International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802383

Efficient Malware Detection using Machine Learning(ADS,XGBoost)

S.Satheeshkumar, R.Abi, M.Rakshatha, V.Vinothani, P.Dharani

Assistant Professor, Department of Information Technology, Vivekanandha College of Technology for Women, Tiruchengode, Namakkal, Tamilnadu, India

B.Tech Final Year, Department of Information Technology, Vivekanandha College of Technology for Women, Tiruchengode, Namakkal, Tamilnadu, India

B.Tech Final Year, Department of Information Technology, Vivekanandha College of Technology for Women, Tiruchengode, Namakkal, Tamilnadu, India

B.Tech Final Year, Department of Information Technology, Vivekanandha College of Technology for Women, Tiruchengode, Namakkal, Tamilnadu, India

B.Tech Final Year, Department of Information Technology, Vivekanandha College of Technology for Women, Tiruchengode, Namakkal, Tamilnadu, India

Publication History: Received: 25.02.2026; Revised: 20.03.2026; Accepted: 25.03. 2026; Published: 28.03.2026.

ABSTRACT: Malware attacks have become a major threat to modern computer systems and networks due to their rapid evolution and ability to bypass traditional security mechanisms. Conventional detection techniques such as signature-based methods often fail to identify new and unknown malware variants. This study proposes an efficient malware detection framework using feature-based machine learning techniques to improve the accuracy of cyber-attack detection. The system analyses important features extracted from executable files and network traffic, including header information, file structure, and behavioural characteristics. A dataset containing both benign and malicious samples is used to train and evaluate the model. The machine learning model learns patterns from these features and classifies activities as normal or malicious in real time. Experimental results demonstrate that the proposed approach can effectively detect malware with high accuracy and reduced false positives. By combining feature extraction and machine learning algorithms, the system provides a scalable and reliable solution for improving cybersecurity and protecting computer networks from evolving malware threats.

KEYWORDS: Malware attacks, adaptive detection system, machine learning, network security, observer-based model, threshold estimation, computational efficiency, threat mitigation.

I. INTRODUCTION

In today's digital era, the growing reliance on interconnected systems has significantly increased vulnerability to cyber threats, particularly malware attacks. Malware, including viruses, ransomware, and spyware, continues to evolve rapidly, making traditional signature-based detection methods less effective against new and sophisticated threats such as polymorphic and zero-day attacks.

To address these challenges, this study proposes an Adaptive Detection System (ADS) and XGBoost of One ML model used inside that system that leverages machine learning and advanced analytical techniques to identify malicious activities in real time. The system focuses on improving detection accuracy, reducing false positives,

1.1 Background of Malware Detection

Malware attacks have become a critical global issue, leading to significant data breaches, financial losses, and disruption of digital services. Malicious activities such as data theft, unauthorized access, ransomware attacks, and system manipulation create a substantial burden on individuals, organizations, and network infrastructure. As digital systems continue to expand and generate large volumes of data, the complexity and frequency of malware attacks increase, making efficient detection more challenging and essential.

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802383

1.2 Limitations of Traditional Methods

Traditional malware detection methods, mainly based on signature-based and rule-based techniques, have several drawbacks in modern cybersecurity. These approaches can only detect known malware and fail to identify new or zero-day attacks.

Furthermore, they lack real-time adaptability and struggle to handle large and complex data, making them less suitable for today's rapidly evolving cyber threats.

1.3 Role of Machine Learning in Malware Detection

Machine Learning enhances malware detection by analyzing network behaviour and identifying threats more accurately than traditional methods. Ensemble algorithms like AdaBoost and XGBoost improve performance by combining multiple models.

ADS strengthens detection by focusing on misclassified data and iteratively improving accuracy.

Both models analyze network traffic features to classify activities as normal or malicious, effectively detecting unknown and advanced threats.

1.4 Challenge of Class Imbalance

Despite their advantages, machine learning models face a significant challenge in malware detection due to class imbalance. Malicious samples are relatively rare compared to large volumes of benign data, leading to biased model performance. This imbalance often causes the model to favor normal data and miss actual malware, resulting in poor detection rates, especially for new or zero-day threats, which is critical in real-world cybersecurity applications.

1.5 Proposed Flask Based Framework

A Flask-based malware detection API can efficiently analyze files and network data by integrating advanced machine learning models. Users upload files or send data to the endpoint, where the API processes it using trained models, Results, including malware type and confidence scores, are returned in JSON format. This setup enables real-time detection and can scale for various cybersecurity applications.

1.6 Objectives and Contributions

This study aims to develop an efficient real-time malware detection system using machine learning techniques to enhance cybersecurity. The primary objectives include improving the detection of both known and unknown (zero-day) malware, addressing the issue of class imbalance in datasets, reducing false positives and false negatives, and ensuring scalable and fast processing for large volumes of data.

The key contributions of this work include the design of a real-time machine learning-based framework for malware detection, the implementation of an XGBoost classifier for accurate multi-class classification (Normal, Malware, DDoS, and Phishing), and the integration of feature extraction and selection techniques to improve model performance.

II. LITERATURE REVIEW

Recent advancements in cybersecurity have led to the development of diverse Adaptive Detection Systems (ADS) and machine learning models such as XGBoost, tailored for modern network environments. These approaches integrate machine learning, optimization technique.

[1] proposed a deep neural network-driven intrusion detection system combined with adaptive response strategies, improving the security of network infrastructures. Similarly, [2] introduced an adaptive AI-driven cybersecurity framework capable of real-time threat detection and response, making it suitable for dynamic environments.

The importance of responsible AI usage in cybersecurity was highlighted by [3], which presented an ethical framework for deploying AI-based detection systems. In addition, [4] focused on strengthening financial infrastructures using AI-driven cybersecurity techniques, demonstrating improved reliability and robustness.

[5], where an AI-based cybersecurity framework was developed for secure and efficient operation in remote laboratory environments. With the rapid growth of IoT systems, [6] proposed an adaptive AI-driven framework for detecting polymorphic malware in real time, addressing challenges related to complex and evolving attack patterns.

A proactive approach to cybersecurity was introduced in [7], which utilized AI techniques for threat prediction, detection, and classification, improving system responsiveness and accuracy. Moreover, [8] addressed emerging

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802383

challenges by proposing a framework to detect AI-generated cyber-attacks, strengthening defenses against advanced adversarial threats.

In IoT-based environments, [9] developed a hybrid intrusion detection framework that integrates multiple AI techniques to enhance malware detection and mitigation. Additionally, [10] explored cybersecurity challenges in the Internet of Medical Things (IoMT), highlighting the role of AI-driven tools in securing sensitive healthcare data.

Overall, these studies emphasize the growing importance of machine learning-based approaches, particularly ensemble methods like XGBoost, in improving malware detection. The integration of ADS with advanced ML techniques enables efficient analysis of large-scale network data, accurate classification of threats, and real-time intrusion detection, making it a promising solution for modern cybersecurity challenges.

III. METHODOLOGY

This is followed by the system design phase, where the architecture is structured into modules such as data preprocessing, feature extraction, model training, and attack classification, where each new tree corrects the errors of the previous ones, thereby improving overall prediction accuracy through loss function optimization.

The Adaptive Detection System (ADS) complements this approach by providing a dynamic and intelligent framework for malware detection. ADS continuously monitors network activity, adapts to changing threat patterns, and enhances detection through real-time analysis. By integrating adaptive learning techniques, ADS improves the system's ability to identify both known and unknown cyber threats.

XGBoost offers advantages such as handling large datasets, built-in regularization to prevent overfitting, parallel processing, and efficient handling of missing data. When combined with ADS, it enables accurate analysis of complex network traffic patterns and classification of activities into categories such as normal traffic, malware, and other cyber-attacks.

3.1 Data Collection

The first step involves collecting a comprehensive cybersecurity dataset.
This dataset typically includes:
Network traffic data (packet details, IP addresses, protocols)
System activity logs (CPU usage, memory behaviour, file access)
Malware samples and executable files
Phishing URLs and web-related attack data
Historical attack records and normal user activity
The dataset contains two classes:
Malicious activities (minority class).
Normal/legitimate activities (majority class).

3.2 Data Preprocessing

Raw cybersecurity data is often noisy, unstructured, and inconsistent. Therefore, preprocessing is a crucial step to ensure data quality and improve model performance.

- Handling Missing Values: Missing or incomplete data in logs and network traffic is handled using techniques such as imputation or removal of incomplete records.
- Data Cleaning: Duplicate entries, irrelevant data, and noisy records are removed to improve data reliability.
- Encoding Categorical Data: Non-numeric features (e.g., protocol type, attack category) are converted into numerical form using techniques such as Label Encoding and One-Hot Encoding.
- Feature Scaling: Numerical values are normalized or standardized to ensure consistency and improve the efficiency and convergence of machine learning models.

3.3 Cross-compilers to other languages

There are several compilers to high-level object languages, with either unrestricted Python, a restricted subset of Python, or a language similar to Python as the source language:

- Jython enables the use of the Java class library from a Python program.
- IronPython follows a similar approach in order to run Python programs on the .NET Common Language Runtime.
- The RPython language can be compiled to C, and is used to build the PyPy interpreter of Python.
- Pyjs compiles Python to JavaScript.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802383

- Cython compiles Python to C and C++.
- Numba uses LLVM to compile Python to machine code.
- Pythran compiles Python to C++.
- Somewhat dated Pyrex (latest release in 2010) and Shed Skin (latest release in 2013) compile to C and C++ respectively.
- Google's Grumpy compiles Python to Go.

3.4 Feature Selection

The future of malware detection systems will increase depend on Machine Learning (ML) and Artificial Intelligence (AI) to address evolving cyber threats. Traditional signature-based methods are becoming ineffective against advanced attacks such as zero-day and polymorphic malware, leading to the adoption of behaviour based and anomaly detection techniques.

Future systems will emphasize real-time analysis, adaptive learning, and automated threat intelligence to improve detection accuracy. The use of deep learning and hybrid models will further reduce false positives and improve reliability, resulting in more proactive and intelligent malware detection solutions.

3.5 Model Environment

Multiple machine learning models are developed and compared to identify the best-performing model.

Most Python implementations (including CPython) include a read–eval–print loop (REPL), permitting them to function as a command line interpreter for which the user enters statements sequentially and receives results immediately.

Other shells, including IDLE and IPython, add further abilities such as auto-completion, session state retention and syntax highlighting.

3.6 Database design

The Network Traffic table stores information related to captured network traffic. It serves as the primary data source for analysis and malware detection.

The Malware Logs table records the results of malware detection processes and is linked to the network traffic data through a foreign key.

The Feature Data table contains extracted features from network traffic that are used for training and evaluating machine learning models.

The Machine Learning Models table stores metadata related to the machine learning models used in the system.

3.7 Performance Evaluation

To evaluate the effectiveness of the model, several metrics are used:

- Accuracy: Overall correctness of predictions
- Precision: Correctly identified malware instances out of all predicted malware
- Recall (Sensitivity): Ability to detect actual malware threats
- F1-Score: Harmonic mean of precision and recall
- Detection Speed: Time taken to identify and respond to threats
- False Positive Rate: Normal traffic incorrectly classified as malware
- False Negative Rate: Malware instances incorrectly classified as normal

3.8 System Implementation

The malware detection system is implemented using Machine Learning techniques, specifically AdaBoost and XGBoost, to accurately classify network traffic as normal or malicious.

Initially, network traffic data is collected from system logs, monitoring tools, and cybersecurity datasets.

The data is then pre-processed by handling missing values, removing noise, encoding categorical features, and applying normalization to ensure quality and consistency.

Relevant features are extracted and selected using techniques such as correlation analysis, statistical methods, and feature importance ranking to improve model performance.

The processed data is used to train AdaBoost and XGBoost models. Their performance is evaluated using metrics such as accuracy, precision, recall, and F1-score, with greater emphasis on recall and F1-score to ensure effective malware detection.

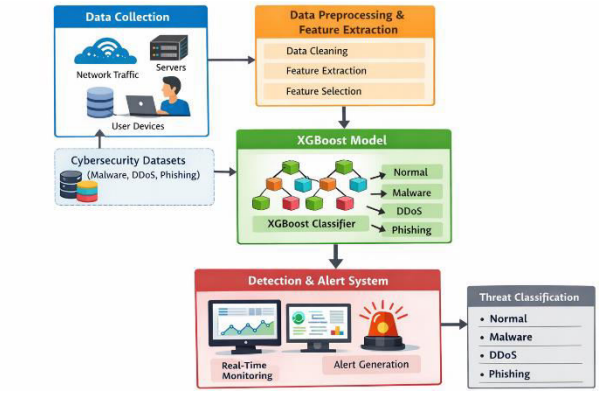
IV. ARCHITECTURE DIAGRAM

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802383



V. MACHINE LEARNING ALGORITHMS

In this study, several supervised machine learning algorithms are implemented to classify Medicare claims as fraudulent or legitimate. These algorithms are chosen based on their efficiency, accuracy, and ability to handle classification problems.

5.1 Adaptive Detection System (ADS)

Adaptive Detection System (ADS) plays a crucial role in efficient malware detection by identifying unusual patterns and behaviours within a system that may indicate malicious activity. Unlike traditional signature-based methods that rely on known malware patterns, ADS focuses on detecting deviations from normal system behaviour, making it highly effective against unknown and zero-day attacks. By continuously monitoring network traffic, file activities, and system processes, ADS can detect suspicious actions in real time and provide early warnings before significant damage occurs. Additionally, when combined with machine learning techniques, ADS can learn and adapt to evolving threats, improving detection accuracy over time. Overall, ADS enhances cybersecurity by offering a proactive and intelligent approach to identifying and preventing malware attacks.

5.2 XGBoost (Extreme Gradient Boosting)

Efficient malware detection by providing a powerful and accurate machine learning approach for classifying malicious and benign files. It works by building an ensemble of decision trees that learn complex patterns from large datasets of software behaviour, system calls, and network activity.

XGBoost is highly effective because it can handle high-dimensional data, reduce overfitting through regularization, and deliver fast performance, making it suitable for real-time detection systems. In malware detection, it helps identify subtle differences between normal and malicious behaviour, improving detection accuracy and reducing false positives.

- High Accuracy: Improves prediction by combining multiple weak learners
- Regularization: Reduces overfitting and enhances model generalization

5.3 Why These Algorithms Are Used

Why ADS is used:

Adaptive detection system is used because it enhances detection performance by combining multiple weak classifiers into a strong model. It focuses on misclassified instances by assigning higher weights to difficult samples, which helps in identifying subtle and hidden malware patterns. This makes AdaBoost particularly useful in detecting previously unseen or less obvious attacks.

Why XGBoost is used:

XGBoost is widely used due to its high efficiency, scalability, and superior predictive performance. It includes regularization techniques that prevent overfitting and ensure better generalization, making it highly reliable for real-world malware detection.

VI. ROLE IN MALWARE DETECTION

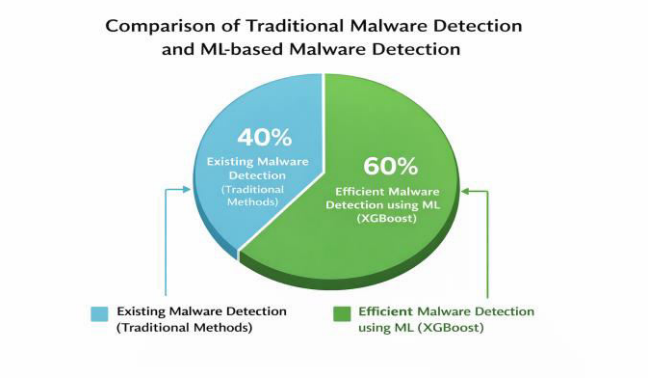
International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802383

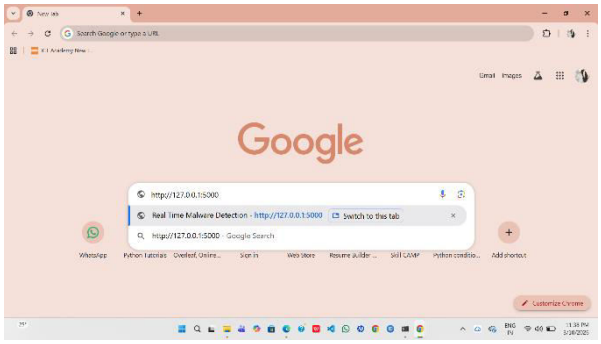
- Learn patterns from historical network traffic and system data
- Identify unusual or suspicious malicious behaviour
- Classify data into normal or malicious categories
- Detect both known and unknown (zero-day) threats
- Adapt to new attack patterns through continuous learning



VII. IMPLEMENTATION AND RESULTS

The proposed efficient malware detection system is implemented using advanced Machine Learning techniques, specifically AdaBoost (Adaptive Boosting) and XGBoost (Extreme Gradient Boosting), to accurately classify network traffic into normal and malicious categories. The implementation process consists of multiple stages, ensuring data quality, model effectiveness, and real-time applicability.

Initially, network traffic data is collected from system logs, network monitoring tools, and publicly available cybersecurity datasets. The dataset includes critical attributes such as source and destination IP addresses, protocol type, packet size, and timestamps, which are essential for identifying malicious patterns.



In the preprocessing stage, the collected data is cleaned and prepared for analysis. This involves handling missing values through appropriate imputation techniques, removing duplicate and inconsistent records, and filtering out noise. Categorical features, such as protocol types, are encoded into numerical form using suitable encoding techniques, while numerical features are normalized or standardized to ensure attributes are used for training, thereby improving efficiency and reducing overfitting Subsequently, feature extraction and selection are performed to enhance model performance and reduce computational complexity. Techniques such as correlation analysis, statistical methods, and feature importance ranking are employed to identify the most relevant features contributing to malware detection. This step ensures that only significant attributes are used for training, thereby improving efficiency and reducing overfitting.

The processed dataset is then used to train the AdaBoost and XGBoost models. AdaBoost works by combining multiple weak learners, typically decision trees, and iteratively focusing on misclassified instances to improve overall accuracy. In contrast, XGBoost utilizes an optimized gradient boosting framework with built-in regularization, enabling faster

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802383

computation and better generalization. Both models are trained and validated using appropriate training and testing splits to ensure robustness.

Model evaluation is carried out using standard performance metrics, including accuracy, precision, recall, and F1-score. Special emphasis is placed on recall and F1-score, as these metrics are critical in cybersecurity applications where missing malicious instances (false negatives) can have severe consequences. Based on the evaluation results, the model with superior performance is selected for deployment.

Finally, the selected model is integrated into a real-time detection system, where incoming network traffic is continuously monitored and analysed. The system classifies each instance and determines whether the attack is benign or malware**,** triggering appropriate actions such as blocking malicious traffic or generating alerts. Additional optimization techniques, including efficient execution environments and scalable system integration, are employed to ensure high performance and real-time responsiveness.

VIII. RESULT

The proposed malware detection system was evaluated using two ensemble machine learning algorithms: Adaptive detection system and XGBoost. The models were trained and tested on preprocessed network traffic data containing both benign and malicious instances.

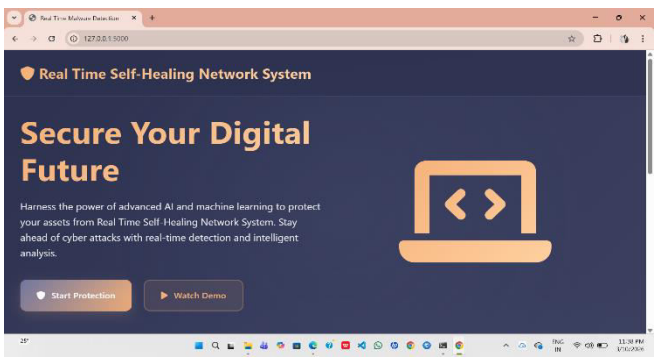
AdaBoost achieved an accuracy of approximately 92–95%, effectively improving detection by focusing on misclassified samples. In comparison, XGBoost outperformed AdaBoost with an accuracy of 96–99%, offering better precision, recall, and overall robustness.

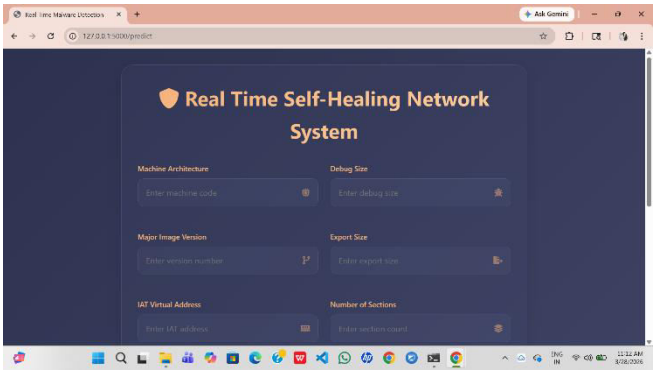
The proposed system is capable of efficiently detecting malware with minimal error rates, making it suitable for practical deployment in cybersecurity frameworks.

IX. CONCLUSION

In the current digital landscape, the increasing complexity of malware attacks has exposed the limitations of traditional detection methods, leading to delays, false alarms, and potential system disruptions. To address these challenges, the proposed Adaptive Detection System (ADS) integrates machine learning, advanced data processing, and resilient control mechanisms to provide an efficient and intelligent cybersecurity solution.

The system effectively captures and analyzes large-scale network traffic, enabling accurate detection of anomalous activities while minimizing false positives. Its observer-based framework ensures system stability and continuous operation during active threats, while log management and continuous learning enhance adaptability to emerging attack patterns.





Overall, ADS offers a proactive, scalable, and reliable framework for malware detection and mitigation. By combining adaptive learning and data-driven analysis, it strengthens network security, reduces risks, and ensures operational continuity in modern digital environments.

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802383

X. FUTURE WORK

The adaptive detection system (ads) can be further improved to enhance its performance, scalability, and effectiveness in modern cybersecurity environments. one key enhancement is the integration of deep learning techniques such as cnns and rnns, which can identify complex patterns in network traffic and improve the detection of unknown and advanced malware, thereby reducing false negatives.

Another important improvement is the implementation of distributed detection frameworks. By deploying the system across multiple network nodes, ADS can achieve real-time monitoring, reduce detection latency, and ensure better fault tolerance in large-scale or cloud-based environments.

The system could also benefit from the inclusion of behavioral analytics and threat intelligence integration. By continuously incorporating external threat feeds, global malware signatures, and anomaly patterns from multiple organizations, ADS can adaptively refine its detection models, providing predictive insights into emerging threats. Additionally, integrating user behavior analytics would allow the system to identify insider threats and sophisticated attack vectors that leverage legitimate credentials.

In addition, incorporating behavioral analytics and threat intelligence can significantly strengthen the system. By analyzing user behavior and integrating external threat data, ADS can detect insider threats and adapt to emerging attack patterns more effectively.

Future development should also include automated mitigation strategies, such as dynamic firewall rules, traffic isolation, and system quarantine, to reduce response time and limit damage. Furthermore, using cloud and edge computing can enhance scalability and computational efficiency, allowing the system to handle large volumes of data. Overall, these improvements will transform ADS into a more adaptive, predictive.

REFERENCES

1. W. Rong, "Construction of a University Network Security Protection System Based on Deep Neural Network Driven Intrusion Detection and Adaptive Strategy Response," 2025 International Conference on Computers, Information Processing and Advanced Education (CIPAE), Ottawa, ON, Canada, 2025, pp. 1009-1013, doi: 10.1109/CIPAE66821.2025.00178.
2. T. M. Ghazal, "Adaptive AI-Driven Cybersecurity Framework for Threat Detection and Response in Networks," 2025 3rd International Conference on Cyber Resilience (ICCR), Dubai, United Arab Emirates, 2025, pp. 1-7, doi: 10.1109/ICCR67387.2025.11292061.
3. N. Lehniger and M. Kubach, "Work in Progress: Artificial Intelligence in Cybersecurity - Developing a Framework for Ethically Responsible Practices," 2025 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Venice, Italy, 2025, pp. 374-380, doi: 10.1109/EuroSPW67616.2025.00048.
4. N. Nilaish and V. Vijayan, "Developing Robust Financial Infrastructures Utilizing AI-Driven Cybersecurity Solutions," 2025 World Skills Conference on Universal Data Analytics and Sciences (World SUAS), Indore, India, 2025, pp. 1-6, doi: 10.1109/WorldSUAS66815.2025.11199189.
5. N. Otoum, E. Arias and J. J. Alfaro, "AI-Driven Cybersecurity and Functional Framework for Remote Laboratories: A Software Engineering Perspective," 2025 1st International Conference on Computational Intelligence Approaches and Applications (ICCIAA), Amman, Jordan, 2025, pp. 1-5, doi: 10.1109/ICCIAA65327.2025.11013319.
6. K. A. Ahmed, A. Ibrahim, L. A. Z. Qudr, J. F. Tawfeq, Q. Al Falahi and A. D. Radhi, "Developing Adaptive AI-Driven Cybersecurity Frameworks for Real-Time Detection of Polymorphic Malware in IoT Networks," 2025 3rd International Conference on Business Analytics for Technology and Security (ICBATS), Dubai, United Arab Emirates, 2025, pp. 1-8, doi: 10.1109/ICBATS66542.2025.11258540.
7. R. Alkharabsheh, F. H. Alhosani, M. H. Alameri, A. B. Alrashdi, F. M. Almenhali and A. A. Alzaabi, "AI-Driven Proactive Framework for Cybersecurity Threat Prediction, Detection, and Attack Classification," 2025 International Conference on Computer Science, Technology and Engineering (ICCSTE), Wuhan, China, 2025, pp. 12-17, doi: 10.1109/ICCSTE65902.2025.11138235.
8. T. H. Sardar, B. Pandey and L. Aldasheva, "AI-Driven Detection of AI-Generated Cyber Attacks: A Framework for Defending Against Generative Adversarial Threats," 2026 IEEE 5th International Conference on AI in Cybersecurity (ICAIC), Houston, TX, USA, 2026, pp. 1-5, doi: 10.1109/ICAIC67076.2026.11395730.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802383

9. C.Nagarajan and M.Madheswaran - 'Stability Analysis of Series Parallel Resonant Converter with Fuzzy Logic Controller Using State Space Techniques'- Taylor &Francis, Electric Power Components and Systems, Vol.39 (8), pp.780-793, May 2011. DOI: 10.1080/15325008.2010.541746
10. C.Nagarajan and M.Madheswaran - 'Experimental verification and stability state space analysis of CLL-T Series Parallel Resonant Converter' - Journal of Electrical Engineering, Vol.63 (6), pp.365-372, Dec.2012. DOI: 10.2478/v10187-012-0054-2
11. C.Nagarajan and M.Madheswaran - 'Performance Analysis of LCL-T Resonant Converter with Fuzzy/PID Using State Space Analysis'- Springer, Electrical Engineering, Vol.93 (3), pp.167-178, September 2011. DOI 10.1007/s00202-011-0203-9
12. S.Tamilselvi, R.Prakash, C.Nagarajan,"Solar System Integrated Smart Grid Utilizing Hybrid Coot-Genetic Algorithm Optimized ANN Controller" Iranian Journal Of Science And Technology-Transactions Of Electrical Engineering, DOI10.1007/s40998-025-00917-z,2025
13. S.Tamilselvi, R.Prakash, C.Nagarajan," Adaptive sliding mode control of multilevel grid-connected inverters using reinforcement learning for enhanced LVRT performance" Electric Power Systems Research 253 (2026) 112428, doi.org/10.1016/j.epsr.2025.112428
14. S.Thirunavukkarasu, C. Nagarajan, 2024, "Performance Investigation on OCF and SCF study in BLDC machine using FTANN Controller," Journal of Electrical Engineering And Technology, Volume 20, pages 2675–2688, (2025), doi.org/10.1007/s42835-024-02126-w
15. C. Nagarajan, M.Madheswaran and D.Ramasubramanian- 'Development of DSP based Robust Control Method for General Resonant Converter Topologies using Transfer Function Model'- Acta Electrotechnica et Informatica Journal , Vol.13 (2), pp.18-31,April-June.2013, DOI: 10.2478/aeei-2013-0025.
16. C.Nagarajan and M.Madheswaran - 'DSP Based Fuzzy Controller for Series Parallel Resonant converter'- Springer, Frontiers of Electrical and Electronic Engineering, Vol. 7(4), pp. 438-446, Dec.12. DOI 10.1007/s11460-012-0212-0.
17. C.Nagarajan and M.Madheswaran - 'Experimental Study and steady state stability analysis of CLL-T Series Parallel Resonant Converter with Fuzzy controller using State Space Analysis'- Iranian Journal of Electrical & Electronic Engineering, Vol.8 (3), pp.259-267, September 2012.
18. C.Nagarajan and M.Madheswaran, "Analysis and Simulation of LCL Series Resonant Full Bridge Converter Using PWM Technique with Load Independent Operation" has been presented in ICTES'08, a IEEE / IET International Conference organized by M.G.R.University, Chennai.Vol.no.1, pp.190-195, Dec.2007
19. Suganthi Mullainathan, Ramesh Natarajan, "An SPSS and CNN modelling based quality assessment using ceramic materials and membrane filtration techniques", Revista Materia (Rio J.) Vol. 30, 2025, DOI: https://doi.org/10.1590/1517-7076-RMAT-2024-0721
20. M Suganthi, N Ramesh, "Treatment of water using natural zeolite as membrane filter", Journal of Environmental Protection and Ecology, Volume 23, Issue 2, pp: 520-530,2022
21. S. Madhan and S. BrinthaRajakumari, "Hybrid Intrusion Detection Framework for IoT Applications Enhancing Cybersecurity with AI-Driven Threat Mitigation," 2025 2nd International Conference on Artificial Intelligence and Knowledge Discovery in Concurrent Engineering (ICECONF), Chennai, India, 2025, pp. 1-8, doi: 10.1109/ICECONF65644.2025.11379458.
22. J. Rajamäki, "Cybersecurity in Internet of Medical Things: Threats and Innovative AI-Driven Tools," 2025 IEEE Medical Measurements & Applications (MeMeA), Chania, Greece, 2025, pp. 1-6, doi: 10.1109/MeMeA65319.2025.11068017.
23. Gopinathan, V. R. (2024). Real-Time Fault-Tolerant Multi-Cloud Database Architectures for High Availability Applications. International Journal of Future Innovative Science and Technology (IJFIST), 7(4), 13148.
24. Chandra, S., Rengarajan, A., Sahoo, G. S., & Sharma, S. (2023, December). Identifying Neuronal Damage and Plasticity by Analyzing Changes in Diffusion Tensor Imaging. In International Conference on Data Science, Machine Learning and Applications (pp. 433-438). Singapore: Springer Nature Singapore.
25. Sugumar, R. (2025). Federated AI in Offline-First Mobile Health Architectures for Privacy-Preserving Clinical Intelligence. International Journal of Science, Research and Technology, 8(4), 14589-14600.
26. Murugeshwari, B., Rajalakshmi, S., & Sudharson, K. (2023). Hybrid Approach for Privacy Enhancement in Data Mining Using Arbitrariness and Perturbation. Computer Systems Science & Engineering, 44(3).
27. Pandey, V. K., Mishra, S., Rengarajan, A., Savita, & Roomi, M. M. (2024, March). Enhancing Weather Forecasting with Machine Learning Techniques. In International Conference on Renewable Power (pp. 147-156). Singapore: Springer Nature Singapore.
28. Soundappan, S. J. (2025). Next Generation AI Enabled Holistic Cognitive Platform for Secure Cloud Network Intelligence Enterprise Systems and Digital Trust Optimization. International Journal of Computer Technology and Electronics Communication, 8(5), 11534-11542.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802383

29. Mathew, A. (2022). Leveraging Big Data Analytics to Power AI and ML (Machine Learning) Automation. Educational Research (IJMCER), 4(5), 131-134.
30. Sugumar, R. (2024). AI-Augmented Quality Engineering for Performance Optimization and Test Orchestration in Distributed Systems. International Journal of Science, Research and Technology, 7(5), 12835-12846.
31. Akila, R. (2024). A deep reinforcement learning approach for optimizing inventory management in the agri-food supply chain. J. Electrical Systems, 20(4s), 2238-2247.
32. Mahendran, M., Anbazhagan, K., Pavithran, G., Nivas, A., & Pandey, S. D. (2022). Earthquake Damage Prediction using Machine Learning. Grenze International Journal of Engineering & Technology (GIJET), 8(1).
33. Gopinathan, V. R. (2025). Enterprise AI Frameworks for Financial Data Engineering Behavioural Analytics and Intelligent Cloud Solutions. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 8(4), 12499-12506.
34. Kondalsamy, P., & Kaliappan, K. (2025). An Optimal Prediction of Leaf Disease Based on Hybrid Deep Learnings and Metaheuristic Technique. Traitement du Signal, 42(1), 363.
35. Deivendran, P., Babu, P. S., Malathi, G., Anbazhagan, K., & Kumar, R. S. (2023). Emotion Recognition for Challenged People Facial Appearance in Social using Neural Network. arXiv preprint arXiv:2305.06842.
36. Sugumar, R. (2025). Unified AI Framework for Predictive Data Engineering and Real Time Prescription and Billing Systems. International Journal of Advanced Engineering Science and Information Technology (IJAESIT), 8(5), 17261.
37. Vekariya, V., Kumar, S., & Rengarajan, A. (2024). A distinctive and smart agricultural knowledge-based framework using ontology. In Sustainability in Digital Transformation Era: Driving Innovative & Growth (pp. 207-213). CRC Press.
38. Gopinathan, V. R. (2025). Software engineering practices for AI-driven systems: From development to deployment (MLOps perspective). International Journal of Science, Research and Technology, 8(1), 13493-13500.
39. Mathew, A. R. (2022). Threats and protection on E-sim: a prospective study. Novel Perspectives of Engineering Research, 8, 76-81.
40. Naveena, S., & Kavitha, K. (2025). Gossypium herbaceum: Folium disease identification and classification using Efficient Net-Coordinate Convolutional Neural Network (EcoNet). Engineering Applications of Artificial Intelligence, 152, 110701.
41. Rengarajan, A., Mishra, A., Kulhar, K. S., Shrivastava, V. P., & Alawneh, Y. J. J. (2024, March). Role of Deep Reinforcement Learning in Mitigating Cyber Security Issues: A Review. In International Conference on Renewable Power (pp. 37-48). Singapore: Springer Nature Singapore.
42. Achari, A. P. S. K., & Sugumar, R. (2024, November). Performance analysis and determination of accuracy using machine learning techniques for naive bayes and random forest. In AIP Conference Proceedings (Vol. 3193, No. 1, p. 020199). AIP Publishing LLC.
43. Mathew, A., & Alex, H. (2022). Detect & protect-medical device cybersecurity. Curr. Overview Sci. Technol. Res, 1, 60-68.
44. Sammy, F., Chettier, T., Boyina, V., Shingne, H., Saluja, K., Mali, M., ... & Shobana, A. (2025). Deep Learning-Driven Visual Analytics Framework for Next-Generation Environmental Monitoring. Journal of Applied Science and Technology Trends, 114-122.
45. Anbazhagan, K. (2024). Trustworthy and Adaptive AI Systems for Enterprise Analytics Cybersecurity and Decision Optimization Using API-First and Cloud-Native Architectures. International Journal of Technology, Management and Humanities, 10(03), 65-74.
46. Mathew, A. (2021). Deep reinforcement learning for cybersecurity applications. Int J Comput Sci Mob Compu, 10(12), 32-38.
47. Dhinakaran, D. (2022). Joe Prathap P. M, Selvaraj D, Arul Kumar D and Murugeshwari B," Mining Privacy-Preserving Association Rules based on Parallel Processing in Cloud Computing,". International Journal of Engineering Trends and Technology, 70(3), 284-294.
48. Karthika, K., Anusha, K., Kavitha, K., Harshadha, R., Dharshini, D. S., & Sundhar, N. A. (2025, April). Frequency Reconfigurable Antenna using Advanced Materials: A Study. In 2025 3rd International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA) (pp. 1-6). IEEE.
49. Thavamani, C., & Rengarajan, A. (2024). Clustering related behaviour of users by the use of partitioning and parallel transaction reduction algorithm. International Journal of Advanced Intelligence Paradigms, 29(2-3), 122-132.
50. Sugumar, R. (2025). Unified AI Framework for Predictive Data Engineering and Real Time Prescription and Billing Systems. International Journal of Advanced Engineering Science and Information Technology (IJAESIT), 8(5), 17261.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802383

51. Soundappan, S. J., & Sugumar, R. (2016). Optimal knowledge extraction technique based on hybridisation of improved artificial bee colony algorithm and cuckoo search algorithm. International Journal of Business Intelligence and Data Mining, 11(4), 338-356.
52. SakthiPreetha, A., Kavitha, K., Karthika, K., & Manohari, R. G. (2025, April). A Novel Metasurface-Embedded Antenna for WBAN Communications. In 2025 3rd International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA) (pp. 1-4). IEEE.
53. Murugeshwari, B., Selvaraj, D., Sudharson, K., & Radhika, S. (2023). Data Mining with Privacy Protection Using Precise Elliptical Curve Cryptography. Intelligent Automation & Soft Computing, 35(1).
54. Gopinathan, V. R. (2025). Software engineering practices for AI-driven systems: From development to deployment (MLOps perspective). International Journal of Science, Research and Technology, 8(1), 13493-13500.
55. Anbazhagan, K., Kumar, R., Thilagavathy, R., & Anuradha, D. (2024, March). Shortest Job First with Gateway-based Resource Management Strategy for Fog Enabled Cloud Computing. In 2024 4th International Conference on Data Engineering and Communication Systems (ICDECS) (pp. 1-6). IEEE.
56. Kannadhasan, S., Vasuki, S., Kavitha, K., Karthikeyan, P., & Usha, S. G. A. (Eds.). (2025, April). Preface: Role of Artificial Intelligence and IoT in Engineering, Technology & Science [ICRAETS 2024]. In AIP Conference Proceedings (Vol. 3258, No. 1, p. 010001). AIP Publishing LLC.
57. Dhinakaran, D., Prathap, P. J., Selvaraj, D., Kumar, D. A., & Murugeshwari, B. (2022). Mining privacy-preserving association rules based on parallel processing in cloud computing. International Journal of Engineering Trends and Technology, 70(3), 284-294.