WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

ML-Based Intrusion Detection System (IDS)

Nethra H L[1], Ashutosh Ranjan[2], Aman Chaudhary[3], Ayush Kumar[4], Aman Kumar[5]

[1]Assistant Professor, Computer Science and Engineering, Dayananda Sagar Academy of Technology & Management, Bengaluru, India
nethra-cse@dsatm.edu.in
[2]Student, 4th Year, B.E, Computer Science and Engineering, Dayananda Sagar Academy of Technology and Management, Bengaluru, India
ashutoshr04122001@gmail.com
[3]Student, 4th Year, B.E, Computer Science and Engineering Dayananda Sagar Academy of Technology and Management, Bengaluru, India
ac123kvvk@gmail.com
[4]Student, 4th Year, B.E Computer Science and Engineering Dayananda Sagar Academy of Technology and Management Bengaluru, India
28ayushnagar@gmail.com
[5]Student, 4th Year, B.E Computer Science and Engineering Dayananda Sagar Academy of Technology and Management Bengaluru, India
amansingh4467@gmail.com

ABSTRACT-

As cyber threats continue to grow in complexity and frequency, ensuring data protection and integrity has become more critical than ever. Intrusion Detection Systems (IDS) play a vital role in identifying and preventing abnormal activities across networks. This work focuses on how Artificial Intelligence (AI) can be effectively integrated into IDS to enhance their performance in real-world environments.

Unlike traditional IDS solutions, this approach leverages advanced AI techniques such as machine learning and deep learning to improve the detection of unusual or malicious traffic. These intelligent systems are designed to minimize false alarms that can otherwise overwhelm security teams and network operations. The paper explores how AI-enhanced IDS can help maintain data authenticity and strengthen security across diverse and dynamic network environments.

In addition, the study looks at practical challenges such as scalability, adaptability to evolving threats, and operation under resource constraints. Experimental results demonstrate that AI-powered IDS not only improve detection of sophisticated attacks but also maintain system stability and performance. The paper concludes by outlining future research directions, including the development of hybrid models and more advanced real-time threat response techniques to further strengthen cybersecurity frameworks.

Keywords— Artificial Intelligence, Intrusion Detection System, Cybersecurity, Real-World Deployment, Data integrity, Data security.

I. INTRODUCTION

The rise of digital platforms has ushered in an age of cyberattacks, which have become increasingly sophisticated and can cause great harm to individuals and organizations. Traditional rule-based security systems are shown to be static and unable to act as defense mechanization against zero-day attacks. Machine Learning provides a means to analyze and monitor massive network data in real time and identify malicious activities in adaptive and dynamic ways. The proposed study is to assess the role of ML, in designing suitable IDS that functional adapts to variations in attack patterns by learning from historical and real-time behavior of network activity.

Reliability and protection of data require, and sometimes prefer, a scientist constructed under the assumption of belief in their voice. One must therefore come to terms with the rightful demarcation of the limits of that authority. One key test is whether the validity or invalidity of a piece of data over its life cycle is defined by its reliability and trustworthiness—a tenet central to the measure of data integrity. Methods like data integrity checks, validations, and cryptographic verifications are some of the typical procedures used to maintain these tenets.

Applying ML is no longer a science fiction for security frameworks, it has become a necessity. As businesses increasingly depend on digitized infrastructures, the attack surface of cyberthreats also expands. ML-based IDS can inspect large volumes of network traffic and identify outliers–those that deviate from normal behavior—which are often the warning signs of an attack. They continue learning, and refining, their concept of normal vs. anomalous behavior, which offers a far superior benefit to outmoded methods of detection based on static rules or signatures.

While AI and machine learning hold incredible promise for enhancing data security, many business leaders are beginning to question just how much they can really trust these technologies. According to a global survey by KPMG involving 2,190 senior executives, there is a noticeable lack of confidence when it comes to using AI in data-driven decision-making. Only 35% of employees felt that data gathering and analytics were working well within their organizations. Even more striking, 92% expressed concerns about the reputational risks associated with poor data handling and analytics practices. These numbers make it clear: AI and ML should not be adopted blindly. Instead, their use calls for transparency, strong ethical standards, and a responsible approach to governance.

Interestingly, just over a third of executives surveyed felt confident in how their organizations were using data analytics. This isn't just a reflection of technological gaps—it points to a broader divide in leadership mindset and trust when it comes to AI adoption. The skepticism voiced wasn't limited to the performance of the technology itself; it extended to concerns about the long-term reliability of automated systems and the risks of overreliance on machine-based decisions. Nearly two-thirds of respondents expressed doubts about the consistency and dependability of their data and analytics infrastructure.

As we move deeper into an era shaped by intelligent systems, it becomes more important than ever to understand the foundations of AI and ML—not just the algorithms, but the full context in which data is collected, processed, and used. Before diving into more advanced applications like behavioral analytics, real-time threat detection, or predictive modeling, it's essential to build a solid understanding of how these systems work. That includes not only technical principles like statistical learning, but also the data lifecycle, usage environments, and the ethical responsibilities involved in algorithmic decision-making.

This paper contributes to that understanding by exploring how machine learning techniques can be applied to intrusion detection and data security. Through practical implementation and comparative analysis, we aim to highlight the strengths, limitations, and real-world implications of using ML to protect digital environments—and, ultimately, to foster greater trust in data-driven systems

II. SYSTEM ARCHITECTURE

A. Overall Architecture

The experiments for training and evaluating the proposed machine learning-based Intrusion Detection System (IDS) were conducted in a high-performance computing environment to ensure real-time analysis and scalable deployment.

1. Hardware Configuration

- **Small Network:** 1 Gbps speed; supports up to 100 hosts
- **Medium Network:** 1–10 Gbps; supports 100–1,000 hosts
- **Large Network:** 10+ Gbps; supports thousands to millions of hosts

Recommended Specs:

- **Processor:** Quad-core minimum; 8–16 cores (Intel Xeon/Ryzen) recommended for traffic analysis
- **RAM:** Minimum 8 GB; 16–64 GB for optimized ML processing
- **Storage:** 500 GB SSD minimum; 1 TB+ preferred for log and model data
- **Power/Cooling:** UPS support and dedicated cooling for server-grade workloads.

2. Software Environment

- **Operating Systems**: Ubuntu server, Debian, or Windows Server (if needed)
- **Programming Language:** Python 3.7+ for ML model and data processing. And Node.js for high performance backend services.
- **AI/ML Libraries:** TensorFlow, PyTorch, Keras, Scikit-learn
- **Traffic Analysis Tools:** Wireshark, Tcpdump, ntopng, nfdump
- **Databases:** PostgreSQL, MySQL for structured data; Elasticsearch for fast querying of logs
- **Development Tools:** Jupyter Notebook, VS Code, PyCharm
- **Deployment Tools:** Docker/Kubernetes for containerization; Jenkins, Ansible for CI/CD and configuration management

This configuration ensured robust performance during model training, testing, traffic capture, and system orchestration.

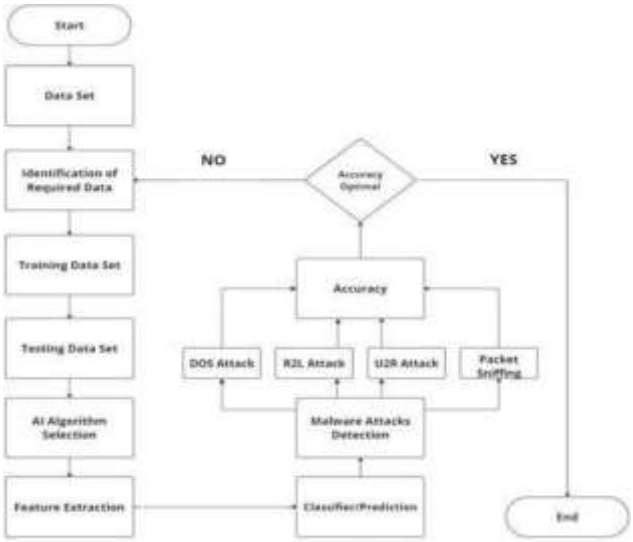

Figure 1. Proposed System for Malware Attacks detection

B. Workflow

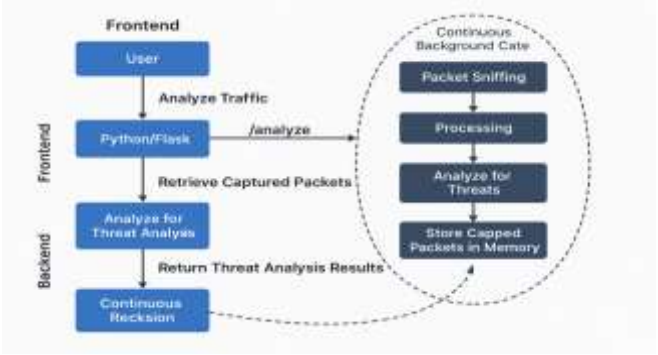

Figure 2. workflow of the system ensures ensures **real-time traffic monitoring**, **low-latency analysis**, and **responsive user interaction**, all while managing memory efficiently.

This diagram shows how an AI-powered Network Intrusion Detection System (NIDS) works behind the scenes to monitor and analyze network activity in real time. The process kicks off when a user clicks the "Analyze Traffic" button on the dashboard. That action sends a request to the backend (built with Python and Flask), which pulls in recently captured packets and runs them through a threat detection engine. The results—highlighting any suspicious or malicious activity—are then sent back to the user interface for easy viewing.

Meanwhile, the system is constantly running in the background, quietly capturing new packets as they flow through the network. It processes each one, pulls out the important details, and checks for signs of threats using detection logic or machine learning models. To stay efficient and avoid memory overload, it only keeps a limited number of recent packets in memory. This setup ensures the system can respond quickly, deliver insights in real time, and stay stable—even as it keeps an eye on everything happening across the network.

(And just a quick note: there's a typo in the diagram “Reckssion” should probably be “Detection” or “Recognition.”)

III. METHODOLOGY

We approached this project by blending solid technical foundations with practical system development to create a smart and responsive network intrusion detection system. The focus was on analyzing real-time network traffic, identifying threats using advanced detection strategies, and adapting to potential risks as they evolve. Each part of the system—from capturing and processing data to flagging and reporting suspicious activity—was designed to work smoothly together. The result is a flexible, user-friendly solution that not only reacts to current cyber threats but also learns and improves over time.

A. Data Collection

Traffic Sources: Collect data from network components like switches, routers, firewalls, and access points.

Data Types:

- Packet headers and payloads.
- Protocol-level details (e.g., TCP, UDP, HTTP, FTP).
- Flow-level data (e.g., NetFlow, IPFIX).

B. . Data Preprocessing

- Packet Inspection: Perform deep packet inspection (DPI) to extract detailed information from headers and payloads.
- Feature Engineering: erive features such as connection duration, byte count, packet rate, and protocol usage.
- Noise Reduction: Filter out irrelevant traffic (e.g., background traffic, benign broadcasts).
- Normalization: Standardize feature values to improve model convergence during training.

*C. **System Integration***

- Placement:

Deploy at critical network locations such as:

- Network Gateway: To monitor inbound and outbound traffic.
- Internal Segments: To monitor lateral movement within the network.
- Real-Time Monitoring: Integrate with real-time packet capturing tools (e.g., Zeek, Snort, Suricata).
- Alerting and Response: Implement mechanisms to alert administrators of threats via dashboards, email, or SMS.

D . Continuous Learning and Adaptation

- Feedback Loop: Use new attack signatures or anomalies to retrain the model.
- Threat Intelligence: Integrate external threat intelligence feeds to stay updated on emerging threats

E. Security and Privacy

- Encryption:Encrypt captured traffic to protect sensitive data.
- Access Control:Restrict access to the NIDS system to prevent tampering.

F. Evaluation in Real-World Environments

- Deploy in a live network and monitor performance under real traffic conditions.
- Analyze detection accuracy, false-positive rates, and impact on network performance.

IV. IMPLEMENTATION DETAILS

To bring the proposed multi-model machine learning-based Network Intrusion Detection System (NIDS) to life, we followed a modular pipeline approach. The system was built in stages—starting from data collection and preprocessing, through model training and deployment, to real-time monitoring and continual learning. The entire architecture was designed with flexibility and scalability in mind, allowing it to adapt to evolving cyber threats while delivering real-time threat analysis.

Model Training:

We trained a blend of models to handle both known and unknown threats.

- **Supervised models**—including Random Forests—were trained on labeled datasets to accurately classify attack types such as DDoS

To streamline performance, we used feature selection techniques like correlation analysis and model-driven importance scoring to trim down irrelevant data and improve training efficiency.

Random Forest: it is a machine learning model used to help detect unusual or suspicious activity in network traffic-essentially spotting potential cyber-attacks.

- It is good at handling big and complex data.

- It is reliable and gives accurate, even if data is not perfect.

- It helps to identify which attributes of the dataset are most important for detecting threats like IP address and port numbers.

1. Gather Data: Collect network traffic data from the network.

2. Train the Model:

- The Random Forest creates many decision trees and combines them.
- Each tree makes a guess, and the final answer is based on the majority vote.

3. Test & Evaluate:

- Run the model on new data.
- Measure how well it catches real threats using metrics like accuracy, precision, and recall.

Accurate: it can catches both known and unknown attacks on network effectively.

Stable: It's stable and does not tend to overfit or give a bunch of false alarms.

Insightful: it shows which features (like IP addresses or ports) are most relevant for spotting intrusions.

Challenges:

Speed: it can be a bit slow with very large data.

Tuning Needed: You will have to adjust settings like how many trees to use or how deep they go.

Where It's Used:

Network Security: it will protects enterprises networks.

Server Protection: it can monitors activities on individual servers

V. RESULT

The project successfully implemented a trading system that integrates real-time data analysis, trading strategy development, risk assessment, and personalized trade recommendations.

A. Network Traffic Analysis

Upon capturing and analyzing network data, the system provides a comprehensive snapshot of traffic patterns over time. It examines key attributes such as packet size, flow duration, protocol types, and port activity, serving as the foundation for identifying normal versus abnormal behavior. This continuous analysis helps establish behavioral baselines essential for detecting anomalies or potential threats.

Figure 2. Network Traffic Analysis (analyzing network data)

B. Threat Detection

Using a multi-model approach, the system applies both anomaly detection and supervised classification techniques. For example, models like Autoencoders and Isolation Forests identify deviations from normal behavior, while Random Forest and SVM classifiers detect known threats. Alerts are generated based on defined thresholds and patterns, enabling timely identification of malicious activities like DoS attacks, port scanning, or brute-force attempts.

Figure 3. Threat Detection(it detects the network traffic of the data and shows the suspicious and malicious packets of the data)

C. Interactive User Interface

The user interface is designed to be simple, clear, and user-friendly, making it easy for anyone to monitor network traffic in real time. It uses familiar color codes—green for safe activity, orange for anything suspicious, and red for potential threats—so users can quickly spot issues without needing deep technical knowledge. Alongside live traffic data, it presents easy-to-understand summaries and visual charts that show how different protocols are being used and what the current threat levels are. This makes it easier for users to stay informed and respond quickly when something unusual happens.

D. System Performance Evaluation

To assess the overall effectiveness of the system, various performance metrics were evaluated, including the accuracy of trade signal generation, the effectiveness of risk assessments, and the quality of the trade recommendations provided.

Performance metrics recorded during testing include:

```
PS E:\NIDS_Project> cd backend
PS E:\NIDS_Project\backend> python model_training.py
🚀 Starting Model Training...
✅ Dataset Loaded Successfully.
✅ Model Accuracy: 0.465
🔹 Classification Report:
              precision    recall  f1-score   support

           0       0.47      0.47      0.47       100
           1       0.46      0.46      0.46       100

    accuracy                           0.47       200
   macro avg       0.46      0.46      0.46       200
weighted avg       0.46      0.47      0.46       200

💾 Model saved as 'intrusion_detection_model.pkl'
✅ Model Training Completed.
```

VI. DISCUSSION

Our implementation showed that a modular AI-based system can make a real difference in how effectively we detect and respond to network threats. By combining different machine learning models—both supervised and unsupervised—we were able to build a system that not only improves the accuracy of intrusion detection but also scales well for complex, high-traffic networks. With multithreaded packet analysis and real-time data processing, the system stayed responsive and stable even under heavy simulated network loads.

Of course, we ran into challenges—like dealing with noisy traffic data and unpredictable, zero-day attacks. To handle these, we fine-tuned our features, used ensemble techniques, and set up regular retraining cycles to keep the models up to date. Looking ahead, we're excited about the potential of incorporating transformer-based models for even smarter threat detection and building adaptive agents that learn directly from ongoing network activity. This will help create a more responsive and resilient intrusion detection system.

VII. FUTURE RESEARCH ASPECTS

A. Adaptive Threat Learning Models

Future intrusion detection systems will benefit from adaptive learning models that personalize threat detection based on evolving network behavior. These models can continuously learn from user-specific patterns, anomalies, and traffic shifts to deliver tailored security responses in real-time, enhancing overall system intelligence and resilience.

B. Scalable Multi-Agent Architectures

To support larger and more complex network environments, future IDS implementations should scale using distributed and lightweight agents. Research into scalable orchestration, fault-tolerance, and efficient agent communication will be key for real-time protection across enterprise-level infrastructures.

C. Cross-Domain Security Intelligence

Intrusion detection can be extended beyond traditional networks to include IoT, cloud-native services, and hybrid infrastructures. Cross-domain agents can aggregate and contextualize security data across diverse platforms, providing a more holistic and intelligent view of an organization's threat landscape.

D. Real-Time Risk Scoring and Response

Integrating continuous risk scoring using edge computing and real-time data streams will significantly enhance detection latency and responsiveness. Such systems would dynamically recalculate threat levels based on live packet behavior and traffic anomalies, improving the accuracy of on-the-fly defense mechanisms.

E. Advanced NLP for Threat Intelligence

Leveraging natural language processing (NLP) for parsing cybersecurity reports, advisories, and threat feeds can enrich the contextual awareness of IDS platforms. Fine-tuned language models could automatically correlate textual threat data with network activity, aiding proactive defense.

F. Resilient Detection During Cyber Crises

AI systems should be robust against high-impact and unpredictable events such as coordinated attacks or zero-day exploits. Training detection models on synthetic data that simulates such crises will help ensure resilience and readiness in worst-case scenarios.

G. Sustainable and Efficient AI Practices

As IDS solutions scale, optimizing energy consumption and computational efficiency will be crucial. Research into green AI—such as lightweight model training and efficient deployment pipelines—will support the development of sustainable, high-performance cybersecurity tools.

VIII. FUTURE ENHANCEMNTS

AI-powered intrusion detection systems are set to become more advanced and smarter. These systems will be able to learn and improve on their own, staying one step ahead of new threats without needing humans to constantly update them.

- Predicting Attack Before They Happen: security that can predict attacks before they happen, thanks to analyzing user and system behavior patterns with machine learning.
- Real Time Threat Intelligence System: They'll also be connected to global security databases, so they stay updated on the latest threats in real time.
- Instant Response On Threat: When a threat pops up, these next-gen systems won't just sound an alert - they'll act fast to block access or quarantine compromised parts of the network instantly.
- Smart Analysis Of Malicious Files: Deep learning will help AI understand even the sneakiest malware's tricks, making it tougher for attackers to sneak by.
- Total Security Monitoring: Security teams will not only know something's wrong but also understand why, making it easier to respond effectively. The entire security picture will be more complete, constantly analyzing network traffic, user activity, and logs to get a full view of what's happening.
- Fast Detection: Detection will also get faster, especially at the edges of the network like routers and gateways, catching threats right at the source with minimal delay.
- Strong Defence: Stand strong against tricky attacks, these systems will be designed to resist manipulation attempts, thwarting hackers trying to fool or poison the AI's learning process.

IX. CONCLUSION

This paper explores the design and implementation of an AI-powered intrusion detection system built to strengthen cybersecurity in today's increasingly complex digital environments. Using a modular, multi-agent approach, the system analyzes network traffic in real time, leveraging machine learning to detect threats with high accuracy and minimal false alarms. Its user-friendly interface allows even non-technical users to understand potential risks through clear visual cues and interactive dashboards, making security monitoring more accessible and intuitive.

The system brings together specialized AI agents that handle different tasks—from identifying unusual traffic patterns to recommending responses—working together to create a smart and responsive defense mechanism. Built using scalable, open-source technologies like Flask, Kafka, and TensorFlow, the platform is designed to grow and adapt alongside evolving cyber threats.

Overall, this project shows how AI can go beyond automation to become a collaborative partner in protecting networks. It lays the groundwork for future systems that are not only more intelligent and adaptive but also more personalized and resilient in defending against the ever-changing landscape of cyberattacks.

REFERENCES

[1] R. Jain and H. Shah. "An anomaly detection in smart cities modeled as wireless sensor network". In International Conference on Signal and Information Processing (IConSIP), pages 1–5, Oct 2016.E.

[2] C. Ioannou, V. Vassiliou, and C. Sergiou. "An intrusion detection system for wireless sensor networks". In 24th International Conference on Telecommunications (ICT), , pages 1–5, May 2017.

[3] Ahmad Javaid, Quamar Niyaz, Weiqing Sun, and Mansoor Alam. "A deep learning approach for network intrusion detection system". In Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (Formerly BIONETICS), pages 21–26, 2016.

[4] *C. Yin, Y. Zhu, J. Fei, and X. He. A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access, 5:21954– 21961, 2017.*

[5] Safa Otoum, Burak Kantarci and Hussein Mouftah "A Comparative Study of AI-based Intrusion Detection Techniques in Critical Infrastructures". In ACM Transactions on Internet Technology (TOIT), Volume 21, Issue 4 Article No.: 81,

[6] Pages 1 - 22.

[7] Rachid Tahri, Youssef Balouki, Abdessamad Jarrar, and Abdellatif Lasbahani "Intrusion Detection System Using machine learning Algorithms". In ITM Web of Conferences 46, 2022.

[8] Rafeeq Ahmad, Humayun Salahuddin, Attique Ur Rehman, Abdul Rehman, Muhammad Umar Shafiq, M Asif Tahir, and Muhammad Sohail Afzal "Enhancing Database Security through AI-Based Intrusion Detection System". In Journal of Computing & Biomedical Informatics Volume 07 Issue 02, 2024.12.8.

[9] BO-Xiang Wang, Jiann-Liang Chen and Chiao-Lin Yu "An AI- Powered Network Threat Detection System". In IEEE Access, 2022.

[10] Zakaria Abou El Houda, Bouziane Brik, and Sidi-Mohammed Senouci "A Novel IoT-Based Explainable Deep Learning Framework for Intrusion Detection System". In IEEE Xplore, 2022.

[11] S. Mane and D. Rao, "Explaining Network Intrusion Detection System Using Explainable AI Framework,"2021 https://www.researchgate.net/publication/350061199_ExplainiNetworkIntrusion_Detection_System_Using_Explainable_AI_Framework.

[12] M. Wang et al., "An Explainable Machine Learning Framework for Intrusion Detection Systems," IEEE Access, vol. 8, 2020, pp. 73,127–41.

[13] S. Wali and I. Khan, "Explainable AI and Random Forest Based Reliable Intrusion Detection System Detection System," Dec. 2021. DOI:10.36227/ techrxiv.17169080.v1.

[14] K. Amarasinghe and M. Manic, "Improving User Trust on Deep Neural Networks Based Intrusion Detection Systems," Proc. IECON 2018 — 44th Annual Conf. IEEE Industrial Electronics Society, 2018, pp. 3262–68.

[15] D. L. Marino et al., "An Adversarial Approach for Explainable Ai in Intrusion Detection Systems," 2018. DOI: 10.1109/IECON.2018.8591457.

[16] Mohammed Mahmoud "THE RISKS AND VULNERABILITIES OF ARTIFICIAL INTELLIGENCE USAGE IN

[17] INFORMATION SECURITY". In International Conference on Computational Science and Computational Intelligence (CSCI),2023.

[18] Nandini, C., and Shiva Sumanth Reddy. "Detection of Communicable and NonCommunicable Disease Using Lenet- Bi-Lstm Model in Pathology Images." International Journal of System Assurance Engineering and Management, springer India- 2022, doi:10.1007/s13198-02201702-5. (Q3 journal).

[19] Kumar, P. R., Meenakshi, S., Shalini, S., Devi, S. R., & Boopathi, S. (2023). Soil Quality Prediction in Context Learning Approaches Using Deep Learning and Blockchain for Smart Agriculture. In R. Kumar, A. Abdul Hamid, & D.

[20] Binti Ya'akub (Eds.), Effective AI, Blockchain, and E- Governance Applications for Knowledge Discovery and Management (pp. 1-26). IGI Global Scientific Publishing.

[21] Shantakumar Patil, Nagaraj M Lutimath, D Jogish, Premjyoti, Bhargav S Patil, “Prediction of Heart Disease Using Hybrid Naïve Bayes Technique”, IEEE 22nd International Symposium on Communications and Information Technologies (ISCIT), Sydney, Australia, 16th - 18th Oct 2023, pp. 257-261.

[22] N. Kumar, P. Nandihal, M. R. B, P. K. Pareek, N. T and S. S. R, "A Novel Machine Learning-Based Artificial Voice Box," 2022 Second International Conference on Advanced Technologies in Intelligent Control, Environment, Computing & Communication Engineering (ICATIECE), Bangalore, India, 2022, pp. 1-7, doi: 10.1109/ICATIECE56365.2022.10046967.

[23] Decentralized Malware Attacks Detection using Blockchain S. Sheela, S. Shalini, D. Harsha, V.T. Chandrashekar, Ayush Goyal ITM Web Conf. 53 03002 (2023)DOI:10.1051/itmconf/20235303002.