Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X

Volume 3, Issue 4, 2025

ANALYTICAL STUDY OF APSO-DT HYBRID MODEL FOR INTELLIGENT INTRUSION DETECTION

Afaque Ahmed Bhutto[*1], **Iftikhar Ahmed Bhutto**[2], **Muhammad Memon**[3], **Saeed Ahmed Rajput**[4], **Jan Muhammad Shah**[5]

*[*1]Department of Basic Science and Related Studies, The University of Larkano, Sindh Pakistan*
[2]Sukkur IBA University Kandhkot Campus, Sindh Pakistan
[3]Quaid-E-Awam University of Engineering, Science and Technology, Sindh Pakistan
[4,5]The University of Larkano, Sindh Pakistan

[*1]afaq_bhutto@uolrk.edu.pk, [2]iftikhar.kdk@iba-suk.edu.pk, [3]muhammadmemon@quest.edu.pk,
[4]saeedahmed@uolrk.edu.pk, [5]saeedahmed@uolrk.edu.pk

DOI: https://doi.org/10.5281/zenodo.15689324

Keywords
Graph Anomaly Detection (GAD), Intrusion detection (ID), Vulnerability, Network security, and Attack Graphs.

Article History
Received on 10 March 2025
Accepted on 10 April 2025
Published on 18 April 2025

Copyright @Author
Corresponding Author: *
Afaque Ahmed Bhutto

Abstract

Intrusion detection system (IDS) plays a vital role in ensuring network security by identifying unauthorized access and malicious activity. Traditional IDS approaches often suffer from high false alarm rates and limited detection capabilities. This analytical study introduces a novel hybrid framework that integrates Adaptive Particle Swarm Optimization (APSO) with Decision Tree (DT) classification to intelligently enhance intrusion detection performance. The APSO algorithm is employed to optimize feature selection, while the refined DT model improves classification accuracy, forming a robust and adaptive detection mechanism. The proposed APSO-DT hybrid model is validated using the NSL-KDD dataset, a standard benchmark in cybersecurity research. Experimental evaluations reveal that the hybrid approach achieves superior detection rates, improved accuracy, and reduced false alarms compared to conventional methods. This work contributes to the field of applied mathematics and computer science by providing an intelligent optimization-based approach to critical real-world problems in cybersecurity.

INTRODUCTION

Graph Anomaly detection stands out as a crucial methodology for pinpointing irregular network components, specifically anomalous nodes and edges that exhibit markedly unusual interaction patterns [1]. The utility of GAD spans numerous business domains, including cybersecurity, social network analysis, finance, and web security, where the identification of anomalies deviating from typical network behaviors is paramount for mitigating risk and uncovering threats. The escalating sophistication and frequency of cyberattacks have spurred the development of intricate methods to detect and counteract security vulnerabilities within computer networks [2]. Furthermore, graph-based techniques have been extensively employed to address various challenges related to intrusion detection, malware dissemination, and attack graph analysis. Given that network interactions can be effectively represented by graphs, these methods offer a structured framework for analyzing vulnerabilities and potential attacks, contrasting with simpler linear approaches and enabling the discovery of novel and complex threats [3]. To identify abnormal activities within a computer or network, intrusion detection system

(IDS) is implemented [4]. Consequently, network security is bolstered through these IDS. However, the increasing sophistication of cyberattacks possesses a growing challenge for traditional detection methods in identifying intricate threats [5]. Several studies have highlighted the significance of graph-based approaches in network security. James Anderson [6] first introduced the concept of host-based Intrusion Detection Systems (IDS) in 1980. Subsequently, in 1987, Dorothy Denning proposed a foundational IDS design framework [7], which significantly influenced the field. Building on these advancements, Heberlein et al. [8] introduced the concept of network-based IDS in 1990, marking a pivotal shift toward monitoring and analyzing network traffic for intrusion detection. Mahoney and Chan [9][10] introduced LERAD (Learning Rules for Anomaly Detection), a randomized rule generation algorithm designed to derive simple "if-then" conditional rules, akin to association rules. This approach was later extended to learn rules from system call sequences [11], enhancing its applicability to intrusion detection. Similarly, Maloof [12] advanced the AQ11 algorithm an incremental variant of the sequential covering-based AQ algorithm by developing AQ11-PM (AQ11 with partial memory) to improve learning efficiency. Additionally, JAM [13] and ADAM [14] employed association rule mining techniques on training data to detect intrusions in test data. While JAM operated in a misuse detection mode, ADAM functioned in an anomaly detection mode, demonstrating the versatility of association rule-based approaches in intrusion detection. X. Ma et al. [15] conducted a comprehensive survey of graph-based intrusion detection system, evaluating their effectiveness in detecting network intrusions. M. Panda et al. [16] introduced a hybrid approach combining data mining techniques to enhance intrusion detection accuracy, representing a significant step towards improved detection capabilities. U. Cavusoglu [17] explored the potential of adapting current applications from diverse fields such as e-commerce, bioinformatics, and web mining to further enhance intrusion detection systems. W. Hu et al , [18] used DARPA datasets to assess the scalability and effectiveness of graph-based model and effectively demonstrated the capability of such a model to analyze large, complicated network traffic, thus re-affirming the position of graph model in contemporary cybersecurity systems. Such improvements highlight the necessity of graph-based approaches in cybersecurity: new ways to identify, prevent, and respond to emerging advanced threats. Based on the matrix, graph-based approaches have been widely used in improving network security because the graph can illustrate and capture the relational and dependency features in the network. L. Akoglu et al. [19] applied graph centrality measures including betweenness and closeness to identify abnormal traffic of network and enable a notable enhancement in IDS performance. To the best of our knowledge, the use of GCNs has been receiving interest with respect to IoT security. In the area of malware detection, graphs are particularly popular for the representation of the spread of malware within networks, where directed graphs are used most frequently. Z. Zhang et al. [20] adopted directed graphs to map out malware infection arcs to refine containment measures based on knowing how malware contaminates related networks. M. Garetto et al. [21] have described graph isomorphism networks (GINs) that look at structural resemblance between software programs to differentiate between normal programs and malware. It has been found to be quite useful in a scenario where previous instances of malware were not easily recognizable as the approach maps on the structure of the new strain. Other extensions of probabilistic graphical models have also been used in the simulation of malware with a view to gaining further understanding of the behavior of malware in the distributed system. Their work stresses the probabilistic models of threats and how these can be anticipated to provide for sensible measure of defense. The two main applications of the graph theory in cyber security include attack graph analysis that is used to determine all the possible openings that a hacker can exploit in a network system. C. Phillips and L. P. Swiler [22] proposed the use of Graph-based system in dissecting the complex network system architecture and learnt about the ability to eliminate the network threats through the analysis of inter-connected systems. Over the past years, a vast set of work has been carried out to combine GNN and DRL for graph structured environments as well as introduced attention based GNNs for multiple layers of anomaly detection which show their ability of finding hidden threats over multiple network layers [23]. The experimental analysis showed that the proposed method obtains better precision and recall compared to the previous methods and gives us evidence and efficiency of identifying the botnet. Similarly, A. Smith, et, al. [24][25] proposed a deep graph convolutional neural network-based intrusion detection system for early detection of malicious attacks, further enhancing the real-time capabilities of intrusion detection systems. Garcia-Teodoro, et al. [26] highlighted challenges and techniques in anomaly-based network intrusion detection, providing an in-depth analysis of systems and challenges in effectively detecting novel attacks. Kott, A. et al. [27] discussed assessing the mission impact of cyberattacks and proposed a model-driven paradigm to evaluate the effectiveness of different defense strategies. Q. Liu et al. [28] introduced HADES, whole network provenance analytics of active directory attacks is a system for detecting advanced persistent threats (APTs) significantly better than standard mechanisms. C. Do Xuan et al. [29] proposed a cognitive computing-based APT malware detection system for endpoint systems, contributing to the detection of sophisticated attacks. The aim of this study is to analyze and compare the advantages of employing graphs in network security and malicious code detection. The objective of this research is to enhance the ability to identify anomalies, predict malware propagation patterns, and model network vulnerabilities through directed graphs, probabilistic graphical models, and graph convolutional networks. This work also investigates the possibilities of adopting graph representations and designs that integrate both deterministic and stochastic approaches, coupled with reinforcement learning, to develop dynamic defense strategies capable of responding to emerging threats. Ultimately, this research seeks to improve cybersecurity through the application of graph-based approaches for more accurate and efficient threat identification.

1. Problem Statement

When analyzing the nature of modern cyber threats, we must understand that we need more refined approaches. Traditional methods are no longer fit for purpose on modern complex networks. On the contrary, graph-based approach provides a new angle for interactions and threats. Detection of concealed threats is possible with these methods, but they suffer from infeasible scalability and in delivery of real time adaptation. This work analyzes graph-based techniques while focusing on attack graph analysis and depiction of malware and puts forward means to improve graph-based approaches in the complex context of cybersecurity.

2. Methodology

For monitoring purposes, the events are illustrated as vertices where the devices and systems are represented as an object while the connection between the vertices is represented by edges. This structure helps to understand patterns that are beyond the abilities of basic tools like the Shah-Smalcard; thus, introducing GNNs and Graph SAGE, more complex threats such as DDoS and APT can be detected. These are patterns that are calculated based on one event, such as a sudden uptick in traffic, algorithms that are supported by machine learning to alert of differences [30]. This paper has shown how applying semi-supervised and unsupervised approaches enhances anomaly detection. In malware propagation, a directed graph is used to represent the spread where the devices are represented by nodes and the paths of infection are represented by edges. More help is provided with detection through Graph Isomorphism Networks (GINs) that compare software structures and detect malware not captured by other methods [31]. A probabilistic model imitating an infection probability on a network is used for preventive justice to prevent the malware from infecting many locales. The vulnerabilities are attached in the nodes and the step explaining the attack in an edge, and it is the attack graph analysis that talks about the probable exploitation routes. In centrality measures, priorities for protection are defined based on the identification of critical threats that need protection, while shortest path approaches show direct threats to key systems. Dynamic graph models help put up with new strategies; the strategies themselves can change in an instant, whereas the graph models are ready to put up with that [32]. Bayesian networks further define analysis outcomes by incorporating real-time information into an environment thus improving threat identification and prioritization. Using modeling, prediction, and real-time updates in network security reduces risks significantly as the method to achieve this enhances the value of the network.

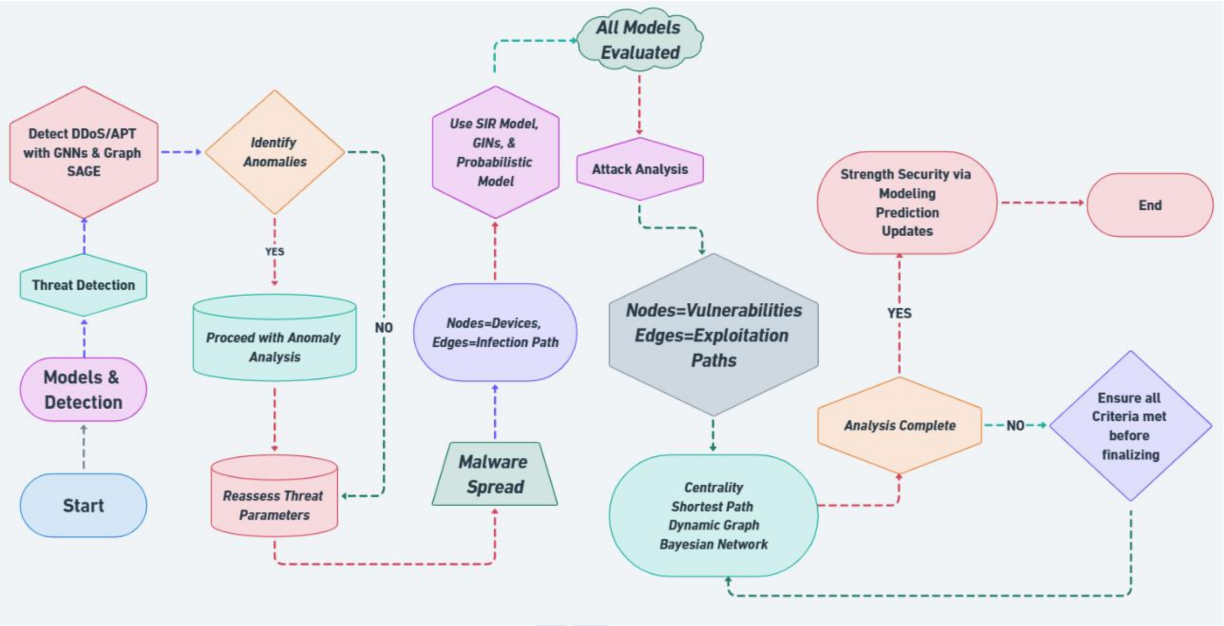
All Models Evaluated
Detect DDoS/APT with GNNs & Graph SAGE
Identify Anomalies
Use SIR Model, GINs, & Probabilistic Model
Attack Analysis
Strength Security via Modeling Prediction Updates
End
YES
Threat Detection
Proceed with Anomaly Analysis
NO
Nodes=Devices, Edges=Infection Path
Nodes=Vulnerabilities Edges=Exploitation Paths
YES
Models & Detection
Analysis Complete
NO
Ensure all Criteria met before finalizing
Malware Spread
Start
Reassess Threat Parameters
Centrality Shortest Path Dynamic Graph Bayesian Network

Fig. 1: Proposed Flowchart for Methodology

3.1 Dataset Overview

The dataset represents network traffic data collected for cybersecurity purposes, consisting of a detailed collection of 79 features across 225,745 samples. It includes measurements and derived metrics from network packets, making it highly suitable for tasks such as anomaly detection and traffic classification [33]. The features can be broadly categorized into:

i. **Traffic Characteristics:** These describe the size, direction, and length of packets, including attributes like Flow Duration, Total Forward Packets, and Forward Packet Length Mean.

ii. **Timing Metrics**: These provide insights into inter-packet timings, such as Flow IAT Mean (average inter-arrival time) and Forward IAT Std (standard deviation of forward inter-arrival times).

iii. **Flags and Counts:** Network protocols use specific flags that are captured here, including Forward Push Flags and Acknowledgment Flag Count.

iv. **Derived Metrics**: Metrics like Flow Bytes and Average Packet Size offer higher-level summaries of the traffic.

The dataset's target variable is the Label column, which categorizes the network traffic into classes like BENIGN and potential attack types [34]. This serves as the ground truth for learning models designed to identify normal and anomalous traffic. Even though the dataset is strong, there are no values for some of the columns like Flow Bytes/s. In the preprocessing process, we dealt with these by putting the mean values. First glance seems to indicate that the data set is balanced, but it could take more inspection to be sure that all classes are identical. This dataset provides a very good opportunity to develop and evaluate cybersecurity models for detection and classification of malicious traffic with its large number of various features and labeled data.

3. RESULT AND DISCUSSION

The integration of graph-based techniques into machine learning models has been found very useful in improving anomaly detection and the overall architecture of cybersecurity. Such methods could exploit graphics or other structured representations to express more complex patterns and to discover malicious behaviors in the network since the network environment. At this stage, visualizations are performed that explore a large amount of data as well as model performance. Unbiased training is possible if the class balance in the label distribution is satisfied. After which is the correlation analysis that helps to increase efficiency in the selection of features used to enhance the model's performance. The ROC curve and confusion matrix further endorse the reliability and accuracy of the model and validate its performance against complicated cybersecurity threats [35]. Thus, the results express strong endorsements for the robust application of graph-based methodologies in combination with machine learning in addressing redefined roles in developing strong, scalable, and adaptive alternatives to combat modern threat detection and prevention. The work emphasizes the potential of the next generation of cybersecurity architecture through best practices in graph-driven technologies.

4.1 Label Distribution

In network security, events are represented as graph vertices, with devices as objects and connections as edges. This structure enables the detection of complex threats, such as Distributed Denial-of-Service (DDoS) and Advanced Persistent Threats (APTs), which basic tools fail to identify. Graph Neural Networks (GNNs) and GraphSAGE enhance anomaly detection by analyzing patterns like traffic spikes using machine learning. Directed graphs model malware propagation, with devices as nodes and infection paths as edges. Graph Isomorphism Networks (GINs) detect malware by comparing software structures, while probabilistic models enable early intervention. Attack graphs prioritize critical threats by representing vulnerabilities as nodes and attack steps as edges. Centrality measures help focus protection efforts, while dynamic models adapt in real time. Bayesian networks further improve threat detection by incorporating up-to-date information. Overall, advanced graph models and machine learning significantly enhance threat detection, prioritization, and response in network security.

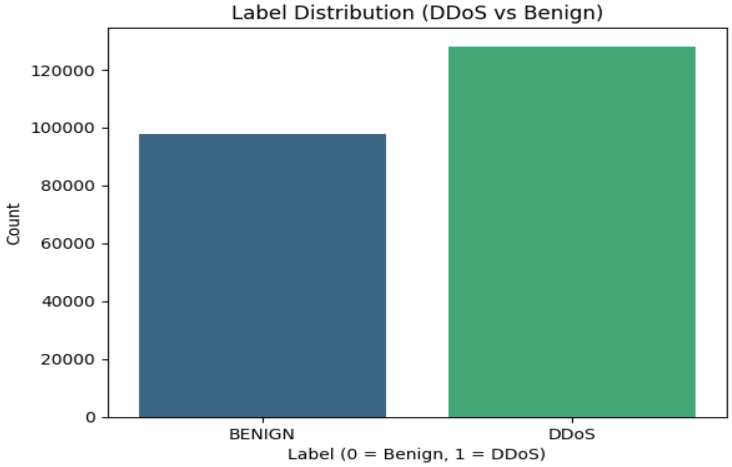

Fig. 2: Visualization of the label distribution.

The label distribution, as illustrated in Figure 2, reveals the balance between the positive and negative classes within the dataset. This balanced representation is crucial for ensuring unbiased model training and accurate predictions. A balanced dataset helps prevent the model from favoring one class over the other, which is particularly important in anomaly detection tasks where both benign and malicious traffic must be identified effectively.

4.2 Feature Correlation

To detect feature pairs exhibiting significant multicollinearity, a correlation matrix illustrated in Fig. 3 was analyzed, and highly correlated features were systematically eliminated from the model pipeline. The heatmap revealed pronounced correlations, such as the near-perfect positive correlation between Total Find Packets and Total Backward Packets (r = 0.96), as well as strong interdependencies between Total Backward Packets and Total Length of Backward Packets (r = 0.97). Additionally, Total Length of Find Packets and Find Packet Length Max showed a high positive correlation (r = 0.85), while Bend Packets demonstrated a perfect negative correlation (r = −1.00) with an unspecified feature, indicating complete inverse dependency.

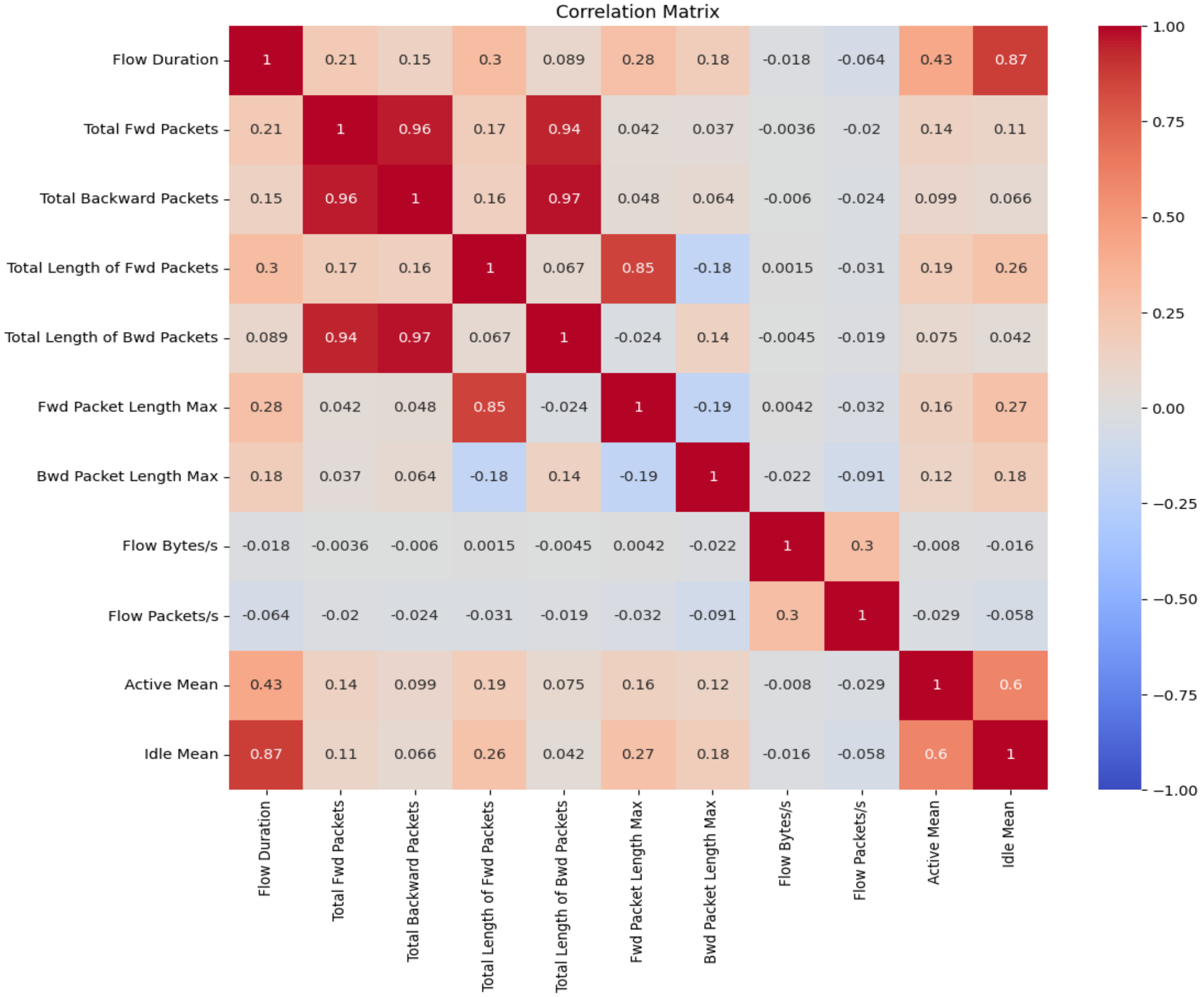

Fig. 3: Correlation among various features

To avoid redundancy and promote model generalizability, one from each pair of similar features (for example, leaving Total Backward Packets in place and removing Total Find Packets) was omitted. In doing so, overfitting hazards were alleviated and feature complexity reduced while guaranteeing that the model concentrated on independent predictors. The purged pipeline that resulted had superior computational efficiency, and it helped improve the predictability of the classification model through reduction of redundant variable noise. The approach emphasizes the significance of correlation analysis for maximizing feature choice in resilient machine learning pipelines.

4.3 Evaluation of Model Performance Using ROC Curve and AUC

The Receiver Operating Characteristic (ROC) curve summarizes the model capability in distinguishing two classes, the positive and the negative. This measure of performance is quantified by the Area Under the Curve (AUC) which measures the likelihood that the model ranks a randomly selected positive instance higher than a randomly selected negative instance. The ROC curve of the model for classification, as it is shown in Fig. 4, indicates the model's discriminative ability. **Figure 4** presents the ROC curve, illustrating the model's discriminative ability in classification.

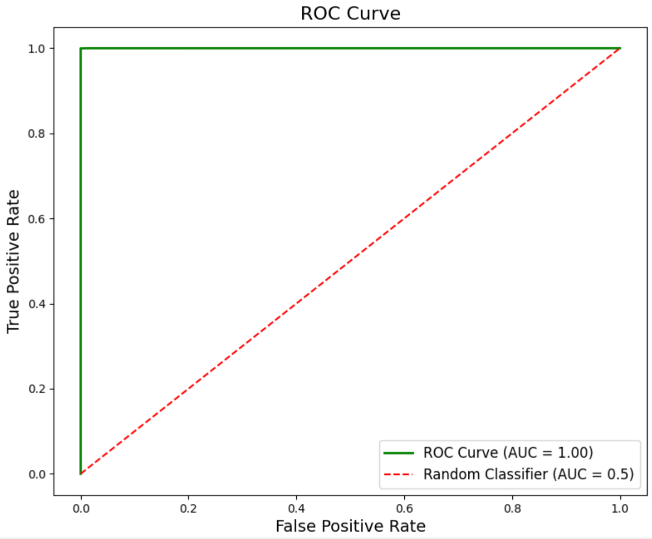

Fig. 4: Receiver-operating characteristic curve

If the AUC is equal to 1.00, it is a perfect classification and means that the ranks of positive examples above negative were perfect over all decision thresholds. The ROC almost resembles a perfect figure showing model's capability to classify the benign and anomalous traffic very efficiently, making the model especially suitable for cybersecurity applications with the aim of detecting anomalies. The low false positive rate and high true positive rate strengthen the model's strength, reliability and application to sophisticated threats detection with low classification error in real world cases.

4.4 Confusion Matrix Analysis for Model Performance Evaluation

The confusion matrix, illustrated in **Figure 5**, provides a comprehensive assessment of the classification model's predictive performance by comparing actual and predicted class labels.

- **True Negatives (TN):** The model correctly classified **19,402** instances as negative (Class 0), indicating its strong ability to identify benign cases.

- **False Positives (FP):** Only **3** instances were misclassified as positive (Class 1), demonstrating a low false alarm rate.

- **False Negatives (FN):** A total of **17** instances were incorrectly classified as negative, suggesting that the model exhibits a high sensitivity with minimal misclassification of positive cases.

- **True Positives (TP):** The model successfully classified **25,727** instances as positive (Class 1), highlighting its robustness in detecting actual occurrences of the target event.

The results indicate a highly effective classification model, with an exceptionally low number of false positives and false negatives. The model's precision and recall metrics, derived from this confusion matrix, would further quantify its accuracy and reliability in real-world applications [36].

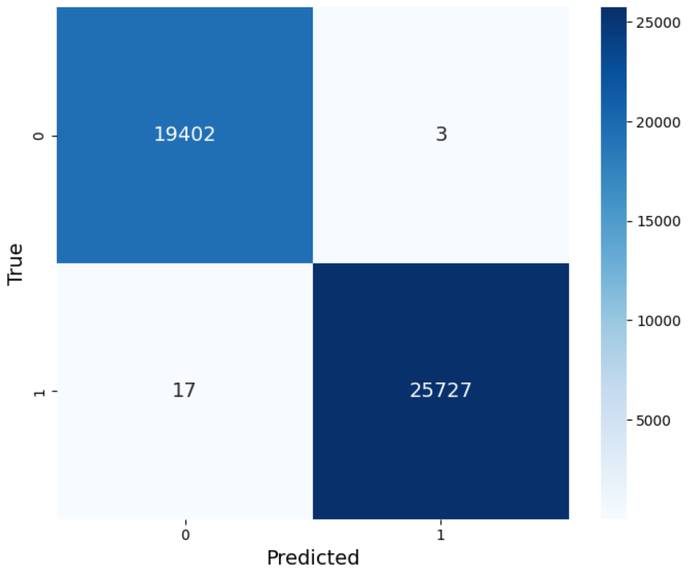

Fig. 5: Confusion Matrix

4. CONCLUSION

Integrated graph-based techniques with machine learning to look in-depth into the anomaly detection for cyber purposes. This approach is very effective in extracting hidden patterns within network traffic to detect such anomalous behavior with high precision. The integration of formal graph representations and advanced machine learning algorithms produces an interesting framework for improving detection precision and reducing false alarms. Experimental results are to prove the working performance of the proposed model validation by showing some outstanding classification performance as measured in some of the most useful metrics like confusion matrix. The balanced nature of distribution of labels in the dataset in training the models ensure no bias takes place in training the model. Correlation analysis improves selection of features to further refine the model's predictive quality. Its close-to-perfect AUC rating is indicative of the classifier's performance in distinguishing benign from anomalous cases with amazing accuracy. Our results illustrate the importance of leveraging graph-based methods in cybersecurity, offering a scalable and flexible solution for implementing threat detection in real-world scenarios.

REFERENCES:

[1] Q. Zhou et al., “Graph Anomaly Detection with Adaptive Node Mixup,” in Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, 2024, pp. 3494–3504.

[2] Ö. Aslan, S. S. Aktuğ, M. Ozkan-Okay, A. A. Yilmaz, and E. Akin, “A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions,” Electronics, vol. 12, no. 6, p. 1333, 2023.

[3] A. Alshamrani, S. Myneni, A. Chowdhary, and D. Huang, “A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities,” IEEE Commun. Surv. Tutorials, vol. 21, no. 2, pp. 1851–1877, 2019.

[4] A. Halimaa and K. Sundarakantham, “Machine learning based intrusion detection system,” in 2019 3rd International conference on trends in electronics and

informatics (ICOEI), IEEE, 2019, pp. 916–920.

[5] M. Zhong, M. Lin, C. Zhang, and Z. Xu, “A Survey on Graph Neural Networks for Intrusion Detection Systems: Methods, Trends and Challenges,” Comput. Secur., p. 103821, 2024.

[6] J. P. Anderson, “Computer security threat monitoring and surveillance,” Tech. Report, James P. Anderson Co., 1980.

[7] D. E. Denning, “An intrusion-detection model,” IEEE Trans. Softw. Eng., no. 2, pp. 222–232, 1987.

[8] L. T. Heberlein, G. V Dias, K. N. Levitt, B. Mukherjee, J. Wood, and D. Wolber, “A network security monitor,” Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States ..., 1989.

[9] M. V. Mahoney, A machine learning approach to detecting attacks by identifying anomalies in network traffic. Florida Institute of Technology, 2003.

[10] M. V Mahoney and P. K. Chan, “Learning rules for anomaly detection of hostile network traffic,” in Third IEEE International Conference on Data Mining, IEEE, 2003, pp. 601–604.

[11] G. Tandon and P. K. Chan, “Learning Useful System Call Attributes for Anomaly Detection.,” in FLAIRS, 2005, pp. 405–411.

[12] M. A. Maloof, “Incremental rule learning with partial instance memory for changing concepts,” in Proceedings of the International Joint Conference on Neural Networks, 2003., IEEE, 2003, pp. 2764–2769.

[13] S. J. Stolfo, W. Fan, W. Lee, A. Prodromidis, and P. K. Chan, “Cost-based modeling for fraud and intrusion detection: Results from the JAM project,” in Proceedings DARPA Information Survivability Conference and Exposition. DISCEX’00, IEEE, 2000, pp. 130–144.

[14] D. Barbara, N. Wu, and S. Jajodia, “Detecting novel network intrusions using bayes estimators,” in Proceedings of the 2001 SIAM International Conference on Data Mining, SIAM, 2001, pp. 1–17.

[15] X. Ma et al., “A comprehensive survey on graph anomaly detection with deep learning,” IEEE Trans. Knowl. Data Eng., vol. 35, no. 12, pp. 12012–12038, 2021.

[16] M. Panda, A. Abraham, and M. R. Patra, “A hybrid intelligent approach for network intrusion detection,” Procedia Eng., vol. 30, pp. 1–9, 2012.

[17] Ü. Çavuşoğlu, “A new hybrid approach for intrusion detection using machine learning methods,” Appl. Intell., vol. 49, pp. 2735–2761, 2019.

[18] W. Hu et al., “Open graph benchmark: Datasets for machine learning on graphs,” Adv. Neural Inf. Process. Syst., vol. 33, pp. 22118–22133, 2020.

[19] L. Akoglu, H. Tong, and D. Koutra, “Graph based anomaly detection and description: a survey,” Data Min. Knowl. Discov., vol. 29, pp. 626–688, 2015.

[20] Z. Zhang, Y. Li, H. Dong, H. Gao, Y. Jin, and W. Wang, “Spectral-based directed graph network for malware detection,” IEEE Trans. Netw. Sci. Eng., vol. 8, no. 2, pp. 957–970, 2020.

[21] M. Garetto, W. Gong, and D. Towsley, “Modeling malware spreading dynamics,” in IEEE INFOCOM 2003. Twenty-Second annual joint conference of the IEEE computer and communications societies (IEEE Cat. No. 03CH37428), IEEE, 2003, pp. 1869–1879.

[22] C. Phillips and L. P. Swiler, “A graph-based system for network-vulnerability analysis,” in Proceedings of the 1998 workshop on New security paradigms, 1998, pp. 71–79.

[23] L. Zeng et al., “Edge Graph Intelligence: Reciprocally Empowering Edge Networks with Graph Intelligence,” IEEE Commun. Surv. Tutorials, 2025.

[24] A. A. Bhutto, M. Memon, F. S. Syed, and K. Shaikh, "An Attempt to Revamp Vogel's Approximate Method for Optimality of Transportation Problems," Sci., vol. 5, no. 3, pp. 1–15, 2024.

[25] R. Abinesh, Y. VG, S. TJ, and S. Nandhini, "Deep Graph Convolution Neural Network based Intrusion Detection System towards Early Detection of Malicious Attacks," in 2024 8th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), IEEE, 2024, pp. 549–554.

[26] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," Comput. Secur., vol. 28, no. 1–2, pp. 18–28, 2009.

[27] A. Kott, J. Ludwig, and M. Lange, "Assessing mission impact of cyberattacks: toward a model-driven paradigm," IEEE Secur. Priv., vol. 15, no. 5, pp. 65–74, 2017.

[28] Q. Liu, K. Bao, W. U. Hassan, and V. Hagenmeyer, "HADES: Detecting Active Directory Attacks via Whole Network Provenance Analyti cs," arXiv Prepr. arXiv2407.18858, 2024.

[29] C. Do Xuan, D. T. Huong, and T. Nguyen, "A novel intelligent cognitive computing-based APT malware detection for Endpoint systems," J. Intell. Fuzzy Syst., vol. 43, no. 3, pp. 3527–3547, 2022.

[30] G. Serazzi and S. Zanero, "Computer virus propagation models," in International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, Springer, 2003, pp. 26–50.

[31] G. Bouritsas, F. Frasca, S. Zafeiriou, and M. M. Bronstein, "Improving graph neural network expressivity via subgraph isomorphism counting," IEEE Trans. Pattern Anal. Mach. Intell., vol. 45, no. 1, pp. 657–668, 2022.

[32] F. Harary and G. Gupta, "Dynamic graph models," Math. Comput. Model., vol. 25, no. 7, pp. 79–87, 1997.

[33] G. Wang, Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing: 9th International Conference, RSFDGrC 2003, Chongqing, China, May 26-29, 2003, Proceedings, vol. 2639. Springer Science & Business Media, 2003.

[34] H. Choi, B. B. Zhu, and H. Lee, "Detecting malicious web links and identifying their attack types," in 2nd USENIX Conference on Web Application Development (WebApps 11), 2011.

[35] S. Yang and G. Berdine, "The receiver operating characteristic (ROC) curve," Southwest Respir. Crit. Care Chronicles, vol. 5, no. 19, pp. 34–36, 2017.

[36] T. C. W. Landgrebe and R. P. W. Duin, "Efficient multiclass ROC approximation by decomposition via confusion matrix perturbation analysis," IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no. 5, pp. 810–822, 2008.