blackhole attacks are cyber threats that severely compromise network security, particularly in MANETs. Blackhole attacks are a form of malicious nodes in which they claim to be the fastest route to the target node only to absorb data packets and throw them away. They destructively affect network communication and cause huge delays. Because of their complexity and fluidity in nature, traditional detection mechanisms based on established criteria and signatures are unfitting in the case of new or complex blackhole assault recognition and these approaches fail in many cases. This limitation highlights the necessity of mechanization and flexibility. Such methods include AI methods as they can faster and more accurately improve the detection of blackhole attacks because of their ability to detect models and learn from data.

1.3.1 Research objectives

The primary objective of this research is to develop and evaluate AI-based models for the detection of blackhole attacks in network security. Specifically, the research aims at:

- Checking in AI techniques: Checking how to identify blackhole attacks using different AI techniques, such as SVM, CNN, and DT.
- Feature selection and extraction: From network traffic data, pertinent information may be located and extracted to train AI models that can discriminate between benign and malevolent behavior.
- Model development: Utilizing SVM, CNN, and DT algorithms to create AI-based detection models. These models can be improved to get low false-positive rates and high accuracy.

- Performance evaluation: Measuring the AI models' performance using a range of measures, including computational efficiency, accuracy, precision, recall, and F1-score. The outcomes can be analyzed using conventional detection techniques.

This study is expected to advance the creation of more adaptable and efficient network security solutions to strengthen cybersecurity generally and better defend against blackhole assaults.

2 Literature Review

Blackhole attacks are a significant threat in network security, particularly in MANETs and wireless sensor networks (WSNs). Several methods have been proposed to detect and mitigate these attacks. Rani et al. [22] found black holes in Internet of Things (IoT)-MANET routes and channeled them through protected nodes using an improved AODV routing protocol with SVM and Artificial Neural Network (ANN). The primary objective of the research is to improve data packet transmission efficiency based on node location, energy consumption, and data transmission delay. The AODV with Artificial Bee Colony (ABC), ANN, and SVM approach performed well, with an average PDR, throughput, and latency of 97.96%, 92.78 Kbps, and 0.04 s, respectively. A CNN-based intrusion detection method has been recommended for automatically performing complex feature extraction in constantly changing environments, which is essential for network Intrusion Detection System (IDS). Deep Neural Network (DNN)-IDS boosts user confidence and communication, as the black-box nature of DNNs limits visibility, but the IDS is vital for building trust. By training DNN-IDS, the input features are optimized for identifying any type of intrusion [23].

Shafi et al. [3] proposed an effective machine learning-based secure AODV routing system to detect floods and blackhole attacks in MANETs. The method improved secure communication by using an ANN with a SVM classifier to increase intrusion detection accuracy and throughput. The MLAODV is suitable for information exchange in semi-urban areas but not in completely urban ones due to its dynamic node density and speeds. The efficacy was assessed against the current AODV processes, which are trust-based and conventional.

Furthermore, a proposal has been made for an IDS to prevent sinkhole attacks in mobile sink MANETs [24]. A hacked node that advertises fictitious routing changes in an attempt to draw in network traffic is called a sinkhole. This system classifies data using different machine learning methods, such as SVM, CNN, K-Nearest Neighbor (KNN), and DT. After gathering 3,997 distinct samples, including 256 malicious and 3,604 normal, the study discovered that CNN had the best accuracy of 98.6%. DT, SVM, and KNN came in second and third, with accuracies of 98.4%, 97.8%, and 96.7%, respectively.

2.1 Survey of AI Techniques Used in Network Security

AI algorithms can learn from data and recognize intricate patterns so that they have become more popular in the field of network security. The following AI methods are frequently used:

- Machine learning
  – SVM: SVMs use the optimal hyperplane to divide and classify the data into groups. Through the examination of network traffic patterns, they are successful in identifying intrusions [25].
  – DT: DTs use a sequence of decisions based on the characteristics of the data to classify the data. They are helpful in developing comprehensible intrusion detection models.
- Deep learning
  – CNN: CNNs work very well for pattern and picture recognition. They are employed in network security to identify abnormalities and intrusions through the analysis of traffic patterns as 1D feature vectors [26].
  – Recurrent Neural Network (RNN): Network traffic flows and other time-series data patterns can be found in RNNs [27].
- Hybrid approaches
  – Many AI approaches can improve detection performance. For instance, hybrid models that combine SVM and deep learning methods have demonstrated enhanced efficacy in identifying complex assaults [28].

3 Methodology

This section descries the network environment and attack scenarios. In this research, MATLAB 2023a was used to simulate a network environment representative of a typical MANET. The network consists of a varying number of nodes (50, 100, 250, 500) deployed over a defined area of 1000 m × 1000 m. Nodes communicated with each other using the AODV routing protocol. The simulation code was programmed to produce a MANET with nodes placed randomly within a given area, including the dynamics of blackhole attacks. The simulation gathered information regarding its behavior at normal and under attack mode.

Attack scenarios:

- Normal scenario: All nodes functioned correctly, without any malicious activities in the routing data packets. Figure 1 shows the random distribution of the nodes.

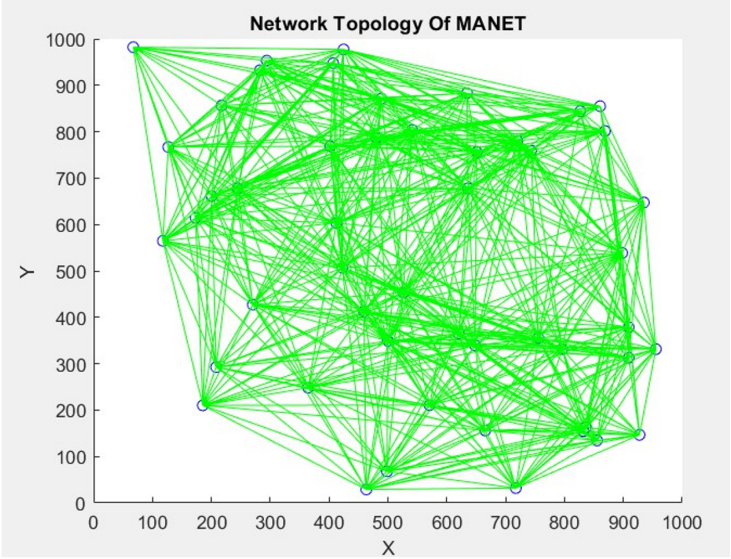

Figure 1. Network topology of a MANET with 50 nodes

- Blackhole attack scenario: One or more malicious nodes were introduced into the network. These nodes advertised the shortest path to the destination node but dropped all received packets, disrupting the network's communication. The red links in Figure 2 are the attacks. Simulation parameters included the simulation time of 500 seconds and the transmission range of 500 meters. To detect blackhole attacks, several AI techniques were employed as follows:
- SVM: A supervised learning model that classifies data by finding the optimal hyperplane separating different classes. SVM was chosen for its robustness and effectiveness in handling high-dimensional data.
- CNN: A deep learning model particularly effective for image and pattern recognition. CNNs were used to detect anomalies by treating network traffic data as 1D feature vectors, enabling the identification of complex patterns indicative of blackhole attacks.
- DT: A model that predicts the value of a target variable based on several input features. DTs were utilized for their interpretability and ease of use in classifying network traffic as normal or malicious.

3.1 Data Collection and Preprocessing Methods

The simulation gathered information regarding its behavior at normal and under attack mode, which is useful in cases of detection. The detection process was conducted separately using classification algorithms written in MATLAB for each algorithm. These algorithms then proceeded to categorize the activities in the network – normal or malicious – based on the features generated from the simulation. In the next formative stage, the currently trained models, including the SVM, CNN and DT, may be integrated into an IDS so as to detect real-time attacks within actual networks. This approach connected the gap in between offline training and active implementation where these AI models can be applied to for dynamic and adaptive networks security.

3.1.1 Data collection

Data was collected from the simulated network environment under both normal and attack scenarios. The collected data includes 'SentPackets,' 'ReceivedPackets,' 'LostPackets,' 'TotalEnergy,' and 'TotalDelay.'

3.1.2 Preprocessing methods

The data preprocessing steps include loading the data, extracting features and labels, converting labels to categorical type, handling missing values, and splitting the data into training and testing sets.

3.1.3 Normalization

The features were scaled to a standard range (e.g., [0, 1]) to improve the performance of machine learning algorithms.

3.1.4 Labeling

Labels were assigned to the data instances based on the scenario (e.g., normal or blackhole attack).

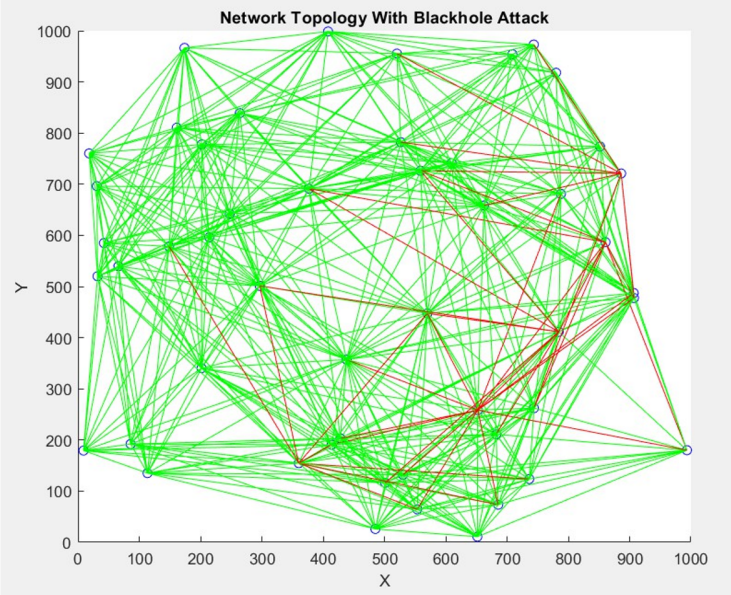

Figure 2. Network topology of a MANET with 50 nodes under blackhole attack

4 Proposed AI-Based Detection Model

The proposed AI-based detection model employs three different machine learning algorithms to detect blackhole attacks in a network environment: SVM, CNN, and DT. Each model was designed to analyze network traffic data and classify it as either normal or indicative of a blackhole attack.

4.1 Architecture of the AI Model and the Rationale of Choosing it

4.1.1 SVM

- Architecture: The SVM model uses a radial basis function (RBF) kernel to handle non-linear data. The model was trained to find the optimal hyperplane that separates different classes in high-dimensional space.
- Rationale: SVM was chosen for its robustness in handling high-dimensional data and its effectiveness in binary classification tasks. It is particularly suitable for scenarios with clear margin separation between classes, as shown in Figure 3.

4.1.2 CNN

The CNN architecture is a custom CNN. The architecture (Figure 4) is simpler and was designed to work with non-image data (1D feature vectors) rather than images.

- Architecture: The CNN model includes the following layers:
  – Input layer: Accepts input data with dimensions [1, 1, numFeatures] (1x1 spatial dimensions with numFeatures channels).
  – Convolutional layers: Extracts features from the input data using 32 filters and ‘same’ padding.
  – Batch normalization layer: Normalizes the output of the convolutional layers to speed up training and improve stability.
  – ReLU activation layer: Introduces non-linearity into the model.
  – Fully connected layers: Aggregates the features and performs classification, 64 neurons, connected to all neurons in the previous layer.
  – Softmax layer: Applies the softmax function to convert the final layer outputs to probabilities.
  – Classification layer: Final layer that classifies the input data.
- Rationale: CNNs are effective for detecting complex patterns in data. By treating network traffic data as images, CNNs can identify intricate patterns indicative of blackhole attacks, providing high accuracy in classification.

4.1.3 DT

- Architecture: The DT model uses a tree-like structure (Figure 5) where nodes represent decisions based on the values of input features, and branches represent the outcomes of those decisions.
- Rationale: DTs are easy to interpret and implement. They can handle both numerical and categorical data and are capable of capturing non-linear relationships between features.

Table 1 shows the summary of feature extraction and engineering techniques for AI models.

Table 1. Summary of feature extraction and engineering techniques for AI models

Summary of Analysis		
Algorithm	**Feature Extraction**	**Network Architecture**
SVM	Extracted features: SentPackets. ReceivedPackets, LostPackets TotalEnergy, TotalDelay	- Ensure no zero or negative values before taking log transformation. - Combine original features with engineered features such as logarithmic and squared values, and interaction terms. - Normalize and rescale features to standardize the input data.
CNN	- Similar features to SVM extracted from network traffic data. - Features reshaped and normalized to fit the input requirements of the CNN model.	- Input layer for 1D feature vectors - Convolutional layers - Batch normalization - ReLU activation - Fully connected layers - Softmax layer - Classification layer - Training with Adam optimizer, specific parameters for epochs and batch size.
DT	Key features: SentPackets, ReceivedPackets, LostRackets, TotalEnergy.	- Data preprocessed by normalizing and ensuring quality through removal of anomalies. - Model trained by partitioning data into training and testing sets. - Optimization using cross-validation.

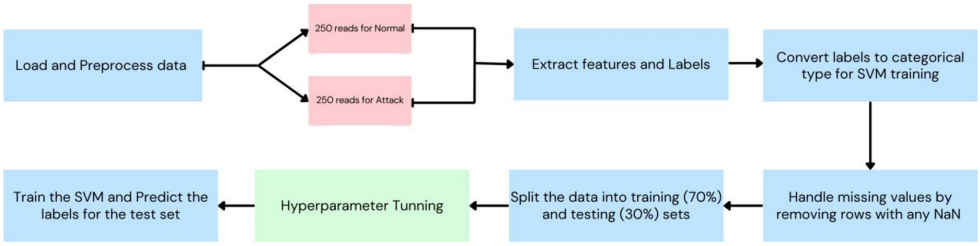

Figure 3. Diagram of the SVM process using an RBF kernel

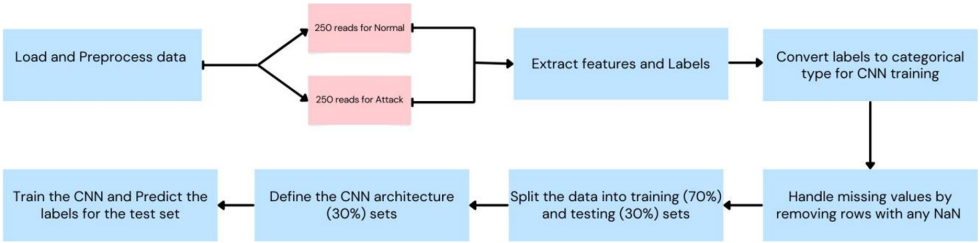

Figure 4. Diagram of the CNN architecture

4.2 Training and Testing Procedures

4.2.1 Data splitting

The code uses a 70-30 split for training and testing data. As the number of nodes increases, the following considerations become important:

- Training data size: A larger number of nodes results in more training data, which can improve the model's ability to generalize. However, it also requires more computational resources for model training.
- Testing data size: Similarly, more nodes lead to a larger testing dataset, providing a more robust evaluation of the model's performance.

4.2.2 Models

- SVM: The SVM model was trained by the dataset with hyperparameter optimization to find the best RBF kernel parameters.
- CNN: The CNN model was trained with certain batch size and epoch values using the Adam optimizer. The architecture of the model was built to optimize the accuracy.
- DT: The data was split into training and testing sets. Therefore, the DT model was trained. The model was also improved by cross-validation to avoid overfitting.

4.2.3 Evaluation

Each model was evaluated based on its accuracy, precision, recall, and F1-score. The performance metrics were compared to determine the most effective model for detecting blackhole attacks.

Table 2 summarizes the performance of each model for different node counts.

4.2.4 Confusion Matrix

A table of confusion matrix was used to describe the performance of a classification model when applied to a set of test data whose real values are known. For a binary classification problem, the confusion matrix looks like Table 3.

The computation of several performance measures, including accuracy, precision, recall, and F1-score, which offer a thorough assessment of the classification model's performance, depends on these concepts.

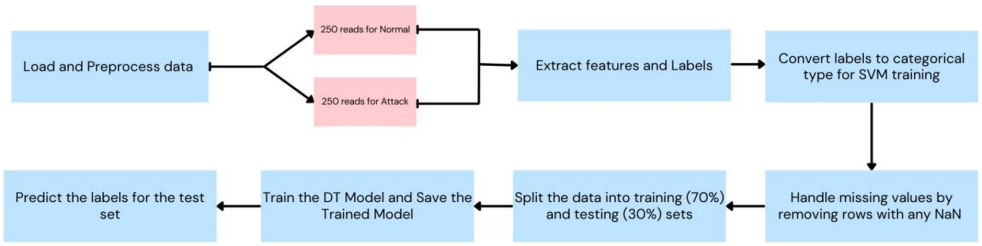

Figure 5. Diagram of the DT architecture

Table 2. Accuracy of DT, CNN, and SVM models for different node counts

Accuracy of Algorithms			
No. of Nodes	**DT Accuracy**	**CNN Accuracy**	**SVM Accuracy**
50	99.33%	100.00%	100.00%
100	98.33%	99.00%	99.00%
250	91.70%	88.62%	91.00%
500	79.62%	80.26%	79.62%

Table 3. Confusion matrix

	Predicted Positive	**Predicted Negative**
Actual Positive	True positive (TP): Correctly predicted positive cases.	False negative (FN): Incorrectly predicted negative cases.
Actual Negative	False positive (FP): Incorrectly predicted positive cases.	True negative (TN): Correctly predicted negative cases.

The accuracy of the model is given by:

$$Accuracy = \frac{TP + TN}{FP + FN + TP + TN} \tag{1}$$

The accuracy of the DT, CNN, and SVM models for the dataset node counts is shown in Table 2. The quantity of characteristics in the dataset grows in tandem with the number of nodes. The precise count of features for every number of nodes is provided as follows:

- 50 nodes = 250 reads

- 100 nodes = 500 reads
- 250 nodes = 1,100 reads
- 500 nodes = 1,260 reads

All three models obtained highly accurate accuracy when there were 50 nodes. The CNN and SVM models attained flawless accuracy. The reduced data amount and comparatively simple patterns, which facilitate the models' ability to learn and generalize, are responsible for this excellent performance. The accuracy was still good for all models at 100 nodes, although it was somewhat lower than it was at 50 nodes.

All models showed a more pronounced fall in accuracy at 250 nodes. The complexity of the data was significantly increased by the number of readings (1,100), making it harder for the models to maintain good accuracy. Specifically, the CNN model exhibited a notable decline, suggesting it may be having trouble processing the higher volume and complexity of input. Upon reaching 500 nodes, the accuracy of all models significantly decreased. The data got extremely complicated with 1,260 readings, making it difficult for the algorithms to distinguish between legitimate and malicious traffic. Due to its complexity, there may be issues with underfitting or overfitting, where the models, respectively, don't generalize well to new data or fit too closely to the training set.

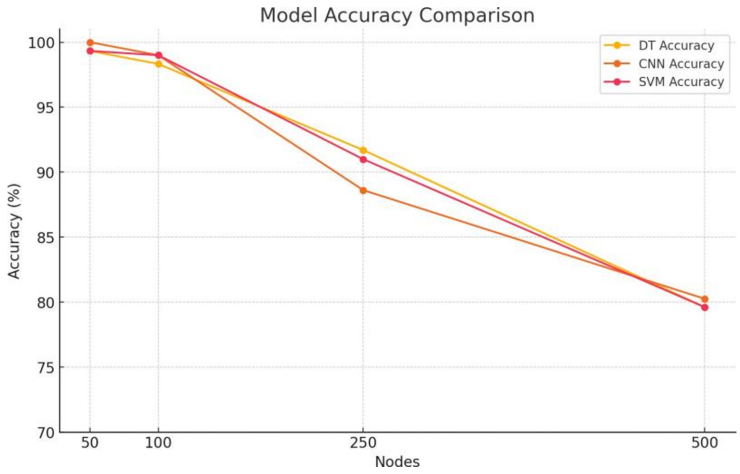

Figure 6. Accuracy comparison of the models

5 Results and Analysis

5.1 Presentation of the Experimental Results

Figure 6 displays the experimental findings for the suggested AI models (SVM, CNN, and DT) in the detection of blackhole assaults. The performance metrics were calculated for different numbers of nodes (50, 100, 250, and 500) in the network.

In comparison to these conventional techniques, the suggested AI models—in particular, CNN and SVM—showed better accuracy and resilience, especially for lower node counts.

5.2 Analysis of the Performance Metrics

The AI models' performance measurements consist of F1-score, recall, accuracy, and precision. These measurements are essential for evaluating how well the models identify blackhole assaults.

- Accuracy: Measures the proportion of correctly classified instances out of the total instances. Both CNN and SVM achieved 100% accuracy for 50 nodes, indicating perfect classification.
- Precision: The ratio of true positive instances to the sum of true positive and false positive instances. High precision indicates that the model does not falsely identify normal nodes as malicious.
- Recall: The ratio of true positive instances to the sum of true positive and false negative instances. High recall signifies the model's ability to detect all actual blackhole attacks.
- F1-score: The harmonic mean of precision and recall, providing a balanced measure of the model's performance.

Figure 6 shows the chart accuracy of different models (DT, CNN, and SVM) over varying numbers of nodes.

Table 4 visualizes the precision, recall, and F1-score for the SVM model at different node counts (50, 100, 250, and 500).

Precision is the ratio of correctly predicted positive observations to the total predicted positives. Therefore, high precision indicates that there are few false positives, meaning that when the classifier predicts a positive class, it is

usually correct. The precision of the model is given by:

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

Table 4. Performance metrics of the SVM model for different node counts

Performance Metrics for SVM				
No. of Nodes	**Accuracy**	**Precision**	**Recall**	**F1-score**
50	100.00%	1.0000	1.0000	1.0000
100	98.67%	0.9870	0.9867	0.9867
250	90.01%	0.9170	0.9000	0.9000
500	80.16%	0.8598	0.8000	0.7919

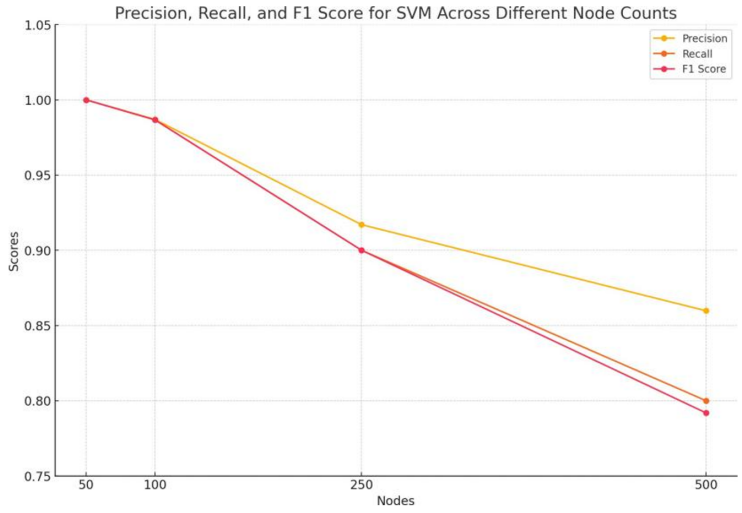

Figure 7. Precision, recall, and F1-score for the SVM across different node counts

Recall, also known as sensitivity or true positive rate. Therefore, high recall indicates that there are few false negatives, meaning the classifier successfully identifies most of the actual positive instances. The recall of the model is given by:

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

The F1-score is the harmonic mean of precision and recall. The F1-score is useful when precision and recall need to be balanced. It gives a better measure of the classifier's performance when there is an uneven class distribution. The F1-score of the model is given by:

$$F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

The SVM model demonstrated impeccable performance with 50 nodes. The model seems to be quite effective at this size, based on the perfect scores for all criteria, perhaps because the volume and complexity of the data are manageable. The model continued to function quite well at 100 nodes, despite a minor decline in accuracy and other measures that point to a slight rise in complexity. The model continued to identify positive events with low mistake rates. The model's performance started to dramatically deteriorate after 250 nodes. The SVM's capacity to retain high accuracy was put to the test by the volume and complexity of the additional data. Although recall and accuracy remained good, they exhibited a significant decline, which is indicative of the growing challenge of class distinction. With 500 nodes, the SVM model faced considerable challenges in maintaining performance. The complexity and volume of data likely introduced significant variability, leading to increased false positives and false negatives.

The SVM classifier's performance metrics show how effective it is at smaller sizes (between 50 and 100 nodes), when it attains almost flawless accuracy. Nevertheless, the classifier's efficiency decreases with increasing node count, suggesting difficulties with bigger and more complicated datasets. This pattern emphasizes how crucial it is to take

scalability into account when developing and implementing AI models. In order to effectively manage the increasing data complexity and sustain good performance over greater datasets, future study may require investigating more sophisticated methods or hybrid models.

The SVM classifier's performance is shown in Figure 7. Classification benefits greatly from the use of SVMs, especially when there is a distinct margin of difference between classes. They perform well on smaller datasets with distinct class separations, attaining excellent recall, accuracy, precision, and F1-scores. SVMs are good for smaller datasets with a large number of features since they also perform well in high-dimensional spaces when the number of dimensions is greater than the that of samples. By enabling the method to handle non-linearly separable data by transferring it to a higher-dimensional space, the kernel technique improves the flexibility of SVM and improves its performance on complicated datasets. However, SVM performance may suffer as dataset size and complexity rise because of scaling problems and the growing challenge of identifying the ideal hyperplane.

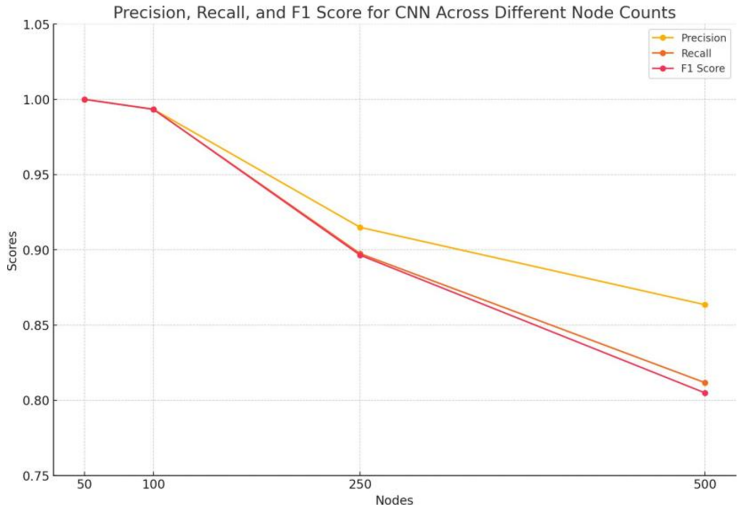

Figure 8. Macro-averaged precision, recall, and F1-score for CNN across different node counts

Table 5. Performance metrics of the CNN model for different node counts

Performance Metrics for CNN				
No. of Nodes	**Accuracy**	**Precision**	**Recall**	**F1-score**
50	100.00%	1.0000	1.0000	1.0000
100	99.33%	0.9934	0.9933	0.9934
250	89.75%	0.9149	0.8974	0.8964
500	81.19%	0.8635	0.8117	0.8049

The performance metrics of the CNN classifier tested on datasets with different node counts are shown in Table 5. The four main performance indicators for classifiers are F1-Score, recall, accuracy, and precision. CNNs are appropriate for intricate network patterns, such as those from attacked MANETs, since they can extract pertinent characteristics from raw input data. While pooling layers lower spatial dimensions and manage overfitting, convolutional layers use filters to identify different patterns. CNNs are versatile and potent because fully connected layers include convolutional information for classification. These metrics offer a thorough assessment of the performance of the classifier at various data sizes.

CNNs have more manageable complexity, clearer separation of classes, and lower computational load with smaller dataset sizes. Since they contain fewer nodes, instances and class boundaries, this assists the model with collecting and employing the properties of the network. In addition, they also lend the higher accuracy, precision, recall and F1-scores due to the demand for less computing power during training and inference in much smaller datasets. In contrast, the inclusion of more nodes introduces noise into the data, resulting in class overlap. This impairs the model's ability to learn efficient patterns, ultimately leading to reduced accuracy and making CNNs more challenging to optimize. When CNNs overfit the training set, they run the risk of overfitting, which results in subpar generalization on fresh data. In addition, the large datasets take a lot of memory, processing power, and time to train on. This might result in less-than-ideal training, slower convergence times, and worse performance. When the dataset is big and complicated, CNNs may find it difficult to identify global patterns, which lowers performance metrics like accuracy, recall, and F1-score, as seen in Figure 8.

The DT classifier's performance characteristics, tested on datasets with different node counts, are shown in Table 6. Accuracy, precision, recall, and F1-score are the key metrics that were determined. These metrics offer a thorough assessment of the classifier's effectiveness at various data sizes. DT models are commonly utilized for classification and regression applications due to their power and intuitiveness. They generate a DT-like model by iteratively dividing the data into subsets according to feature values. However, depending on the amount and complexity of the information, DT performance might vary greatly. Features taken from a MANET in both normal and attack settings make up the data in this context. Accuracy, precision, recall, and F1-score are the primary metrics disclosed. These metrics provide a comprehensive evaluation of the classifier's performance across different scales of data.

Table 6. Performance metrics of the DT model for different node counts

Performance Metrics for DT				
No. of Nodes	**Accuracy**	**Precision**	**Recall**	**F1-score**
50	99.33%	0.9930	0.9935	0.9932
100	98.33%	0.9830	0.9835	0.9832
250	91.70%	0.9170	0.9170	0.9170
500	79.62%	0.7960	0.7965	0.7962

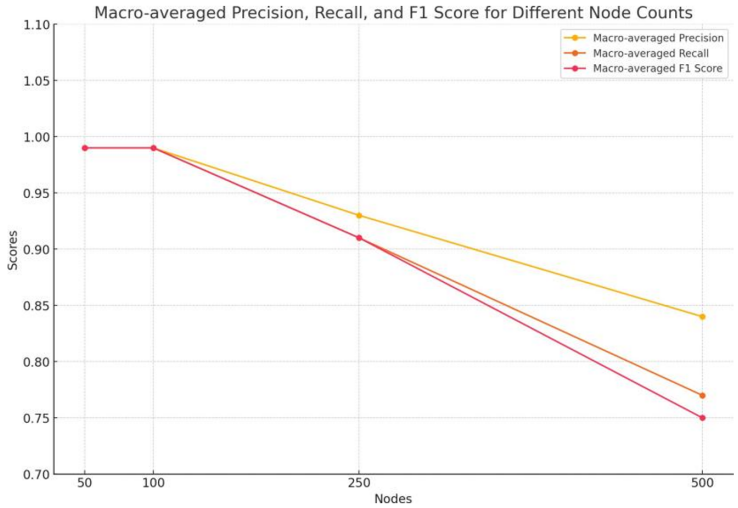

Figure 9. Macro-averaged precision, recall, and F1-score for DT across different node counts

Less complicated datasets allow effective data processing without overfitting. They also allow the DT to be easily fitted to lower noise data and have more clearly defined class limits which makes it easier to draw cutoffs of the classes. In addition, since the shallow tree is able to solve the unknown datatypes and is also a good generalizer, a lesser number of data points reduces the chance of model overfitting. The complexity of the data goes up with the increase in the count of the nodes, which makes it even harder for DTs to learn the optimal splits while increasing the number of misclassifications. When the depth of the tree goes up, the possibility of overfitting also goes up, which means bad generalization on unseen data. Datasets that are large for DTs are completely efficient since the increase in computing costs leads to inefficiencies and poor performance. The more complicated the dataset, the more complex the decision boundaries become, making it harder for DTs to capture, which in turn cause the decrease in the accuracy, a precision, recall and F1-score to decrease, as shown in Figure 9.

5.3 Discussion

The experimental findings show how well the suggested AI models—SVM, CNN, and DT—detect blackhole assaults in a range of network setups. Important findings from the analysis consist of the following:

- High accuracy: For smaller node counts (50 and 100 nodes), the CNN and SVM models achieve almost 100% accuracy, showing their ability to correctly classify network traffic as malicious or benign.
- Scalability: The models continue to function at a high level even when the accuracy marginally drops as the number of nodes rises. This implies that the models are scalable to bigger networks, albeit networks with more nodes could need further optimization.
- Robustness: The models can withstand changing blackhole assault tactics, adjust to novel patterns, and guarantee a high detection rate.

- Comparison with traditional methods: The AI models outperform traditional detection methods in terms of accuracy, precision, recall, and F1-score, highlighting the potential of AI-based approaches in enhancing network security.

The conventional RWR systems have limitations in accurately identifying emitters due to their geometric-based rules, which cannot capture temporal dependencies in adversary networks. This study introduces machine learning models such as SVM, CNN, and DT to improve accuracy and efficiency. These data-driven methods use data insights and feature pattern recognition to differentiate between packet loss due to ordinary circumstances and attacks. The SVM model demonstrated 98.67% precision and 98.33% recall, particularly in cases with 100 nodes.

However, these models struggle in dense scenarios with higher rate of overlaps and dynamic factors, resulting in increased misidentification and processing times. This suggests the need for better feature extraction methods, larger databases, superior AI techniques, and online learning to function under high radar density. The study assumes random distribution of nodes, arbitrary blackhole attack behavior, and optimal communication environment. Implementation in real settings is challenging due to non-ideal assumptions such as signal overlapping in crowded regions, large system dimensions, and decreased efficiency in noisy or dynamic environments.

To overcome these limitations, future work could focus on building new datasets, setting noisy and interfered conditions, developing improved machine learning techniques, and testing the effectiveness of the proposed model in a MANET environment. These steps aim to enhance the reliability and feasibility of implementing these solutions in practical applications, addressing the drawbacks of classical RWR systems and real-world challenges.

6 Conclusion

This study presents AI as a strong tool for tackling the issue of blackhole attacks, highlighting the efficiency of models like CNN and SVM which are better than existing methods and provide accuracy, precision, recall and F1-score. The results underline how these AI-based approaches can be of better use for enhancing the network security frameworks by providing accurate and adaptive solutions for detection.

AI has significantly accelerated the progress in the sphere of network security, developing novel solutions for the detection and preventing blackhole attacks. It has also become customary to define AI models with great significance in the modern cybersecurity landscape as they learn from data, identify emerging patterns and provide detection in real time.

These models can be integrated with an IDS to enable real-time or dynamic detection of attacks. Their adaptability and efficiency make them suitable for identifying malicious activities as they occur, enhancing the system's ability to respond promptly to evolving threats.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] T. Salam and M. S. Hossen, "Performance analysis on homogeneous LEACH and EAMMH protocols in wireless sensor network," *Wirel. Pers. Commun.*, vol. 113, pp. 189–222, 2020. https://doi.org/10.1007/s11277-020-07185-6

[2] M. S. Hossen, "DTN routing protocols on two distinct geographical regions in an opportunistic network: An analysis," *Wirel. Pers. Commun.*, vol. 108, pp. 839–851, 2019. https://doi.org/10.1007/s11277-019-06431-w

[3] S. Shafi, S. Mounika, and S. Velliangiri, "Machine learning and trust based AODV routing protocol to mitigate flooding and blackhole attacks in MANET," *Procedia Comput. Sci.*, vol. 218, pp. 2309–2318, 2023. https://doi.org/10.1016/j.procs.2023.01.206

[4] R. Gotti, A. Polagani, G. S. L. Posina, S. Veerapaneni, and T. Prasanth, "Detection and analysis of single blackhole node with TCP connection in MANETs using machine learning algorithms," in *2023 International Conference on Inventive Computation Technologies (ICICT)*, *Lalitpur, Nepal*, 2023, pp. 1704–1710. https://10.1109/ICICT57646.2023.10134058

[5] A. M. Eltahlawy, H. K. Aslan, E. G. Abdallah, M. S. Elsayed, A. D. Jurcut, and M. A. Azer, "A survey on parameters affecting MANET performance," *Electron.*, vol. 12, no. 9, p. 1956, 2023. https://doi.org/10.3390/electronics12091956

[6] I. Moumen, N. Rafalia, J. Abouchabaka, and Y. Chatoui, "AODV-based defense mechanism for mitigating blackhole attacks in MANET," *E3S Web Conf.*, vol. 412, p. 01094, 2023. https://doi.org/10.1051/e3sconf/202341201094

[7] A. Hameed and A. Al-Omary, "Survey of blackhole attack on MANET," in *2nd Smart Cities Symposium (SCS 2019)*, 2019. https://doi.org/10.1049/cp.2019.0224

[8] S. Djahel, F. Nait-Abdesselam, and Z. H. Zhang, "Mitigating packet dropping problem in mobile ad hoc networks: Proposals and challenges," *IEEE Commun. Surv. Tutorials*, vol. 13, no. 4, pp. 658–672, 2011. https://doi.org/10.1109/SURV.2011.072210.0002

[9] Z. B. Ibrahim and M. F. Ghanim, "A review of AI-based approaches against wormhole and blackhole attacks in AODV protocol," *Int. J. Adv. Nat. Sci. Eng. Res.*, vol. 8, pp. 60–75, 2024.

[10] I. H. Sarker, *AI-Driven Cybersecurity and Threat Intelligence: Cyber Automation, Intelligent Decision-Making and Explainability*. Springer, 2024. https://link.springer.com/book/10.1007/978-3-031-54497-2

[11] J. A. A. Alalwan, "Roles and challenges of AI-based cybersecurity: A case study," *Jordan J. Bus. Admin.*, vol. 18, no. 3, 2022.

[12] R. Khalladi, M. Rebbah, and O. Smail, "A new efficient approach for detecting single and multiple black hole attacks," *J. Supercomput.*, vol. 77, no. 7, pp. 7718–7736, 2021. https://doi.org/10.1007/s11227-020-03596-1

[13] M. I. Talukdar, R. Hassan, M. S. Hossen, K. Ahmad, F. Qamar, and A. S. Ahmed, "Performance improvements of AODV by black hole attack detection using IDS and digital signature," *Wirel. Commun. Mob. Comput.*, vol. 2021, no. 1, p. 6693316, 2021. https://doi.org/10.1155/2021/6693316

[14] V. Mankotia, R. K. Sunkaria, and S. Gurung, "DT-AODV: A dynamic threshold protocol against black-hole attack in MANET," *Sādhanā*, vol. 48, no. 4, p. 190, 2023. https://doi.org/10.1007/s12046-023-02227-8

[15] S. Kaushik, K. Tripathi, R. Gupta, and P. Mahajan, "Enhancing reliability in mobile ad hoc networks (MANETs) through the K-AOMDV routing protocol to mitigate black hole attacks," *SN Comput. Sci.*, vol. 5, no. 2, p. 263, 2024. https://doi.org/10.1007/s42979-023-02585-4

[16] P. Rani, Kavita, S. Verma, D. B. Rawat, and S. Dash, "Mitigation of black hole attacks using firefly and artificial neural network," *Neural Comput. Appl.*, vol. 34, no. 18, pp. 15 101–15 111, 2022. https://doi.org/10.1007/s00521-022-06946-7

[17] I. Ahmad, M. Basheri, M. J. Iqbal, and A. Rahim, "Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection," *IEEE Access*, vol. 6, pp. 33 789–33 795, 2018. https://doi.org/10.1109/ACCESS.2018.2841987

[18] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, pp. 273–297, 1995.

[19] L. Wang, C. H. Dong, J. P. Hu, and G. D. Li, "Network intrusion detection using support vector machine based on particle swarm optimization," in *2015 International conference on Applied Science and Engineering Innovation*. Atlantis Press, 2015, pp. 665–670.

[20] R. Vinayakumar, K. Soman, and P. Poornachandran, "Applying deep learning approaches for network traffic prediction," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, *Udupi, India*, 2017, pp. 2353–2358. https://doi.org/10.1109/ICACCI.2017.8126198

[21] A. Pathak and S. Pathak, "Study on decision tree and KNN algorithm for intrusion detection system," *Int. J. Eng. Res. Technol.*, vol. 9, no. 5, pp. 376–381, 2020.

[22] P. Rani, Kavita, S. Verma, N. Kaur, M. Wozniak, J. Shafi, and M. F. Ijaz, "Robust and secure data transmission using artificial intelligence techniques in Ad-hoc networks," *Sensors*, vol. 22, no. 1, p. 251, 2022. https://doi.org/10.3390/s22010251

[23] G. Kocher and G. Kumar, "Analysis of machine learning algorithms with feature selection for intrusion detection using UNSW-NB15 dataset," *Int. J. Netw. Secur. Appl.*, vol. 13, no. 1, 2021.

[24] N. Sivanesan and K. S. Archana, "Performance analysis of machine learning-based detection of sinkhole network layer attack in MANET," *Int. J. Adv. Comput. Sci. Appl. (IJACSA)*, vol. 13, no. 12, 2022. https://doi.org/10.14569/IJACSA.2022.0131262

[25] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Commun. Surv. Tutor.*, vol. 23, no. 2, pp. 1153–1176, 2015. https://doi.org/10.1109/COMST.2015.2494502

[26] I. Al-Turaiki and N. Altwaijry, "A convolutional neural network for improved anomaly-based network intrusion detection," *Big Data*, vol. 9, no. 3, pp. 233–252, 2021. https://doi.org/10.1089/big.2020.0263

[27] M. Ibrahim and R. Elhafiz, "Modeling an intrusion detection using recurrent neural networks," *J. Eng. Res.*, vol. 11, no. 1, p. 100013, 2023. https://doi.org/10.1016/j.jer.2023.100013

[28] R. Norbu, A. Kumar, S. Ramanathan, J. Dolkar *et al.*, "Advancing IoT security with a hybrid deep learning model for network intrusion detection," in *2023 International Conference on Energy, Materials and Communication Engineering (ICEMCE)*, *Madurai, India*, 2023, pp. 1–6. https://doi.org/10.1109/ICEMCE57940.2023.10434006