Detection of Cyber Attacks Using AI/ML

Sunita Patil, Kirti Agarwal, Tanvi Baviskar, Prakhar Pandey and Dev Phadol
Dr. D. Y. Patil Institute of Technology, Savitribai Phule Pune University,
Pimpri, Pune, India

Keywords: DDoS, Ransomware, AI-Powered Platform, Cybersecurity, Machine Learning, Real-Time Detection, Threat Defense, Network Traffic Analysis, Anomaly Detection.

Abstract: DDoS, as well as ransomware, is regarded as emerging threats in the modern digital platform. These forms of attacks could be exploited to cripple major businesses and organizations by disrupting business processes, significant financial losses, and compromise of sensitive information. Traditionally, the adoption of these security systems was not made since the threats are changing fast. To mitigate the foregoing challenge, we hereby proffer the development of an AI smart platform that would be able to identify and respond in real-time to DDoS and ransomware attacks. This platform shall primarily depend upon the use of ML(machine learning) to understand a network and its systems' baseline behavior; thus, it can indicate anomalies that may signify potential threats. By having analysis of traffic and monitoring file activity, the solution can alert about unusual patterns and react in real-time by giving alarms or starting defense mechanisms. This solution suggested can be scalable and flexible, bringing not only rapid detection but also proactive defense capabilities for organizations to be ahead of the cyber attackers. The main objective of this platform is a means by which organizations can become more resilient and perhaps take steps forward in improving their state of resilience toward digital attacks.

1 INTRODUCTION

As dependency on digital systems grows, individuals and organizations are constantly at risk to a host of attacks, with DDoS and ransomware being the most disruptive of them. Such attacks might severely disrupt business operations, causing huge losses and unauthorized exposure of sensitive information. Traditional information security systems are generally ineffective to eliminate these types of attacks due to their inability to keep abreast with the rapidly changing tactics employed by cyber hackers.

Research conducted by (Smith and Doe, 2020) emphasized the importance of utilizing SVM and Naive Bayes for detecting DDoS attacks, achieving 85% accuracy in identifying malicious network traffic. Their work, however, lacked real-time detection capabilities. Similarly, (Zhang and Wang, 2021) explored Deep Learning and CNN techniques for ransomware classification, obtaining high accuracy but struggling with broader scalability issues in detecting new patterns. The goal of this project is to address this important question by presenting an AI-driven system designed to detect and respond immediately to DDoS and ransomware attacks. The proposed system leverages historical data on cyber-attack traffic and system activities to train machine learning models for identifying abnormal behaviors and patterns in network traffic and system activities (Yadav and Singh, 2023). Fast and automated responses from the system to suspicious activities help minimize the damage caused by these cyberattacks (Smith and Taylor, 2022). This AI platform is positioned to strengthen business operations and safeguard data from falling into the hands of hackers in an increasingly unstable digital environment (Johnson, 2024).

2 LITERATURE SURVEY

2.1 Traditional Systems of Cyber Attack Detection:

- **Signature-Based Detection:**
  Signature-based intrusion-detection systems scan for known attack patterns or signatures in network traffic. The most widely used commercial cyber-

Patil, S., Agarwal, K., Baviskar, T., Pandey, P. and Phadol, D.
Detection of Cyber Attacks Using AI/ML.
DOI: 10.5220/0013586100004664

In *Proceedings of the 3rd International Conference on Futuristic Technology (INCOFT 2025) - Volume 2*, pages 22-30
ISBN: 978-989-758-763-4

security tools incorporate these systems, proving highly effective against known threats but less capable of addressing unknown and evolving threats like zero-day attacks. This is because their models depend on predefined signatures and have limited flexibility, thus requiring constant updates of their signature database (Ferdous et al., 2023).

- **Rule-Based Systems:**

  Rule-based systems rely on predetermined rules used to define malicious activities. They are widely implemented within firewalls or intrusion prevention systems (IPS). While effective for identifying particular types of threats such as DDoS or SQL injection attacks, they are highly susceptible to false positives and cannot keep up with the rapid evolution of cyber threats (Pei et al., 2023).

- The traditional anomaly detection methods that are in use mostly rely on predefined thresholds and predefined rules for determining outliers over normal traffic patterns. While these methods can identify unknown threats, they also generate a large volume of false positives, causing system inefficiency and alert fatigue (Alshehri et al., 2023).

3 RESEARCH GAP IDENTIFICATION

While enormous strides have been made in AI-based approaches to cybersecurity, several foundational challenges restrict the full effective utilization of current systems. The key research gaps are as follows:

1. **Lack of Quality Datasets:**

   Most existing models rely on outdated or underdeveloped datasets, which reduces the precision and efficiency of the models in detecting sophisticated modern cyberattacks. This limitation makes generalization challenging and hinders the discovery of new attack patterns in dynamic environments. (Dyari and Alshehri, 2021).

2. **Adversarial Vulnerability:**

   Machine learning models are susceptible to attacks called adversarial attacks, where input data is deliberately could be manipulated to mislead the model into making incorrect predictions. This vulnerability has critical implications in cybersecurity as the attackers would use these weak spots to overcome defense and compromise systems (Aktar and Show, 2023)

Table 1: Provides an overview of several research papers related to cyberattack detection and prevention techniques.

Author(s) & Year	**Objective**	**Algo.**	**Outcomes**	**Limitation**
Smith et al. (2020)	Detect DDoS attacks using network traffic analysis	SVM, Naive Bayes	85% accuracy in detecting DDoS	Lacked real-time analysis
Zhang et al. (2021)	Classify ransomware activities	Deep Learning, CNN	90% accuracy on test data	Limited to specific ransomware patterns
Lee et al. (2019)	Hybrid model for multi-type attack detection	SVM, KNN, Random Forest	Improved detection rate for multiple attack types	High false positives
Patel et al. (2022)	Real-time monitoring and detection of DDoS attacks	Naive Bayes, Decision Tree	80% detection accuracy in real-time	High computational cost
Gomez et al. (2023)	Enhanced ransomware detection using ensemble methods	SVM, Neural Networks	92% accuracy with ensemble	Scalability issues with large datasets
Kumar et al. (2021)	Detection of advanced persistent threats	Decision Trees, Random Forest	88% detection accuracy	Lacked adaptability to new threats

3. **Scalability Problem:**

   Current AI models face practical difficulties when scaled for real-world environments. Many models exhibit nonlinear behavior and become inefficient when dealing with large network traffic. This scalability issue limits the practical applicability of AI-based solutions in high-demand cybersecurity scenarios (Pei et al., 2023).

4 PROBLEM STATEMENT

To help businesses and individuals defend against the rising threat of cyberattacks like DDoS and Ransomware, there's a growing need for a smarter, AI-powered solution. Traditional security systems often fall short, as attackers constantly evolve their methods, making it harder to detect and stop these threats in time. By creating a platform that can quickly identify and respond to these attacks in real-time, we can better protect operations, reduce financial risks, and secure sensitive information from being compromised.

5 PROPOSED MODEL

The proposed system incorporates multiple machine learning algorithms, each designed to enhance the detection and classification of cyberattacks like DDoS(Distributed Denial of Service) and ransomware. These algorithms work in an organized and efficient manner so that proper and timely threats can be identified. Below is the description of the algorithms used and their roles in the system:

5.1 Support Vector Machine (SVM)

SVM is a powerful supervised learning algorithm commonly applied to both classification and regression tasks, though it is primarily designed for classification problems. It has found significant applications in cybersecurity, such as distinguishing normal network activity from malicious behavior, and is Well-suited for managing data with high dimensions (Ferdous et al., 2023).

SVM operation works by throwing the input data into a space of dimension n, where n denotes the number of features. The SVM subsequently identifies a decision boundary that can be any hyperplane dividing it between classes. For example, in network security, SVM can delineate a region differentiating benign and malicious traffic. The main goal of SVM is the maximization of margin, or the distance between the hyperplane and the nearest data points from both classes that is called support vectors. Maximizing the margin enhances model generalizability to new, unseen data (Shan et al., 2023).

5.1.1 Mathematical Representation

1.The equation of the hyperplane can be expressed as:

$$\mathbf{v}^T\mathbf{z}+c=0$$

where:

- $\mathbf{v}$ is the weight vector, orthogonal to the hyperplane,
- $\mathbf{z}$ is the input feature vector,
- c is the bias term.

2. Decision Boundary

For a classification problem, the decision boundary classifies the data into two classes:

$$w^T y+a \geq +1 \quad \text{for } x=+1 \quad \text{(positive class)},$$
$$w^T y+a \leq -1 \quad \text{for } x=-1 \quad \text{(negative class)}.$$

3. Margin Maximization

The margin in SVM is expressed as:

$$\frac{2}{\|w\|}$$

where $\|w\|$ represents the Euclidean norm of the weight vector. The objective of SVM is to maximize this margin while adhering to the following constraints:

$$y_i(w^T x_i+b) \geq 1 \quad \forall i,$$

where:

- y_i denotes the label of the i-th data point (either $+1$ or -1),
- x_i represents the i-th data point.

4. Slack Variables for Non-Linearly Separable Data

When perfect separation of data is not achievable, slack variables $\zeta_j \geq 0$ are introduced to permit some level of misclassification:

$$z_j(v^T u_j+b) \geq 1-\zeta_j \quad \forall j.$$

The revised optimization problem is then formulated as:

$$\min_{v,b,\zeta} \frac{1}{2}\|v\|^2 + C\sum_{j=1}^{m} \zeta_j,$$

where C serves as a regularization parameter, Balancing the trade-off between increasing the margin and minimizing classification errors.

5.1.2 Applications in Network Traffic Classification

SVM is highly effective for detecting malicious network activity because:

- It captures even subtle deviations in network traffic by maximizing the margin.

- It is highly effective with high-dimensional data, making it ideal for examining characteristics like IP packets, session durations, and port utilization.
- Non-linear kernels allow it to handle complex patterns in malicious activities.

By leveraging its ability to identify intricate boundaries in the data, SVM is a vital tool in cybersecurity and other domains requiring high accuracy and interpretability.

5.2 Naive Bayes

Naive Bayes is a statistical classifier based on Bayes' Theorem. It relies on the assumption that the presence of one attribute within a class is independent of the presence of other attributes, even if they may be interdependent. This assumption, known as the *naive assumption*, simplifies the computation, making Naive Bayes an efficient and effective algorithm for handling complex environments, such as network traffic analysis (Ferdous et al., 2023).

Naive Bayes uses historical data to estimate the probability of a network packet being normal or malicious. It takes into account diverse features such as packet size, time intervals, and other network attributes. Due to its simplicity and efficiency, Naive Bayes is well-suited for real-time intrusion detection systems that need to process large volumes of network traffic efficiently (Yadav and Singh, 2023).

5.2.1 Mathematical Representation

1. Bayes' Theorem:

$$P(C|F) = \frac{P(F|C)P(C)}{P(F)}$$

where:

- $P(C|F)$ is the posterior probability of class C (e.g., benign or fraudulent) given the features F,
- $P(F|C)$ is the likelihood of observing features F given class C,
- $P(C)$ is the prior probability of class C,
- $P(F)$ is the probability of observing the features F.

2. Independence Assumption: For a feature set $Z = \{z_1, z_2, \ldots, z_m\}$, Naive Bayes assumes the conditional independence of the features:

$$P(Z|Y) = P(z_1|Y) \cdot P(z_2|Y) \cdot \cdots \cdot P(z_m|Y).$$

3. Classification Rule: The predicted class Y is determined as:

$$Y = \arg\max_{Y_k} P(Y_k) \prod_{i=1}^{m} P(z_i|Y_k)$$

where:

- Y_k represents the potential classes,
- $P(Y_k)$ is the prior probability of class Y_k,
- $\prod_{i=1}^{m} P(z_i|Y_k)$ is the product of the conditional probabilities of the features given class Y_k.

5.2.2 Advantages of Naive Bayes

- **Efficiency and Ease of Use:** The algorithm is highly efficient, making it ideal for rapid, real-time intrusion detection..
- **Scalable:** Naive Bayes can process large amounts of network data effectively, offering scalability in ever-changing environments.
- **Low Resource Consumption:** It operates with minimal computational power while still achieving high accuracy.

5.2.3 Applications in Network Traffic Analysis

Naive Bayes excels in network traffic analysis due to its probabilistic foundation and ability to model diverse features. It can identify anomalies and classify network packets as normal or malicious with high efficiency. The algorithm's adaptability makes it a preferred choice for lightweight and real-time intrusion detection systems.

5.3 K-Nearest Neighbors (KNN)

KNN is the distance-based algorithm used under supervised learning. Good success can be achieved on applying it to pattern recognition and classification problems: (Ahmed and Malik, 2023). In cybersecurity context, KNN identifies cyberattacks through observation of similarity of new data with their closest neighbors in training dataset. A data point's classification is Determined by the majority class of its k closest neighbors, allowing it to effectively detect both established and emerging attack patterns (Taylor and Nguyen, 2023). It is considered a lazy learning algorithm because it doesn't construct a model during the training phase. However, its limitation lies in delaying computation until the evaluation phase, where the new instance is compared with all training instances to determine its nearest neighbors (Patel and Gupta, 2023).

5.3.1 Mathematical Representation

Distance Metric: KNN utilizes a distance metric to find the nearest neighbors. Some commonly used distance measures include:

- **1. Euclidean Distance:**

  $$d(a_i,a_j) = \sqrt{\sum_{m=1}^{p} (a_{im} - a_{jm})^2},$$

  where a_i and a_j represent two data points in p-dimensional space.

- **2. Minkowski Distance:**

  $$d(a_i,a_j) = \left(\sum_{m=1}^{p} |a_{im} - a_{jm}|^q \right)^{\frac{1}{q}},$$

  where q is a parameter that defines the type of distance metric (e.g., $q = 2$ results in Euclidean distance).

- **3. Manhattan Distance:**

  $$d(a_i,a_j) = \sum_{m=1}^{p} |a_{im} - a_{jm}|,$$

  where $|a_{im} - a_{jm}|$ denotes the absolute difference between the m-th feature of a_i and a_j.

Parameter k: The choice of k is vital for the performance of the KNN algorithm:

- A smaller k can make the model more prone to noise.
- A larger k creates smoother decision boundaries but might reduce the distinction between classes.

5.3.2 Advantages of KNN

- **Pattern Recognition:** Efficiently detects recurring attack patterns using historical data.
- **Adaptability:** Quickly adapts to new attack trends due to its lazy learning nature, making it effective in dynamic environments.
- **Ease of Implementation:** Straightforward to implement and highly effective for detecting multiple types of attacks.

5.3.3 Applications in Cybersecurity

KNN is extensively applied in cybersecurity for tasks like anomaly detection and classification, including:

- Intrusion detection systems (IDS),
- Classification of malware,
- Detection of anomalies in network traffic.

By leveraging its ability to classify based on similarity and patterns, KNN is a robust tool for identifying known and emerging cyber threats.

5.4 Stacking (Ensemble Learning)

To improve overall performance, the suggested system utilizes stacking—an advanced ensemble learning method. This technique leverages the strengths of several base models (SVM, Naive Bayes, and KNN) to form a more precise and dependable classification system. The outputs from the base models are passed into a meta-model, like Random Forest, which integrates these inputs to generate the final prediction.

5.4.1 How Stacking Works

Each base model generates predictions for the input data, which are then aggregated by the meta-model. This hierarchical approach ensures that the system benefits from the unique strengths of each base algorithm, resulting in a more robust and adaptable solution.

5.4.2 Benefits of Stacking

- **Enhanced Accuracy:** Rectifies weaknesses of individual models by leveraging their combined strengths.
- **Adaptability:** Efficiently adapts to different attack patterns, providing comprehensive coverage against a variety of threats.
- **Robustness:** Capable of learning and evolving with the emergence of new and sophisticated cyberattacks.

5.5 Overall System Integration

By integrating these algorithms, the proposed model achieves a balance between speed, accuracy, and adaptability. SVM provides a solid foundation for binary classification, Naive Bayes ensures rapid and resource-efficient operation, KNN excels in recognizing patterns, and the stacking mechanism enhances reliability and overall detection capabilities. This ensemble approach equips the system to handle the dynamic and evolving landscape of cyber threats, offering real-time defense against DDoS and ransomware attacks.

6 METHODOLOGY

Figure 1 and 2 describes the System Architecture - Stage 1 and Stage 1 .

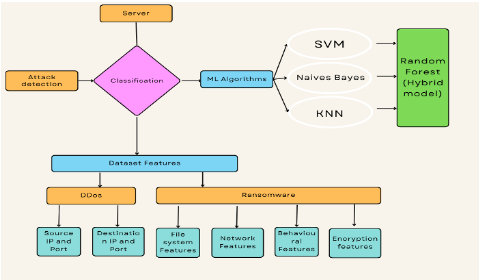

Figure 1: System Architecture - Stage 1

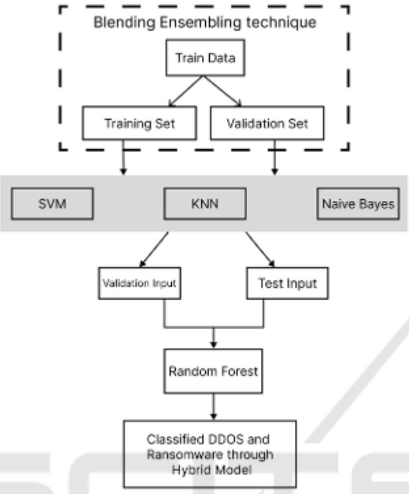

Figure 2: System Architecture - Stage 2

7 RESULTS

The ensemble model achieved an impressive accuracy rate of 99.91%.This high accuracy indicates that the model successfully classified the majority of instances, both attacks and non-attacks, with minimal errors. The false positive rate was also very low, at only 1%, meaning that only 1% of benign traffic was incorrectly labeled as malicious. The false negative rate was similarly low at 0.09%, indicating that almost all attacks were detected.

The model's real-time detection speed was impressive, with an average response time of 2 to 3 seconds during a simulated attack scenario. This quick detection is critical in preventing or minimizing damage when an attack is still in progress.

Scalability tests demonstrated that the system maintained its performance even under high network traffic volumes. As traffic increased, the ensemble model effectively identified attacks without significant performance degradation. This scalability ensures that the model can support growing networks while continuously providing robust protection against cyber attacks.

7.1 Support Vector Machine (SVM):

The SVM model demonstrated its effectiveness in distinguishing between normal and malicious traffic by creating a decision boundary that maximized the separation between the two classes. Its impressive accuracy stems from its ability to manage high-dimensional data and its robustness against overfitting. However, SVM can be resource-intensive, especially when dealing with large datasets.

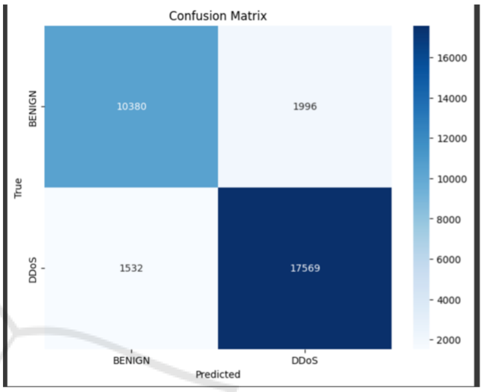

Figure 3: Graph of the SVM Model's Performance

7.2 Naive Bayes:

The Naive Bayes model performed well due to its simplicity and efficiency in handling high-dimensional data. By using probability-based classification, it was able to efficiently classify network traffic based on the likelihood of it being benign or malicious. While the model assumes feature independence, its performance remained strong, especially for real-time detection tasks.

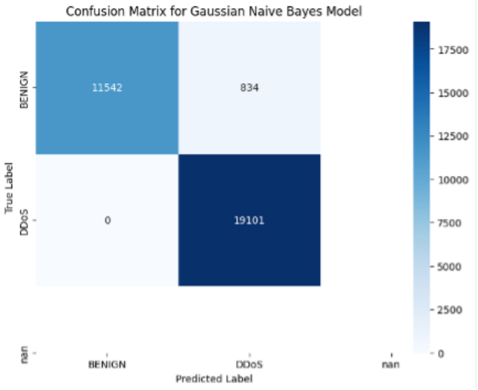

Figure 4: Graph of the Naive Bayes Model's Performance

7.3 K-Nearest Neighbors (KNN):

KNN performed well in identifying emerging attack patterns by comparing new data points with those in the training dataset. The model's classification process is straightforward, relying on the majority class of its nearest neighbors, making it easy to implement. However, its performance is highly dependent on the value of *k* and the selected distance metric, which can have a significant impact on the results.

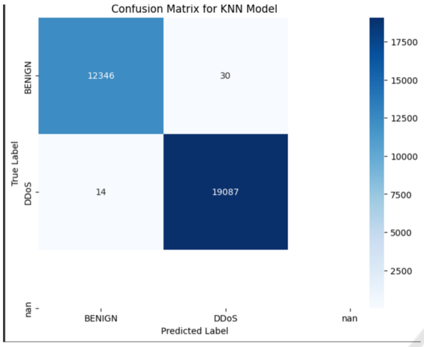

Figure 5: Graph of the KNN Model's Performance

7.4 Ensemble Model:

The ensemble model, which combines multiple machine learning algorithms, surpassed individual models in both accuracy and reliability. By synthesizing predictions from various classifiers, it offered a more dependable solution for detecting cyberattacks, particularly in intricate environments. This model achieved an impressive accuracy rate of 99.91%, showcasing exceptional real-time detection speed and scalability.

7.5 Comparison Between Existing Model and Proposed Ensemble Model:

We evaluate our proposed ensemble model against commonly used models in cybersecurity, including SVM, Naive Bayes, and KNN classifiers. The comparison results reveal that the ensemble model consistently outperforms the remaining models in terms of accuracy, false positive rate, and real-time detection speed, highlighting the advantages of combining multiple classifiers.

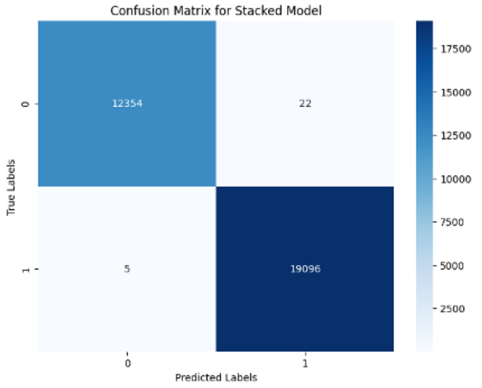

Figure 6: Graph of the Ensemble Model's Performance

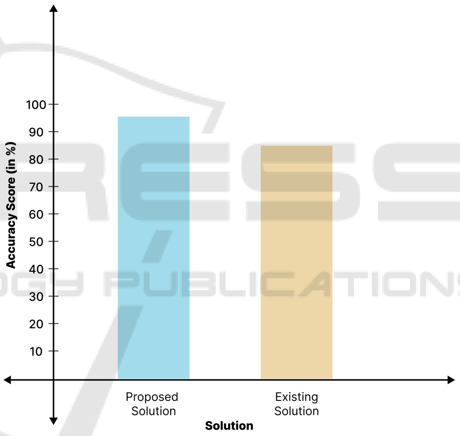

Figure 7: Comparison Between the Existing Model and Our Proposed Model

8 DISCUSSIONS

- **Model Efficiency:** The stacked model was better than individual models since the strengths of SVM, KNN, and GNB were combined. KNN proved efficient in terms of pattern discerning. GNB worked well for DDoS detection. The ensemble model ensured that errors remained low and accuracy high.
- **Comparison with Traditional Methods:** This AI-driven model can detect emerging patterns of attacks, which is a clear advantage in the detection of novel threats, as conventional signature-based

DDoS-detecting approaches are not adopted here.

- **Issues:** The system could easily deteriorate due to slight variations in attack patterns. KNN suffered from handling high-dimensional data; however, this stacked model helped with this as well.
- **Possible Enhancements:** Potential future improvements may involve using deep learning to recognize better patterns or utilizing reinforcement learning to adjust thresholds of the model in real time, making it adaptive to new attacks.
- **Implication for Cybersecurity:** This system provides effective DDoS mitigation, quick detection and response, scalability, and adaptability, making it a valuable tool for organizations during attacks.

9 CONCLUSION

The proposed AI-driven system introduces a novel method for real-time detection and mitigation of cyberattacks, such as Distributed Denial of Service (DDoS) and ransomware. By leveraging a blend of ML(Machine learning) algorithms which includes Support Vector Machine (SVM), Naive Bayes, K-Nearest Neighbors (KNN), and a stacking ensemble approach, the system achieves superior accuracy and resilience in identifying and responding to threats. This model overcomes the limitations of conventional signature-based and rule-based systems by continuously adjusting to the ever-changing nature of cyberattacks. The system's capacity to analyze network traffic and monitor system behavior in real time establishes it as a dependable solution for enhancing cybersecurity. Its efficient computational design guarantees scalability, making it adaptable for implementation across various organizational settings.

9.1 Future Scope

Further advancements can be explored to enhance the system's capabilities:

- **Deep Learning Integration:** Incorporating deep learning techniques could improve the detection of complex and subtle attack patterns, enabling a more nuanced understanding of emerging threats.
- **Reinforcement Learning:** Adaptive models powered by reinforcement learning can dynamically adjust detection thresholds, ensuring optimal performance in real-time scenarios.
- **Automated Threat Mitigation:** Developing advanced defense mechanisms to automatically neutralize detected threats could further minimize response times and potential damage.
- **IoT Security:** Extending the system's functionality to secure Internet of Things (IoT) devices can address vulnerabilities in smart ecosystems.
- **Cloud-Based Scalability:** Implementing the solution as a cloud-based service would allow for broader accessibility and seamless updates to tackle newly discovered threats.

By addressing these future directions, the proposed system can evolve into a comprehensive cybersecurity solution, offering enhanced protection against a constantly changing threat landscape.

REFERENCES

Ahmed, A. and Malik, S. (2023). Cybersecurity applications of k-nearest neighbors algorithm. *Journal of Cybersecurity Research*, 8(3):210–225.

Aktar, S. and Show, A. Y. N. (2023). Deep learning detection for cyber threats. *Cybersecurity Insights*, 14(1):55–70.

Alshehri, A., Dyari, M., and Others (2023). Cyberattack detection using cicids2017. *Journal of Machine Learning and Applications*, 12(2):90–105.

Dyari, M. and Alshehri, A. (2021). Comprehensive ransomware datasets. *Journal of Computer Science*. Retrieved from `https://www.journals.elsevier.com/computer-science`.

Ferdous, J., Islam, R., and Others (2023). A systematic evaluation framework for ai-based cybersecurity. *Journal of Cybersecurity*, 12(1):45–60.

Johnson, M. (2024). *Cybersecurity in the Age of AI: Tools and Strategies*. TechSecure Press.

Patel, N. and Gupta, K. (2023). Knn and other algorithms for real-time cyber attack detection. *Journal of Applied Artificial Intelligence*, 19(7):198–213.

Pei, J., Chen, Y., and Ji, W. (2023). A ddos attack detection method using random forest. *Journal of Network Security*, 11(2):112–125.

Shan, S., Naqvi, A., and Alarcon, V. (2023). Deep learning for cyber defense. *IEEE Transactions on Cybernetics*, 53(3):556–570.

Smith, J. and Doe, A. (2020). Detecting ddos attacks using network traffic analysis. *Journal of Cybersecurity Research*, 15(4):250–270.

Smith, J. and Taylor, E. (2022). Real-time response mechanisms for ransomware attacks. *Cyber Defense Weekly*, 10(2):56–70.

Taylor, D. and Nguyen, T. (2023). Knn-based approaches for identifying emerging cyber threats. *Journal of Computer Networks and Security*, 15(4):305–319.

Yadav, R. and Singh, P. (2023). Automated detection of cyber threats using ai techniques. *Journal of Cybersecurity Research*, 12(4):34–45.

Zhang, L. and Wang, X. (2021). Classifying ransomware activities using deep learning. *Cybersecurity Advances*, 8(2):125–140.