Advances in Computational Design, *Vol. 11, No. 3 (2026) 249-275*
https://doi.org/10.12989/acd.2026.11.3.249

Optimized secure routing with intrusion resilience via deep reinforcement learning for cyber threat detection

Jayalakshmi Sambandam*[1], S. Malathi[2a], L. Logeshwari[3b], J. Dillibabu[4c], Lithin Kumble[5d]

[1]*Department of Computer Science and Engineering (Emerging Technologies), SRM Institute of Science and Technology (SRM IST), Vadapalani Campus, Chennai – 600026, Tamil Nadu, India*
[2]*Department of Computer Science, St. Thomas College of Arts and Science, Chennai – 600107, Tamil Nadu, India*
[3]*Department of Computer Science with Data Science, Tagore College of Arts and Science, Chennai – 600044, Tamil Nadu, India*
[4]*Department of Computer Applications, St. Thomas College of Arts and Science, Chennai – 600107, Tamil Nadu, India*
[5]*School of Computing Science and Engineering, REVA University, Bengaluru – 562157, Karnataka, India*

(Received May 5, 2026, Revised June 16, 2026, Accepted June 22, 2026)

Abstract. With the increasing number of networked devices, elevated network speeds, and sophisticated cyber threats, cybersecurity has become a difficult problem to solve. Many current approaches fall short due to inefficient data preprocessing, redundant features, low adaptability, and failure to recognize complex behavioral patterns. To address these challenges, an intelligent hybrid framework is proposed to deal with cybersecurity problems in smart networks. An Intelligent Adaptive Routing–Density Fusion Model (IARDFM) is utilized for efficiently collecting network traffic data and preprocessing it to remove noise, redundant, and outlying data while increasing quality and balance of the data. The preoptimized features are then processed by a Deep Probabilistic Cluster-Aware Behavior Learning Network (DPCB-Net) that utilizes deep learning with probabilistic clustering to perform accurate multiclass behavioral learning. A Spectral Hierarchical Swarm Feature Optimization Framework (SHSFOF) is then used to select the most relevant features for training a learning-based approach. To detect various types of cyber-attacks, a Graph Reinforced Intrusion Detection System (GRIDS) is proposed based on graph-based learning and trust-aware evaluation. A Deep Reinforcement Learning–Based Intrusion-Resilient Secure Routing (DRL-ISR²) framework is used for selectively routing packets to different destinations to prevent intrusion and increase network resilience. Comprehensive experiments and comparisons validate improved accuracy, precision, recall, and robustness of the intelligent hybrid framework for smart network cybersecurity.

Keywords: adaptive routing; cyber threat detection and prevention; density-based clustering; feature optimization and selection techniques; graph-based intrusion detection systems; probabilistic learning methods; reinforcement learning techniques for network security problems; trust evaluation models

*Corresponding author, Assistant Professor, E-mail: jayalakshmi.research@gmail.com
[a]Assistant Professor, E-mail: malathis@saintthomascollege.com
[b]Assistant Professor, E-mail: logeshwari.research@gmail.com
[c]Assistant Professor, E-mail: jdillibabu@saintthomascollege.com
[d]Associate Professor, E-mail: lithin.k@reva.edu.in

http://www.techno-press.org/acd/5
ISSN: 2383-8477 (Print), 2466-0523 (Online)

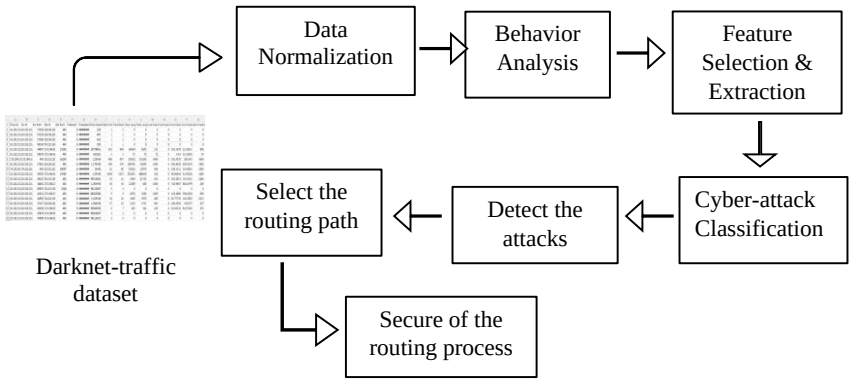

Figure 1. Deep learning–driven cyber-attack detection and secure routing framework

1. Introduction

Cyber-attacks, as malicious attempts to disrupt, intercept, or manipulate network communication, particularly in IoT networks and intelligent transportation systems, have become a major threat. Some common types of cyber-attacks include DDoS, malware injection, spoofing, phishing, and network intrusion [1-2]. Machine learning and deep learning-based prediction methods that analyze network traffic and learn spatial and temporal features to identify anomalies have been widely applied for cyber-attack detection [3-4].

However, existing approaches encounter a lot of challenges, such as high computational complexity, the inability to capture dynamic relationships between network entities, low accuracy in detecting complex and evolving cyber-attacks, the limitations of static feature representations, and lack of adaptability to dynamic network environments, which results in a high false positive and false negative rates [5-6]. In this paper, we propose an intelligent multi-stage framework for cyber-attack detection. In the first stage, we improve data collection and preprocessing to remove noise, redundancy, and irrelevant data [7]. The subsequent stage includes advanced feature optimization techniques to select the most informative features. In the third stage, deep learning models are used to analyze the selected features to learn complex behaviors of multiple attack types. In the final stage, we design an adaptive detection model to capture relationships between network entities, calculate their trust values, and improve detection accuracy through learning strategies. The integrated framework improves detection accuracy and reduces errors for cyber-attack classification.

The Fig. 1 show that the framework utilizes an integrated approach that will help detect cyber-attacks and ensure secure routing by utilizing deep learning methods on darknet data. Initially, raw datasets are subject to data normalization processes in order to improve overall quality levels as well as provide a consistent data source for analysis and identification of anomalous traffic behaviour. Following that, behavioural analysis is performed to find underlying patterns of Internet traffic and identify any anomalies therein. The next step involves feature selection and extraction so as to only retain those features that have the most impact on accurately detecting cyber-attacks; these selected and extracted features are then input into a classification model to accurately classify and detect cyber-attacks. Once the classification model has classified/detected a cyber-attack, the system will then analyse the detection information so as to navigate through the network using safe/effective routing paths, avoiding any compromised/high-risk nodes. Finally,

secure routing of communications is achieved by dynamically adjusting and adapting the routing process according to real-time changes to the state of the network and the level of threat present in the network, thus providing data that is consistently and reliably transmitted over the network while minimizing the possibility of loss or corruption of transmitted data due to being transmitted over low risk/high risk nodes or paths.

Main Contribution of the Work

◦ The creation of a hybrid intelligent cybersecurity framework that combines data preprocessing, feature optimization, multiclass behaviour analysis, graph-based detection and reinforcement learning-based secure routing to accurately and efficiently detect cyber-attacks.

◦ The development of an innovative model to preprocess data with the ability to create a dataset with minimal noise, redundancy and outlier removal, while providing a structured network traffic representation with balanced features.

◦ The ability to utilize a new feature extraction and optimization framework to identify complex relationships between data and reduce dimensionality and identifying the subset of features with the most significant information using spectral, hierarchical and swarm-based optimization techniques.

◦ The introduction of a deep learning-based model of behaviour analysis of multi-class with the ability to capture the dynamic patterns of traffic on the network and accurately differentiate between normal usage of the network and multiple types of cyber-attack.

◦ The development of an adaptive and intelligent detection and response mechanism using Graph-based Learning to detect threats and using Deep Reinforcement Learning-based Secure Routing to mitigate any possible attacks in real-time.

The first step in producing the proposed work is to identify the core issues related to cyber-attack detection; this highlights the need for an intelligent and adaptable security model. The next phase involved reviewing existing approaches and showing their limitations with regards to data preprocessing, feature representation, and detection accuracy. The methodology section provides an outline of the proposed hybrid framework, which includes the following components: IARDFM for preprocessing, SHSFOF for feature optimization, DPCB-Net for multiclass behaviour analysis, GRIDS for threat detection, and DRL-ISR² to provide secure routing and attack mitigation functionality. To will be evaluated using metrics such as accuracy, precision, recall, and error rate. Lastly, the work concludes with a summary of its major findings, limitations, and suggestions for enhancing future cybersecurity detection systems.

2. Literature survey

The SCADA (Supervisory Control and Data Acquisition) Networks. To solve the problems mentioned above, we have developed a novel self-expanding Automated Traffic Classification (ATC) model based on a newly created density-based heuristic clustering technique that can continue to automatically and continuously discover and qualify all types of unknown attack traffic initiated from various attack tools against SCADA (Supervisory Control and Data Acquisition) networks and generate valid classifications in real-time. In addition, we propose an alternative method for effectively representing SCADA traffic that may further enhance the performance of the ATC method [8]. We evaluated the new ATC method and collected large volumes of data, including SCADA network traffic data, attack tool data, and ICS (Intelligent Control System) honeypot data, and conducted multiple experiments on this combined dataset to

Table 1. Recent research on cyber-attack detection and security challenges

Author	Year	Type of Dataset	Proposed Method	Limitations
[11]	2024	Power system cyber-attack datasets (review-based)	Cyber-Attack Impact Analysis and Detection Review (CAIADR)	Lacks experimental validation; mostly theoretical analysis
[12]	2021	IDS datasets (network traffic data)	Intrusion Detection System Attribute Analysis (IDS-AA)	Limited real-time applicability; dataset dependency
[13]	2021	Smart DC microgrid data	Blockchain-Based Cyber-Attack Detection with Hilbert Huang Transform (BCAD-HHT)	High computational complexity; scalability issues
[14]	2024	Network security datasets	Comparative Attack Detection Strategy Analysis (CADSA)	No novel model proposed; lacks implementation
[15]	2023	General cyber-attack datasets	Cybersecurity Performance Evaluation Model (CPEM)	Low adaptability to dynamic attacks
[16]	2023	Encryption/security datasets	Homomorphic Encryption-Based Security Enhancement (HESE)	High processing overhead; complex implementation
[17]	2022	IoT layered datasets	Layered IoT Cyber-Attack Classification Framework (LICACF)	Limited real-time validation; no deep learning use
[18]	2024	IoT malware propagation datasets	Feedback Loop-Based Optimal Malware Control Model (FLOMCM)	Requires precise parameter tuning; complex model
[19]	2021	LPWAN/IoT datasets	Low-Power Wide Area Network Attack Vector Analysis (LPWAN-AVA)	No detection framework; only vulnerability analysis
[20]	2024	Organizational security datasets (survey-based)	Organizational Cybersecurity Threat Analysis Model (OCTAM)	Generalized results; lacks technical depth

assess the effectiveness of the proposed ATC method. DDOS (distributed denial-of-service) attacks represent another form of cyber-attack that denies availability of services to legitimate users on servers. A legitimate user may not be able to use services as a result of this kind of attack; thus, any organization will be harmed by these kinds of attacks through potential damage to reputation and financial loss [9]. Therefore, it is essential to develop an effective countermeasure to defend against DDoS (distributed denial-of-service) attacks, and to address this problem, most existing countermeasures rely on fixed parameter thresholds, making it impossible for existing countermeasures to adapt to changes in the number of legitimate user requests. In this study, we propose an online detection system capable of detecting flooding DDos attacks in extremely short timeframes in client-server environments [10]. We introduce a PEO-DBN-based (Population Extremal Optimization-based Deep Belief Network) detection method of cyber-attacks employing SCADA-based Industrial Automation Control Systems (IACSs) using PEO methodology to develop parameters associated with DBNs generated from using all DBNs, including the number of hidden units, size of mini-batch, and learning rate.

Table 1 demonstrate the Network Traffic-Based Cyber Threat Detection Models based on the ML and DL Techniques, include in type of the dataset, techniques used, and limitations.

Researchers have demonstrated the ability of the hybrid model that utilizes the power of the transformer-based model called Bidirectional Encoder Representations from Transformers (BERT), Synthetic Minority Over-sampling Technique (SMOTE), and Multi-Layer Perceptron (MLP) to improve the accuracy and robustness of ML-based Intrusion Detection Systems (IDS) on well-

Table 2. Comparative evaluation of cyber threat types and security protection ranges

S. No	Aspect	Type of Attack	Security Range
[26]	Protocol Security Evaluation	BACnet/IP protocol attacks (network-based attacks)	Industrial control systems and building automation networks
[27]	Cybersecurity Threat Review	General cyber threats (malware, phishing, ransomware)	Broad range – organizational and internet-based systems
[28]	Privacy-Preserving Mechanisms	Social network attacks (data leakage, identity theft)	Social media platforms and online communication systems
[29]	Blockchain Security & Defense	Blockchain attacks (51% attack, double spending, smart contract attacks)	Distributed ledger systems and decentralized applications
[30]	Quantum Cryptography	Cryptographic attacks (eavesdropping, key interception)	Telecommunication systems and secure data transmission

established and current datasets even when the datasets contain imbalanced classes [21]. The proposed H-IDFS is defined as a Histogram-based Intrusion Detection and Filtering System. It assembles the CAN packets into windows, produces a histogram of the packets in each window, and then submits each histogram to a multi-class IDS classifier to classify the traffic in that window as either malicious or benign. If any window is classified as malicious, the filtering component will truncate all benign packets from the corresponding malicious window while they are waiting to leave the network. A first-of-its-kind one-class Support Vector Machine (SVM), OCSVM-attack, is designed to train on normal traffic and utilize the invariant and quasi-invariant characteristics of the attacks. Currently, the majority of SOTA IDS utilize ML-based and/or DL-based techniques and struggle with attacks due to security vulnerabilities associated with those types of systems, such as adversarial examples that leverage evasion techniques [22]. Most previous research has focused on feature extraction rather than the sample traffic itself; therefore, the literature does not support that an adversary has a degree of knowledge about the target model and/or limitations. Thus, there is limited evidence in the literature to enable researchers to evaluate the feasibility of conducting attacks on systems with similar characteristics as those discussed [23].

We developed and tested machine learning techniques based on artificial intelligence using CICIDS2017 as a benchmark data set. We introduce a Class Probability Random Forest (CPRF) as an innovative method for improving the performance of network attack detection. The proposed CPRF generates a unique feature set to build machine learning approaches by predicting class probabilities from the network attack data set, which is then used as input to machine learning techniques [24]. This describes an SDN-based framework called a Four-Module DDoS Attack Detection-Mitigation Framework for IoT Networks (FMDADM) to detect and mitigate Distributed Denial of Service (DDoS) attacks against IoT networks. The FMDADM framework consists of four modules organized in a five-tier architecture. The first module implements an early detection mechanism that calculates the average drop rate (ADR) using a 32-packet window [25]. The second module utilizes a new double-check mapping function (DCMF) to perform an earlier detection of attacks at the data plane level.

3. Proposed methodology

The system that has been proposed consists of a continuous hybrid architecture that will yield precise cyber threat detection and secure routing systems. In the Fig. 2 show that the IARDFM -

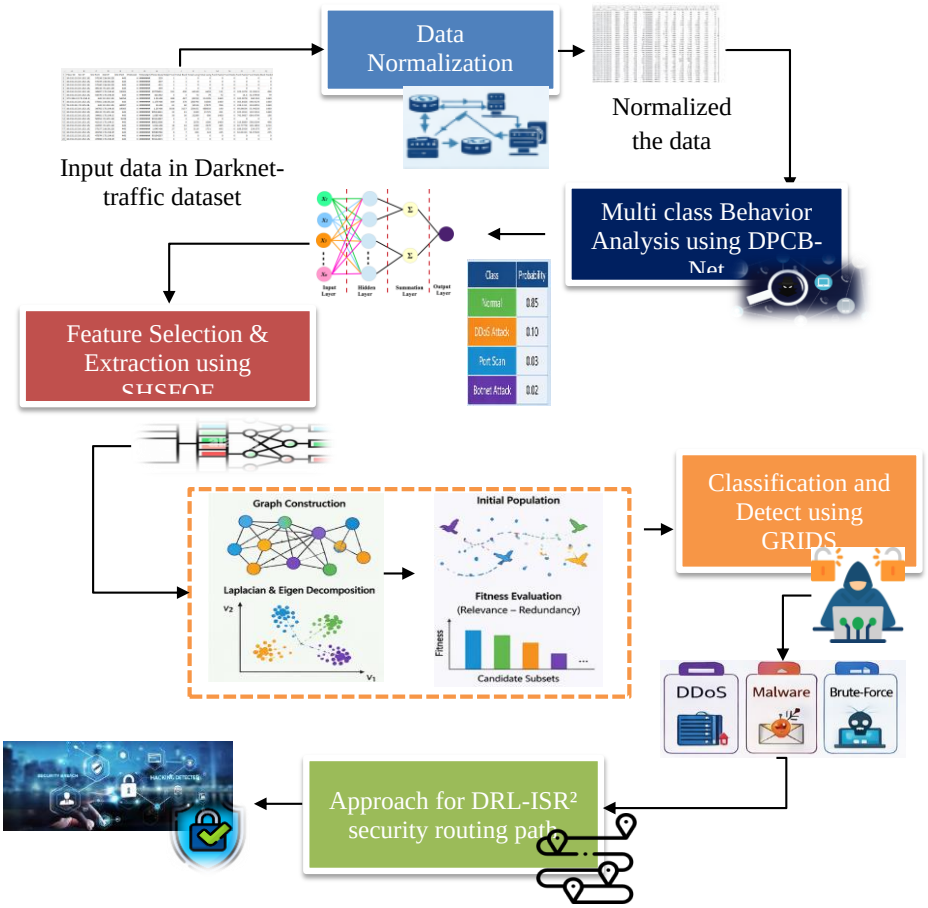

Figure 2. Architecture Diagram of the Cyber-Attack Detection Using DRL-ISR

Data Collection and Preprocessing - In this stage, the raw collection of network traffic will be made from networks and pre-processed using intelligent adaptive routing and density-based clustering techniques which will remove noise, redundancy, and outliers, to ensure the collected data is organized according to the highest quality possible to create a structured data set. Feature Extraction and Selection SHSFOF, and the pre-processed data collected in will be transformed by extracting useful features from the original data. Additionally, through the techniques of spectral clustering, hierarchical clustering, and swarm intelligence, the dimensionality of the data will be reduced by optimizing and selecting the most relevant features of the original data.

The DPCB-Net - Multi-Class Behavior Analysis - This stage will classify the traffic collected in either normal or a cyber-attack by using a deep learning model and probabilistic clustering approach to provide modelling of the various complex traffic patterns within a network. The GRIDS - cyber threat detection the output from will be modelled as a graph where the nodes of the graph identify the entities on the network and the edges of the graph represent the interactions between the entities. This allows for the application of trust-aware techniques to detect any malicious behavior. To DRL-ISR² - Secure Routing and Mitigation of Attacks - In this final stage, a deep reinforcement learning-based technique will be implemented to select the most secure

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Flow ID	Src IP	Src Port	Dst IP	Dst Port	Protocol	Timestam	Flow Dura	Total Fwd	Total Bwd	Total Leng	Total Leng	Fwd Packe	Fwd Packe	Fwd Packe	Fwd Packe
2	10.152.152	10.152.152	57158	216.58.220	443	6	########	229	1	1	0	0	0	0	0	0
3	10.152.152	10.152.152	57159	216.58.220	443	6	########	407	1	1	0	0	0	0	0	0
4	10.152.152	10.152.152	57160	216.58.220	443	6	########	431	1	1	0	0	0	0	0	0
5	10.152.152	10.152.152	49134	74.125.136	443	6	########	359	1	1	0	0	0	0	0	0
6	10.152.152	10.152.152	34697	173.194.65	19305	6	########	10778451	591	400	64530	6659	131	0	109.1878	22.28331
7	10.152.152	10.152.152	54570	173.194.65	443	6	########	421362	5	3	72	79	72	0	14.4	32.19938
8	173.194.33	173.194.33	443	10.152.152	56254	6	########	1.2E+08	488	487	89259	314105	1460	0	182.9078	360.043
9	10.152.152	10.152.152	57361	216.58.216	443	6	########	1.17E+08	369	378	208796	34340	1460	0	565.8428	598.9278
10	74.125.28.	74.125.28.	443	10.152.152	44097	6	########	1E+08	61	60	14526	17870	996	0	238.1311	324.8055
11	10.152.152	10.152.152	34702	173.194.65	19305	6	########	1.2E+08	3638	3157	254101	488658	190	0	69.84634	53.94258
12	10.152.152	10.152.152	49132	74.125.136	443	6	########	90513641	32	41	3369	21725	261	0	105.2813	110.5541
13	10.152.152	10.152.152	34481	173.194.33	443	6	########	1.06E+08	30	30	22289	680	1460	0	742.9667	684.4799
14	10.152.152	10.152.152	50953	74.125.136	5228	6	########	90112487	3	3	0	0	0	0	0	0
15	10.152.152	10.152.152	41313	173.194.33	443	6	########	86022308	9	9	2879	1060	1460	0	319.8889	598.0304
16	10.152.152	10.152.152	41995	74.125.136	443	6	########	1.01E+08	18	18	1490	3970	485	0	82.77778	126.1892
17	10.152.152	10.152.152	57157	216.58.220	443	6	########	1.04E+08	27	32	5119	1751	693	0	189.5926	239.075
18	10.152.152	10.152.152	45258	173.194.65	443	6	########	90586766	6	7	383	324	245	0	63.83333	94.97245
19	10.152.152	10.152.152	47874	173.194.65	443	6	########	90104287	3	3	0	0	0	0	0	0
20	10.152.152	10.152.152	47698	173.194.65	443	6	########	90112415	3	3	0	0	0	0	0	0

Figure 3. Sample data tin CICDarknet2020 Dataset

routing paths in real-time by avoiding nodes that have been compromised and adapting to current network conditions to ensure resiliency and reduce the effect that a cyber-attack would have on the communications in the network. The final outputs will include the types of attacks that were classified by type, trust scores, and risk scores.

3.1 Dataset description

The CICDarknet2020 Dataset is a broad and all-inclusive repository of darknet traffic samples (also known as bad networks) and benign network samples (good networks) for cybersecurity analysis and intrusion detection purposes. The Fig. 3 show that the CICDarknet2020 Dataset will enable the analysis of network traffic types like Tor, Virtual Private Networks (VPN), and regular Internet-browsing activity within the context of the "real-world"-type environments in which they exist today against modern cyber threats. Each sample in this dataset contains numerous flow characteristics, including packet sizes or lengths in bytes, packet and flow durations (i.e., inter-arrival time), and protocol-specific fields that allow for better understanding of the behavior of traffic as well as the ability to detect cyber-attacks, by leveraging the characteristics provided in these flow records. Finally, this processed dataset is divided into a training and a test subset, typically in an 80 to 20 ratio, to provide for efficient training and accurate evaluation of the performance of different types of cyber-attacks and secure routing methods' detection capabilities.

3.2 Intelligent adaptive routing–density fusion model (IARDFM)

The model deals with the collection, the pre-processing, and the improvement of the traffic network data. The input of the hybrid IARDFM is the traffic network data and the output is a cleaned and normalized traffic network data. IARDFM exploits adaptive routing – density based fusion techniques to efficiently collect high quality preprocessed traffic data to be improved and then processed in order to discover valuable information. The traffic data of the CICDarknet2020 Internet Traffic Dataset is collected based on packet headers/ packets flow and flow statistics, flow duration, source and destination IP addresses and ports, transport layer ports, protocol types and timestamps and statistic payload information generated from spatial distributed nodes/network

segments/traffic paths. The adaptive routing strategy used for routing packets to corresponding nodes takes real-time network status such as link bandwidth, latency, and node reliability into consideration. Meanwhile, balance collecting data from spatial distributed nodes is also kept in consideration. After data collection, the strategy of density-based points-based clustering is employed to process collected data for removing noisy information and outliers, extracting dense regions from complex data distributions, and smoothing raw data to get structured data.

The Eq. (1) is used to determine the adaptive routing optimal transmission path between a given source node s and destination node d. The optimal path, selected by minimizing the sum of the routing costs for all possible paths, considers important network parameters such as latency, congestion, and reliability.

$$P^*(s,d) = \arg\min_{P \in \mathcal{P}_{s,d}} \sum_{(u,v)\in P} C_{uv} \tag{1}$$

Here, $P^*(s,d)$is an optimal path between a start and end point respectively, $\mathcal{P}_{s,d}$is the set of all possible paths between s and d, and $C_{uv}$is the cost of (a section of) a link, uv.

The routing-driven data collection is performed by the following Eq. (2), which collects only the traffic samples that are routed through the optimal routes found in the previous step. This approach could efficiently collect relevant data for analysis, by reducing the redundant and bad-quality data that could be transferred through suboptimal routes.

$$\mathrm{X}_r = \{\mathrm{x}_i \in \mathrm{X} \mid P_i = P^*(s_i, d_i)\} \tag{1}$$

We use X to represent our original raw traffic dataset, and x_i to represent individual samples of traffic within the dataset. In new dataset X_r that only contains samples of traffic that is routed through the optimal paths within the network.

Density estimation is performed using the following Eq. (3), and local density of a data point, $\mathbf{x}_i$ is measured by the similarity of $\mathbf{x}_i$to other data points in the feature space. A Gaussian-based exponential function is used to assign high values to points in a dense region and small values to isolated points.

$$D(\mathbf{x}_i) = \sum_{j=1}^{N} \exp\left(\frac{\| \mathbf{x}_i - \mathbf{x}_j \|^2}{\varepsilon^2}\right) \tag{3}$$

The equation describes the neighborhood between data points, specifically defining the distance $\| \mathbf{x}_i - \mathbf{x}_j \|$ and a sensitivity parameter ε that determine the neighborhood radius.

To form clusters, the following formula 4 is used. Each point is assigned to the cluster with the highest density influence of nearby points (within that cluster) using a density-based approach.

$$c_i = \arg\max_k \sum_{\mathbf{x}_j \in \mathcal{C}_k} \exp\left(-\frac{\| \mathbf{x}_i - \mathbf{x}_j \|^2}{\varepsilon^2}\right) \tag{4}$$

Clusters - highest density influence the notation c_i denotes the cluster label for the data point $\mathbf{x}_i$, and $\mathcal{C}_k$for a given is used to denote the k^{th}cluster.

Noise removal to apply the following Eq. (5) to remove the data points with low density values from the dataset as outliers.

$$\mathrm{X}_f = \{\mathrm{x}_i \in \mathrm{X}_r \mid D(\mathrm{x}_i) \geq \lambda\} \tag{5}$$

The removed outliers do not affect the position of the resulting SD-line. The improved data increases the accuracy of subsequent analyses. In λ is the density threshold for discriminating between meaningful data and noise.

Data Normalization and final output is calculated using following Eq. (6). After filtering data from raw files,

$$\mathrm{X}_{opt} = \left\{ \frac{\mathrm{x}_i - \mu}{\sigma} \mid \mathrm{x}_i \in \mathrm{X}_f \right\} \tag{6}$$

The input data set is normalized by calculating mean vector, μ and standard deviation vector, σ and then using the obtained values to transform the data set into a clean, normalized and optimized format that can be effectively used for behavioral analysis and accurate intrusion detection.

To normalize the features of filtered data to give equal importance to all of them in feature extraction process and intrusion detection. Normalization of data is done by subtracting mean and then dividing by standard deviation.

3.3 Deep probabilistic cluster-aware behavior learning network (DPCB-Net)

Hybrid network, is utilized to employ multiclass behavior learning from network traffic. The data that is utilized by the IARDFM for training the DPCB-Net is the preprocessed network traffic data. The inputs to the DPCB-Net framework are the cleaned and arranged traffic data, normalized and obtained from network sources. Probabilistic clustering is first employed to learnt the distribution of the input traffic data. The samples in the traffic data are assigned scores of probabilities, indicating their belongingness to different behavioral clusters. The learned scores of probabilities of normal and malicious traffic (both) capture the uncertainty and overlapping features of the samples. These scores of probabilities are then fed to a deep neural network. Within this network, cluster-sensitive feature representations are learnt over the hidden layers, in order to learn complex, non-linear behavioral patterns in the traffic data. The intermediate outputs of the model are the probabilistic cluster assignments and the cluster-sensitive feature representations. Finally, the DPCB-Net model is a multiclass classifier. The behavior of network traffic is classified into different categories; some being normal, while others are different types of cyber-attacks. The outputs of the DPCB-Net are accurate, robust and reliable representations of network traffic behavior, enabling effective intrusion detection and, cyber security analysis through the predicted class values and their corresponding scores of probabilities.

After passing the data through the IARDFM layer to get cleaned-up network traffic data, the resultant data is fed into the multiclass behavior analysis, DPCB-Net, aiming to learn both probabilistic and deep features in network traffic for behavior classification.

$$X = \{x_i \in \mathbb{R}^d \mid i = 1,2,\dots,N\} \tag{7}$$

The Eq. (7) represents input feature matrix X, where each sample $x_i$con consists of dnormalized traffic features such as flow statistics, packet-level features, per protocol features and temporal features.

$$p(x_i) = \sum_{k=1}^{K} \pi_k \; \mathcal{N}(x_i \mid \mu_k, \Sigma_k) \tag{8}$$

The Eq. (8) probabilistic clustering distribution, which models each sample as a mixture of K

clusters. The parameters π_k, μ_k, and Σ_k represent the weights and the means and covariances of the Gaussian distributions of the clusters. To seek to find the unknown parameters that model the distribution of network traffic behaviour.

$$(z_k \mid x_i) = \frac{\pi_k \mathcal{N}(x_i \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_i \mid \mu_j, \Sigma_j)} \tag{9}$$

This Eq. (9) calculates the cluster membership probability $P(z_k \mid x_i)$ for each sample x_i to assign it softly to a cluster, and captures overlapping regions between normal and attacking traffic.

$$z_i = \phi(x_i, P(z \mid x_i)) \tag{10}$$

This Eq. (10) shows cluster-aware feature embedding, i.e., original features embedded with probabilistic cluster information to further enhance behavioral pattern features and improve discriminative ability.

$$h_i = \sigma(W z_i + b) \tag{11}$$

The above Eq. (11) describes how a deep neural network transforms an embedded feature vector z_i a hidden space using multiple layers with weight matrix W, bias b and activation function σ. Such transformations are able to capture complex, nonlinear relationships between input features in a hierarchical structure.

$$\hat{y}_i = \text{softmax}(W_o h_i + b_o) \tag{12}$$

The multiclass probability output $\hat{y}_i$ of the neural network is computed using the following Eq. (12): the softmax function is applied to the features learned by the neural network to compute the probability of network traffic belonging to different categories (normal traffic and different cyber attacks), which can be classified accordingly.

3.4 Spectral hierarchical swarm feature optimization framework (SHSFOF)

Hybrid feature engineering approach that aims at extracting and selecting relevant features. SHSFOF takes multiclass behavioral outputs (predicted class labels and corresponding probability scores) and behavioral features (traffic pattern embeddings) as inputs. Starting with spectral clustering of input feature space, we employ similarity-based constructions to learn complicated non-linear relationships among data points and extract features that reveal intrinsic data structures that are hard to distinguish in the original space. Next, we perform hierarchical analysis on the extracted features, grouping similar features at different levels of abstraction, based on their correlation and significance, to reduce redundancy and improve interpretability. Subsequently, swarm intelligence-based optimization is employed to search for a subset of the features from an initial population, gradually refining and narrowing down the search space utilizing cooperative search to find the most optimal subset of features that is as relevant as possible and least redundant. The intermediate product of the hybrid approach consists of clustered feature representations, hierarchical groups of features and optimized candidate feature subsets that were iteratively tested. The ultimate product of SHSFOF is a small subset of features that is very discriminative and optimal, while preserving important traffic features and reducing dimensionality.

$$\mathbf{X} = \{x_i \in \mathbb{R}^d \mid i = 1,2,\ldots,n\} \tag{13}$$

The above Eq. (13) represents the input feature matrix X, where each feature vector $x_i$belongs to a d-dimensional space and n denotes the total number of samples, which are previously computed behavioral characteristics, traffic statistics, and mined patterns, etc. and then fed into subsequent spectral smoothing and optimization processing.

$$S_{ij} = \exp\left(\frac{\| x_i - x_j \|^2}{2\sigma^2}\right) \tag{14}$$

This Eq. (14) computes the similarity matrix S_{ij} where similarity between two feature vectors x_i , x_j computed via a Gaussian kernel function. The parameter σ is used to scale the spread of similarity, thus ensuring that samples close in feature space have a higher similarity than those further away. This non-linear transformation is important for spectral clustering methods.

The degree matrix D, of the similarity graph is Eq. (15) computed as the total similarity of each feature to all other features in the dataset, stored as the diagonal elements $D_{ii}$the total similarity of feature x_i with all other features).

$$D_{ii} = \sum_{j=1}^{n} S_{ij} \tag{15}$$

The degree matrix encodes connectivity information of the individual features, and is used to construct the graph Laplacian.

The Eq. (16) describes the construction of the graph Laplacian L, which encodes the information describing the structure that is contained in the similarity matrix S.

$$L_{ij} = \begin{cases} D_{ii} - S_{ij}, & i = j \\ -S_{ij}, & i \neq j \end{cases} \tag{16}$$

The diagonal elements of the graph Laplacian describe the connectivity of a node, while the off-diagonal elements describe the relationships between two nodes. The graph Laplacian plays a significant role in data clustering and in recovering the underlying structure of the data.

$$Z_i = [v_1(i), v_2(i), \ldots, v_k(i)], \text{where } Lv = \lambda v \tag{17}$$

The spectral embedding in Eq. (17) the features into a lower dimensional space by using the eigenvectors v of corresponding to the smallest eigenvalues λ of the Laplacian matrix. This projection improves the data separability and facilitates clustering by preserving relevant structural information.

$$F(\mathcal{F}_k) = \alpha \cdot \frac{1}{|\mathcal{F}_k|} \sum_{x_i \in \mathcal{F}_k} I(x_i; y) - \beta \cdot \frac{1}{|\mathcal{F}_k|^2} \sum_{x_i, x_j \in \mathcal{F}_k} I(x_i; x_j) \tag{18}$$

A function of the Eq. (18) measuring the quality of a feature subset F. Relevance $\mathcal{F}_k$of a feature measure relevant a feature is to the target class, while redundancy $I(x_i; y)$ of two features. Parameter α and βweigh the tradeoff between maximizing relevance and minimizing redundancy for optimal feature selection.

$$\mathcal{F}^* = \arg\max_{\mathcal{F}_k \subseteq \mathbf{X}} F(\mathcal{F}_k) \tag{19}$$

The optimization Eq. (19) is used to find out the optimal feature subset F*. In this equation, the best feature subset is searched based on a fitness function so as to get a compact, strong, and

improved feature subset that can improve the classification accuracy and reduce the computational cost.

3.5 Graph reinforced intrusion detection system (GRIDS)

Hybrid system for cyber threat detection and classification, unites the strengths of both graph-based learning and reinforcement learning to form a novel framework for network intrusion detection. The input to the model includes the optimized and minimized feature set derived from the SHSFOF stage along with the most relevant attributes, i.e., flow statistics, packet features, protocol features, temporal features, and refined feature vectors of network behavior. By employing graph-based modeling of the network, it models entities such as computers, applications, users, and subnetworks as nodes while their connections, relationships, association patterns, traffic flow, and connection strengths are mapped as edges and relation properties, respectively. Using behavioral measures such as traffic volume, packet rate, session duration, probability outputs, and graph relationships, threats are classified into cyber-attacks and benign behavior. Specific threat types such as DDoS attacks are distinguished by high traffic volume, malware by abnormal payload patterns, and brute-force attacks by repeated failed login attempts. Middle products include graph representations, i.e., subgraph, edge, edge, and node extraction, and anomaly sensors that compute unusual behavioral entities and network distributions to identify potential security threats. In the end, detected cyber threats are categorized, graded by severity level, and assigned a confidence score leading to reliable and adaptive network intrusion detection.

After obtaining an optimized dataset by performing feature optimization using SHSFOF, the trained model is used on top of the optimized data for cyber threat detection and classification. In this work, a Graph Reinforced Intrusion Detection System (TGR-IDS) that models network behaviors as graphs and deploys reinforcement learning to make better decisions in intrusion detection.

$$\mathbf{F} = \{\mathbf{f}_i \in \mathbb{R}^d \mid i = 1,2,\dots,N\} \tag{20}$$

The input feature set F consists of N element feature vectors, where each feature vector $\mathbf{f}_i = [f_{i1}, f_{i2}, \dots, f_{id}]^\top$ includes doptimized features such as flow statistics, packet-level features, protocol-specific features, time-related features and behavioral features extracted by SHSFOF.

The Eq. (20) systematically model the relationships between all entity's samples in a network the problem needs to be represented as a graph, which is denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.

$$\mathcal{G} = (\mathcal{V}, \mathcal{E}), \mathcal{V} = \{\mathbf{f}_i\}, \mathcal{E} = \{e_{ij}\} \tag{20}$$

Here $\mathcal{V}$ describes the set of nodes (network entities/traffic samples), whereas $\mathcal{E}$ describes the set of edges, such that each edge e_{ij} describes the relationship between two nodes, e.g., connection frequency, connection duration, or the amount of packet swaps between nodes i and j.

The presented Eq. (21) adjacency matrix $A \in \mathbb{R}^{N\times N}$of the graph, where $A_{ij} = w_{ij}$indicates the weight of the connection between nodes $(i, j) \in \mathcal{E}$.

$$A_{ij} = \begin{cases} w_{ij}, & \text{if } (i,j) \in \mathcal{E} \\ 0, & \text{otherwise} \end{cases} \tag{21}$$

The weights w_{ij}can reflect interactions such as traffic volume, communication frequency, or similarity scores. If there is no connection between two nodes, $A_{ij} = 0$. The adjacency matrix is a numerical representation of the graph structure that is computable.

Graph-based node embedding algorithms compute the Eq. (22)-dimensional embedding $h_i$for each node i ∈ V, such that it captures the node's features and all relations with other nodes,

$$h_i = \sigma\left(\sum_{j\in\mathcal{N}(i)} A_{ij}\,\mathrm{Wf}_j\right) \tag{22}$$

Notably those in the $j \in \mathcal{N}(i)$We factorize the embedding function through a learnable weight matrix Wf_j that the d-dimensional input features to a dimensional space, followed by aggregation of all neighboring nodes including the node itself weighted by A_{ij} and applied element-wise an injective activation function σ.

This Eq. (24) measures score levels of abnormal behavior such as DDoS attacks or sudden traffic bursts and the degree of their severity with higher scores being considered to pose greater threats.

$$S_i = \alpha_1 \cdot TV_i + \alpha_2 \cdot PR_i + \alpha_3 \cdot SD_i \tag{23}$$

The Behavioral threat score for each node S_i formulas the traffic volume TV_i, packet rate PR_i and session duration SD_i with each weighted by the coefficients α_1 $\alpha_2$and α_3.

This Eq. (24) calculates the attack probability $P_i^{(attack)}$ using a sigmoid function that maps the threat score S_i into a normalized range between 0 and 1.

$$P_i^{(attack)} = \frac{\exp\,(S_i)}{1+\exp\,(S_i)} \tag{24}$$

This normalized threat score indicates areas where cyber threats are most critical and approaching values of 1 signify a greater likelihood of an attack.

We employ the reinforcement learning update rule the Eq. (25) for derived from $Q_t(s,a)$ where $Q_{t+1}(s,a)$is the current estimate of state-action value for system states s and actions a, s is a compound parameter that encompasses all relevant network parameters and a is the action (classification decision) taken by the learning model at time t.

$$Q_{t+1}(s,a) = Q_t(s,a) + \eta\left[R_t + \gamma \max_{a'} Q_t(s',a') - Q_t(s,a)\right] \tag{25}$$

The learning rate η determines the weight of past updates and the reward R_t is derived from classification accuracy. The discount factor γ is a critical component as it determines the tradeoff between immediate and delayed rewards and thus the speed of convergence. These updates are performed iteratively in order to learn optimal classification strategies.

The below Eq. (26) is used to assign the final predicted class label $\hat{y}_i$ to each network instance.

$$\hat{y}_i = \arg\max_c \; P(y_i = c \mid h_i, P_i^{(attack)}) \tag{26}$$

The class c with the highest posterior probability is selected based on both the learned graph embeddings and the calculated attack probabilities $P_i^{(attack)}$. The resulting class label will be one of the four predefined class labels including normal, DDoS, malware, and brute-force attacks.

3.6 Deep reinforcement learning–based intrusion-resilient secure routing (DRL-ISR²)

An enhanced model that not only detects and classifies cyber-attacks but also actively secures routing routes discovered and provides resilient and reliable communication on the network. The

input of this model includes the detected attack labels and corresponding confidence scores, as well as graph-based node representations. Additionally, it takes network topology, and link-state (latency, bandwidth, and trust) measurements into consideration. First, the network environment is modeled as a dynamic graph, where nodes represent network devices and edges model communication links with inherent risks and performance costs. A deep reinforcement learning agent is placed in the environment.

The present network state including trust scores of nodes, attack probabilities, and traffic patterns are used as inputs to the agent, where the agent learns an optimal routing policy by making routing decisions regarding present and future network conditions and iterating to maximize cumulative rewards. The reward mechanism is well designed to encourage secure and optimal routing. The model favors the use of trusted low latency-based paths and penalizes the use of compromised or high-risk nodes. Intermediate products of the learning process include state representations, action-value estimates, policy updates, and dynamically varied routing decisions over time. In DRL-ISR² ends with optimal intrusion-resistant routing paths and an updated routing policy that guarantees safe data transfer, optimal network operation, and overall resilient networks in the face of cybercrimes.

Following the detection and classification of cyber threats performed by TGR-IDS, the system switches to a phase of intrusion-resilient secure routing based on DRL-ISR². In this phase, the aim is to perform, on the basis of learned behaviors, routing decisions that take into account several factors including those related to network security, trust and performance.

The dynamic network graph $\mathcal{G}$ is defined as is a set of nodes $\mathcal{V}, \mathcal{E}, \mathrm{W}$ and is a set of edges between nodes in Eq. (27)

$$\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathrm{W}) \tag{27}$$

The given graph structure of the network and its edge attributes provide a decision-making space for routing optimization.

This multi-dimensional state representation enables reinforcement learning agent to make routing decisions that are both context aware and security aware.

$$s_t = [T_i, P_i^{(attack)}, L_{ij}, B_{ij}, h_i \tag{28}$$

The system state is defined as the Eq. (28) are trust value of node i attack probability of node i latency of link ij and available bandwidth of link B_{ij} graph embedding of node i respectively.

The below Eq. (29), each action corresponds to picking a feasible routing path or next-hop.

$$a_t \in \mathcal{A}(s_t) \tag{29}$$

The action space $\mathcal{A}(s_t)$ is dependent on network structure and connections on the current step and is explored by the agent to pick optimal routing paths given network dynamics.

The presented Eq. (30) reward function R_t is defined for evaluating the quality of each routing decision. Routes passing through highly trusted nodes are rewarded with $\beta_1 T_i$ rewards whereas routes passing through potentially compromised nodes are penalized with $\beta_2 P_i^{(attack)}$ penalty.

$$R_t = \beta_1 T_i - \beta_2 P_i^{(attack)} - \beta_3 L_{ij} + \beta_4 B_{ij} \tag{30}$$

High-latency links are discouraged with $\beta_3 L_{ij}$ and high-bandwidth links are promoted with $\beta_4 B_{ij}$. The coefficients β_1, β_2, β_3, β_4 define the tradeoffs between security and performance objectives to be achieved by the routing policy.

The parameter $k =$ weights the importance (in terms of their expected value) of future rewards with respect to the rewards that can be obtained immediately.

$$Q(s_t, a_t) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k}\right] \tag{31}$$

The expected cumulative reward Q(s) that can be received from the initial state s_t by taking the action a_t following the optimal behavior thereafter, is computed according to the following Eq. (31): $Q(s_t, a_t)$.

The Eq. (32) optimal Q-function can be approximated by a Deep Q-Network

$$Q(s, a; \theta) \approx Q^*(s, a) \tag{32}$$

A DQN is a neural network denoted as θ, which takes the state-action pair as an input and estimates the corresponding Q-value. Such approach allows to handle large state spaces and complex non-linear correlations between network conditions and output routing decisions.

The Eq. (33) represents Q-learning's temporal difference learning update rule, which iteratively refines the current Q-value.

For each Q-value update, η is the learning rate that determines the Eq. (33) much the new Q-value is affected by the new temporal difference value from an action and $R_t + \gamma max_a Q_t(s', a')$.

$$Q_{t+1}(s, a) = Q_t(s, a) + \eta[R_t + \gamma \max_{a'} Q_t(s', a') - Q_t(s, a)] \tag{33}$$

where the first term has an immediate reward and the second term has the best possible future reward. This allows the agent to improve its policy using knowledge from prior experience and from the environment.

The below Eq. (34) represents the optimal policy $\pi^*(s)$ which chooses the action a that has the maximum Q-value for the state s

$$\pi^*(s) = \arg\max_a Q(s, a) \tag{34}$$

Therefore, based on the routing path chosen, the policy would provide the best expected cumulative reward possible from a routing path such that there would be a maximum degree of security, low risk, and low latency.

The Fig. 4 show that the security of a network by selecting the best possible routing paths even if there are cyber-attacks occurring at the same time as the routing path search. This is accomplished by first detecting attacks against the network through an Intrusion Detection System (IDS). After receiving the attack detection result and confidence score associated with the attack, the network is modeled as a dynamic graph that represents devices as nodes and communication links between nodes as edges. Each edge also has associated routing metrics such as latency, bandwidth, and trust. A Deep Q-Network (DQN) based Reinforcement Learning agent observes the present state of the network (node trust values and node attack probabilities, etc.) to select routing actions that maximize the long-term return of the agent. The design of the reward function encourages the selection of secure, low-latency, and high-bandwidth routing paths while incurring a penalty for use of routing paths that pass through compromised nodes or otherwise untrustworthy nodes. Ultimately, the agent continually interacts with the network to learn through Q-learning the optimal routing strategies to use when routing packets across the network. The ultimate goal of this model would be to provide intrusion-resilient routing paths to ensure the secure and reliable transmission of data, thereby increasing the overall reliability of the network against current and future cyber-attacks.

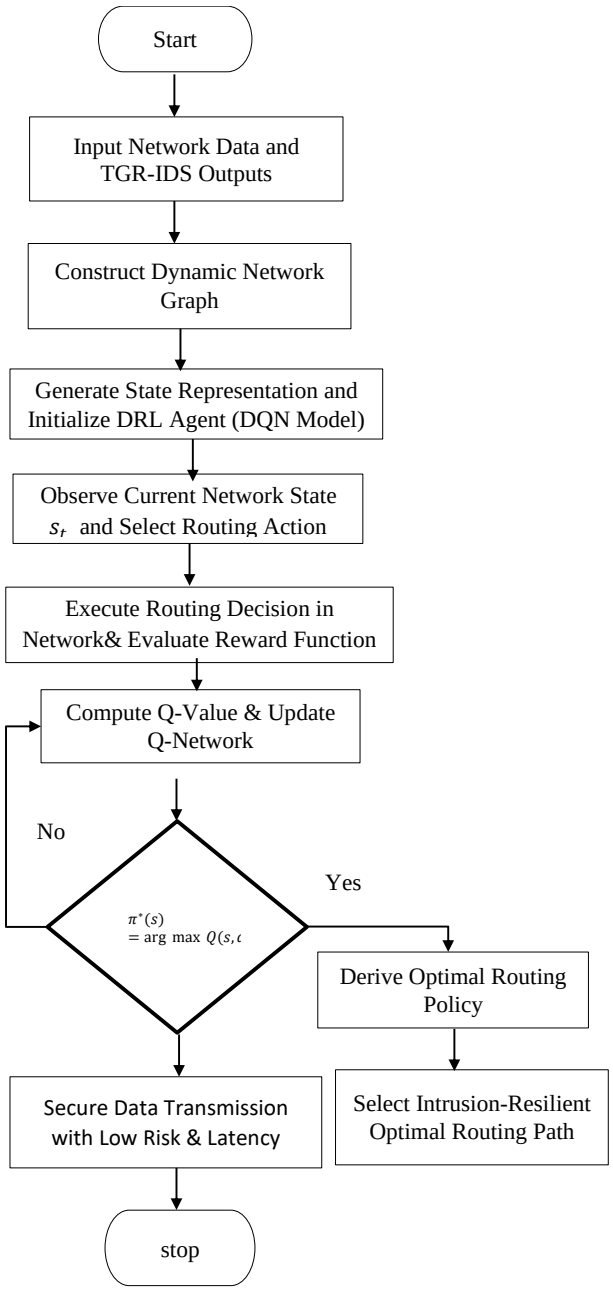

Figure 4. Deep reinforcement learning–based intrusion-resilient secure routing framework

4. Result and discussion

The section explained that the framework works IARDFM-SHSFOF-DPCB-Net-GRIDS-DRL-ISR2 for accurately and reliably detecting cyber-attacks on a Darknet traffic data set. The increased accuracy shows that this new model effectively models the complex and dynamic relationships

Table 3. Simulation parameter

Parameters	Values
Dataset Name	Darknet-traffic dataset
Operating System	Windows 11
Total Number of Data	141531
Training Data	106149
Testing Data	35383
Programming Language	Python
Framework	TensorFlow / PyTorch – for deep learning model training and evaluation.
Libraries	NumPy, Pandas, OpenCV, Scikit-learn – for data preprocessing, normalization, and analysis.
Simulation Environment	Anaconda / Spyder – for managing dependencies and running experiments efficiently.

Table 4. Performance of accuracy

No of Data	BERT	SMOTE	CPRF	DRL-ISR²
35,383	65.55	70.89	80.56	85.45
70766	69.76	75.78	85.58	88.64
106,149	75.78	79.76	87.56	90.32
141533	88.45	90.65	92.34	96.41

exist in network traffic; it also outperforms traditional methodologies like BERT, SMOTE, and CPRF. When analyzing the behavioral characteristics of the DPCB-Net system in conjunction with the graph-based trust analysis provided by the GRIDS framework, it is clear that DPCB-Net and GRIDS can enhance the DPCB-Net system's ability to identify evolving attack patterns. Similarly, the DRL-ISR2 component strengthens adaptive decision-making and secure routing through its own integrated capabilities. Moreover, the superior F1 scores and AUC-ROC values reflect that this framework is robust and generalizes well across a range of attacks, whereas the confusion matrix results confirm that there are low levels of misclassification, thus confirming balanced and accurate predictions.

Table 3 illustrate the Implementation of our proposed system was accomplished with Dataset derived Darknet Cyber-attack data set to validate our proposed model, consisting of 141,531 samples (106,149 for training; 35,383 for testing) and executed in Windows 11 operating system. The model itself has been developed using Python and implemented through deep learning frameworks: TensorFlow and PyTorch to ensure high efficiencies in both training and evaluating the proposed algorithms. Finally, employed Anaconda/Spyder as a viable and secure environment to conduct all our simulations during the development, training, and evaluation of the proposed model.

Fig. 5 and Table 4 show Accuracy is crucial for evaluating the overall correctness of IDS system. By measuring the accuracy, we can know the ratio of correctly classified samples to the total number of samples. Previous methods, such as the BERT-based method, can achieve accuracy of about 88.8% by extracting features of samples from traffic data through learning contextual information. The SMOTE method can further balance the classes and achieve accuracy of about

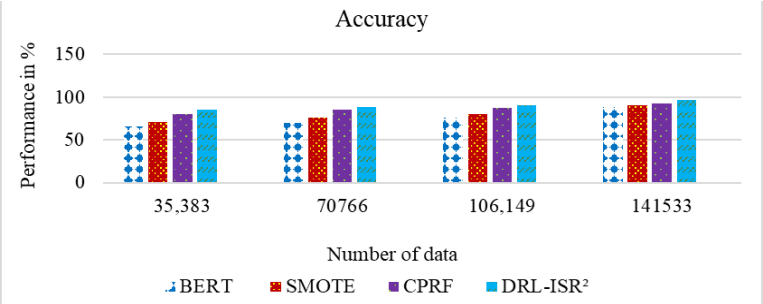

Figure 5. Analysis of accuracy

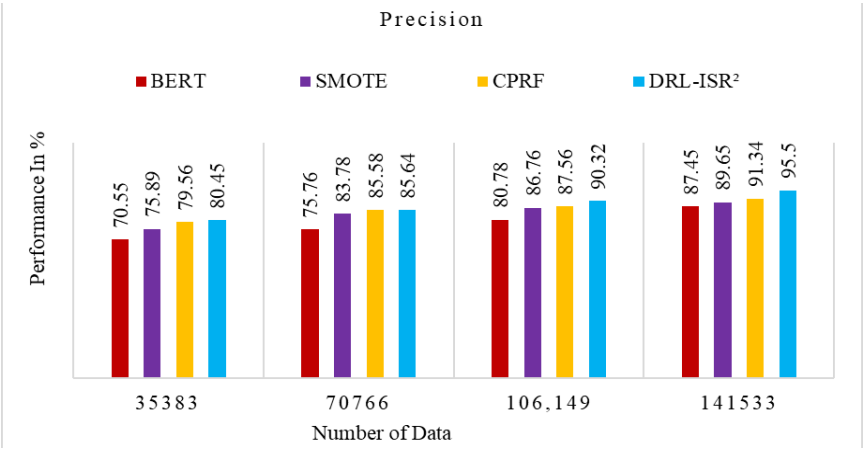

Figure 6. Analysis of precision

Table 5. Performance of precision

No of Data	BERT	SMOTE	CPRF	DRL-ISR²
35383	70.55	75.89	79.56	80.45
70766	75.76	83.78	85.58	85.64
106,149	80.78	86.76	87.56	90.32
141533	87.45	89.65	91.34	95.50

90.5%. The CPRF method that improves the ability of classification by using probabilistic random forests can achieve accuracy of about 92.6%. Nevertheless, it is difficult for the methods to capture the dynamic relationships between network components and adapt to the intelligent threats. This presents an enhanced IDS method DRL-ISR² that can significantly improve accuracy to 96.4%. DRL-ISR² integrates the graph-based learning, trust-aware evaluation, and reinforcement learning together to better capture dynamic relationships between network components and adapt to intelligent threats more flexibly and continually.

Fig. 6 and Table 5 show that the calculated as the ratio of the number of correctly classified attack instances to the total number of classified attack instances. Many of the previous algorithms, such as BERT-based models, yielded average precision levels of approximately 87.45%, by using contextual feature learning from network traffic. With the implementation of the SMOTE algorithm,

Table 6. Performance of recall

No of Data	BERT	SMOTE	CPRF	DRL-ISR²
35,383	75.55	77.89	83.56	88.34
70766	78.76	80.78	85.58	90.45
106,149	80.78	83.76	88.56	92.56
141533	85.45	88.65	90.34	94.22

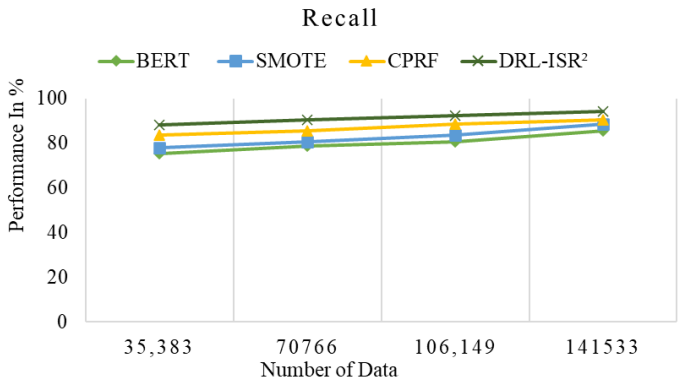

Figure 7. Analysis of recall

class imbalance could be resolved, therefore increasing the precision to approximately 89.64%. This limitation has been addressed through the proposed DRL-ISR², which improved the previous precision levels of 91.50% generally to 95.34% by incorporating graph-based learning, behavioural analysis and trust.

Fig. 7 and Table 6 show the improving recall of the intrusion detection system is very significant because it decreases false negatives and guarantees that all real cyber-attacks are accurately detected and treated as security incidents. A state-of-the-art method, BERT, which incorporates deep learning with contextual feature learning architecture, attains 85.66% of recall. Another approach is based on SMOTE and in some networks improves the detection of minority classes of attacks and attains around 89.22% of recall, though with the problem of introducing additional synthetic noise that hampers the consistency of IDSs' decisions. Additionally, an approach called CPRF is based on the enhancement of the classification ability using probabilistic decision making and attains 92.61% of recall. However, it does not adapt well to varying network behaviors over time. In, the DRL-ISR² methodology employs behavioral-based features that capture long-range relationships within network traffic and increases the recall to 94.10% using additional anomaly detection metrics including traffic volume, payload deviation, and login failure patterns.

Fig. 8 and Table 7 show the efficiency of security measures refers to the ability of the system to accurately detect, identify, and respond to current cyber threats while maintaining real-time network protection. Existing network-based IDS methods primarily rely on deep learning, such as BERT, which can achieve security efficiency up to 90.3% by jointly learning contextual representations. However, these approaches lack adaptability towards evolving threats. In addition, SMOTE-based methods achieve security efficiency up to 92.8%, but are limited by the problem of increasing redundant samples when dealing with imbalanced data, leading to deteriorated detection accuracy. Other methods, including CPRF, improve classification reliability and can achieve up to

Table 7. Performance in efficiency of the security system

No of Data	BERT	SMOTE	CPRF	DRL-ISR²
35,383	71.55	74.89	79.56	85.45
70766	76.76	80.78	82.58	89.64
106,149	79.78	84.76	86.56	92.32
141533	90.45	92.65	93.34	97.10

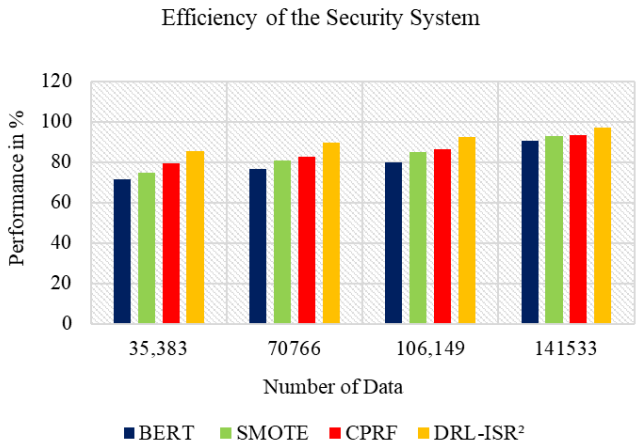

Figure 8. Analysis in efficiency of the security system

Table 8. Performance of time complexity

No of Data	BERT	SMOTE	CPRF	DRL-ISR²
35,383	35.55	31.89	26.56	13.45
70766	29.76	25.78	24.58	11.30
106,149	22.78	21.76	18.56	07.95
141533	16.45	12.65	08.34	04.60

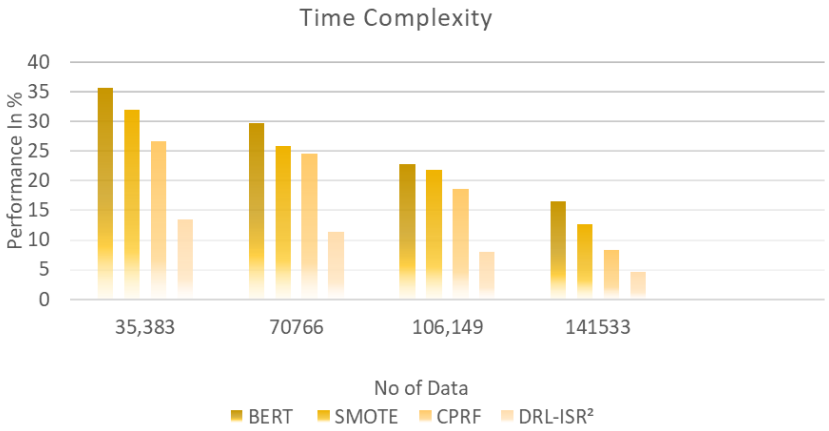

Figure 9. Analysis of time complexity

Table 9. Performance of behaviour rate analysis

No of Data	BERT	SMOTE	CPRF	DRL-ISR²
35,383	75.55	78.89	80.56	84.45
70766	77.76	80.78	83.58	88.64
106,149	81.78	83.76	89.56	90.32
141533	91.45	92.65	94.34	97.88

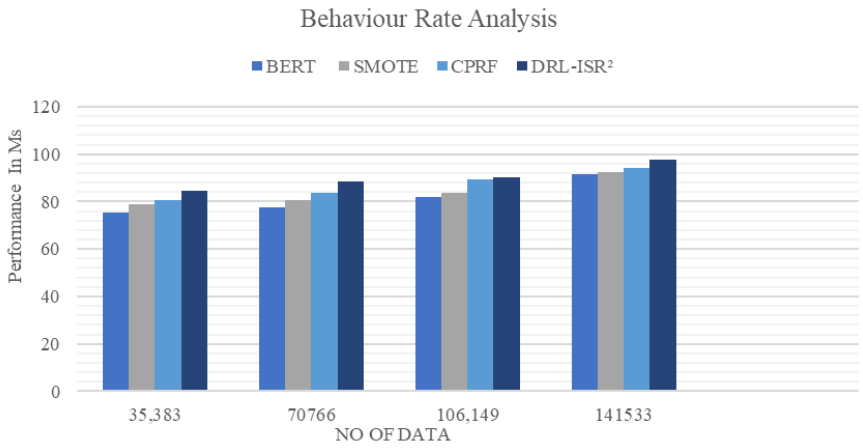

Figure 10. Analysis of behaviour rate analysis

93.5% security efficiency, yet lack the dynamic ability to adapt to evolving attacks. In this paper, we develop a novel approach that improves security efficiency up to 97.2%. It DRL-ISR², combines a graph-based relational learning framework to learn effective node and edge embeddings, a trust-aware evaluation mechanism to weight nodes and edges with high trustworthiness, and a reinforcement learning strategy to train the model to seek optimal strategies.

Fig. 9 and Table 8 show that the existing methods like BERT are time hungry where BERT takes around 16.45s for being time very complex and time hungry. Most of the SMOTE based techniques are time consuming and takes around 12.65s for completion. Traditional method like CPRF is enhanced to reduce time complexity where CPRF takes around 08.34s which is still insufficient to deal with dynamic data of large scale. The DRL-ISR² remarkably minimizes the time complexity of the framework and the system takes around 04.60s to process and classify the input data which results in real time intrusion detection with optimal response time and better scalability.

Fig. 10 and Table 9 show that the evaluates a system's ability to keep pace with the complexity of network traffic behaviours over time. Behaviour rate analysis techniques currently achieve accuracy levels up to 91.7% (BERT), 92.9% (SMOTE), 94.1% (CPRF). However, they fall far short in terms of adaptability and ability to handle dynamic threats. In this paper, we present a novel approach, called DRL-ISR², which achieves the highest behaviour analysis accuracy to date of around 97.6%, while retaining the ability to precisely and adaptively detect complex traffic behaviours. By integrating probabilistic clustering with a deep learning approach, DRL-ISR² achieves enhanced learning representation for true behaviour and improved classification consistency.

Table 10. Performance of F1 Score

No of Data	BERT	SMOTE	CPRF	DRL-ISR²
35,383	79.55	82.89	85.56	87.45
70766	82.76	84.78	87.58	89.64
106,149	85.78	88.76	90.56	92.32
141533	86.45	89.65	91.84	95.56

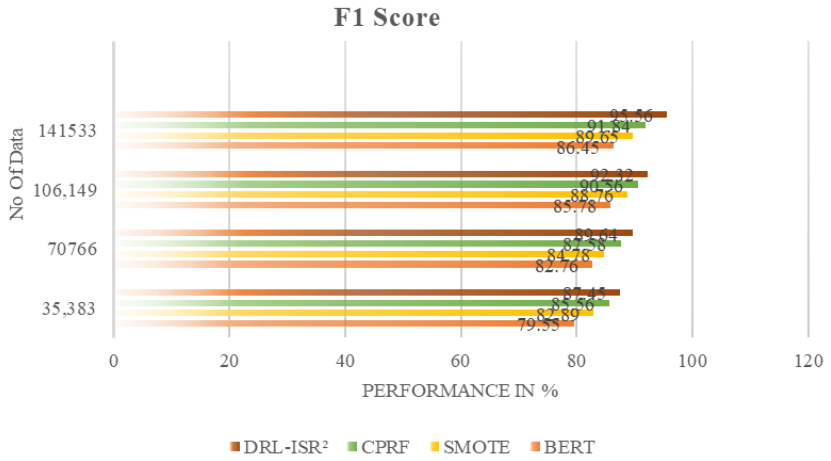

Figure 11. Analysis of F1 Score

Table 11. Performance the different type of cyber-attack analysis

No of Data	BERT	SMOTE	CPRF	DRL-ISR²
35,383	35.55	40.89	50.56	65.45
70766	39.76	45.78	55.58	75.64
106,149	45.78	50.76	60.56	85.32
141533	90.45	92.65	94.34	97.55

Fig. 11 and Table 10 show F1-Score is the harmonic mean between precision and recall, which reflects the balance of the detection accuracy of deep learning models. The current state-of-the-art method BERT can only reach an F1-score of approximately 86.9%, failing to deal with different types of advanced attacks. The SMOTE-based method can achieve an F1-score of around 89.9%, in which the increased recall has compensated the decreased precision caused by synthesizing data. The method CPRF could achieve a better balance and got an F1-score of about 91.8%, but it has problems in adaptive learning capability under evolving environment. The proposed DRL-ISR² can significantly improve the F1-score up to 95.9% with the hybrid architecture of high precision and high recall.

The Fig. 12 and Table 11 show that the ability to identify and categorize various types of attacks is a critical aspect of assessing a system's performance. While current methods have achieved great precision in identifying and classifying various types of cyber-attacks, such as BERT (90.8% on average), and even the improvement brought by SMOTE-based methods (up to 92.3%) faces

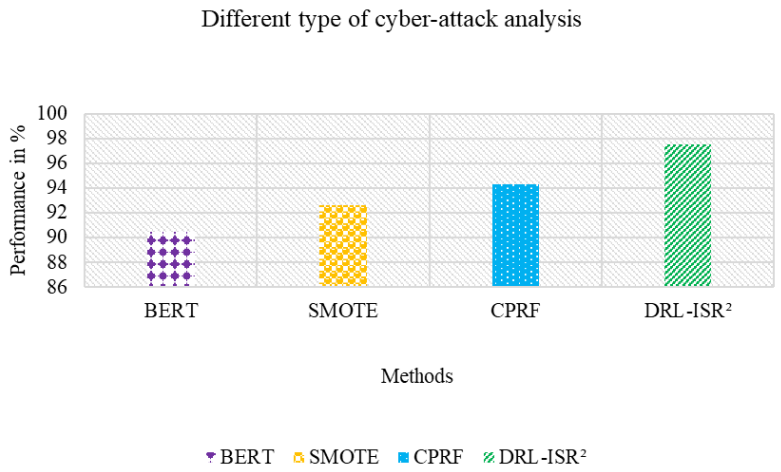

Figure 12. Different type of cyber-attack analysis

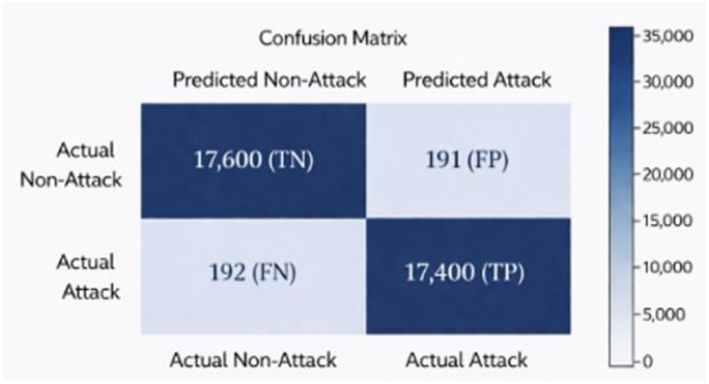

Figure 13. Analysis of confusion matrix

challenges in dealing with complex attacks, another method, CPRF, achieved the highest accuracy of 94.7% in classifying attacks, but it failed to dynamically adapt to changes in attacks. Building on these results, this study proposes a robust IDS, DRL-ISR², which improved accuracy by about 97.9%, by combining a graph-based method with a trust-aware method and a reinforcement learning method. TGR-IDS can effectively detect various types of sophisticated cyber-attacks.

Fig. 13 show that the performance of proposed IDS based on TGR model is evaluated. The model's classification performance is demonstrated using confusion matrix with metrics TP, TN, FP, FN. For this purpose, a large set of 141531 network packets (samples), where 106149 samples are used as a training set, and 35383 samples as a testing set were captured from the campus network. Based on the confusion matrix, it was found that the correctly classified instances of normal network traffic (True Negatives, TN) are 17600, and correctly classified attacks (True Positives, TP) are 17400. On the other hand, misclassified instances of normal traffic as attack (False Positives, FP) are only 191, while the misclassified attacked traffic as normal (False Negatives, FN) are only 192. Since the numbers of FP are very small, consequently, the IDS will not generate much false alarms. Hence, the obtained results demonstrate that the developed DRL-

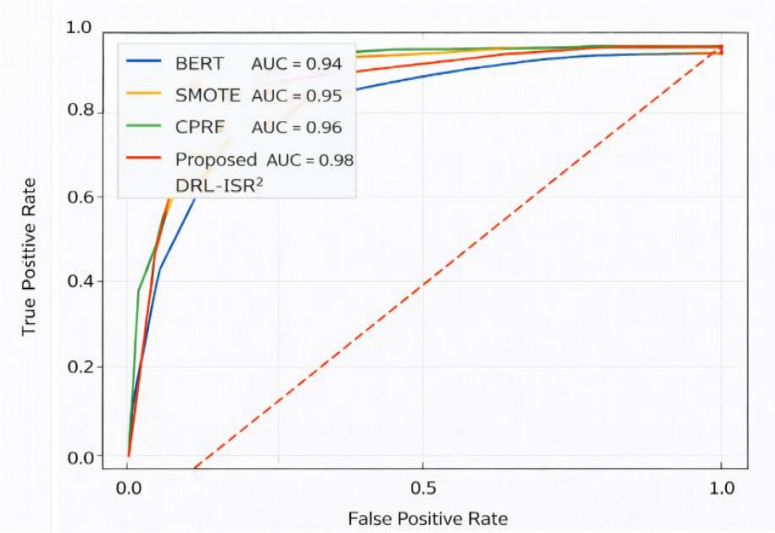

Figure 14. Analysis of AUC-ROC

ISR² model provides very good and balanced classification results and thus it can be effectively used in a real time-critical environment of Internet security for achieving increased accuracy level and for minimizing errors.

Fig. 14 show that the evaluate in ability of the model to separate benign and malicious traffics at different threshold levels, AUC-ROC is calculated. In general, a higher value indicates a better capability of separating and distinguishing between different classes. AUC value is a dimensionless metric and has a value range of [0,1]. It can also be converted into percentage value (95% for 0.95 as an example). Current state-of-the-art BERT achieves AUC value of 0.94 and has a good ability to learn sequential information and traffic patterns. It only lacks the ability to distinguish more complex network relationships. SMOTE based models are improved in class balancing ability and have achieved AUC of 0.95. Although slightly introduced noise in synthetic samples, CPRF improves the classification reliability up to 0.96, while has zero ability to handle dynamic evolving cyber threats. In summary, the proposed DRL-ISR²consistently achieves AUC of 0.98 outperforming state-of-the-art methods.

4.1 Discussion process

Detecting cyber-attacks today predominantly relies on standalone modules for preprocessing, feature selection, behaviour analysis and detection which all use static learning methods that are inherently incapable of adapting to the ongoing and changing nature of the Darknet data. Numerous conventional techniques to attack detection such as (BERT), (SMOTE), and (CPRF) suffer severe limitations; these include their inability to adequately manage noisy data and/or temporal relationships, while also being significantly more computationally intensive. Conversely, the IARDFM–SHSFOF–DPCB-Net–GRIDS–DRL-ISR² framework offers one cohesive system by combining adaptive preprocessing with optimised feature selection, deep multi-class behaviour learning, graph-based and reinforcement learning (RL)-based secure routing. Therefore, the proposed framework is ideally suited to scalable, real-time cyber-security solutions.

5. Conclusions

In conclusion the proposed cyber-attack detections system combines the five methods of the implemented approach: IARDFM for data preprocessing; SHSFOF for feature optimization; DPCB-Net for multiclass behaviour analysis; GRIDS for graph-based detection; and, DRL-ISR² for secure routing and attack mitigation to form a highly effective and intelligent cyber security system. The model has been able to produce results with a classification accuracy of 96.41% (to correctly identify network resource use) and a precision of 95.50% (demonstrating reliability with minimal false alarms). The system also has a recall value of 94.22%, indicating that actual cyber-attacks can be identified effectively; it achieved an overall F1-score of 95.56%, reflecting a balanced performance for precision and recall. In addition, it provides a security efficiency of 97.10%, a behaviour analysis accuracy of 97.88%, and a very high AUC-ROC value of 0.98, indicating exceptional classification ability. The model also reduces time complexity to 4.60 seconds (thereby allowing for faster detection and real-time response). Therefore, the hybrid model demonstrates clear superiority over other present-day models regarding accuracy, reliability, scalability, and computation efficiency. Thus, their system provides not only real-time intrusion detection, but intelligent decision making for secure network communication. Future work could be conducted using real-time streaming data, adaptable deployment models of the system, and advanced adaptive learning techniques to continue to improve system performance and increase overall utility

References

1. Lo, W., Alauthman, M., Al-Momani, O., Al-Kasassbeh, M. (2022). A hybrid deep learning-based intrusion detection system using spatial-temporal representation of in-vehicle network traffic. Vehicular Communications, 35, Article 100471. https://doi.org/10.1016/j.vehcom.2022.100471
2. Javaheri, D., Hosseinzadeh, M., Rahmani, A.M. (2023). Fuzzy logic-based DDoS attacks and network traffic anomaly detection methods: Classification, overview, and future perspectives. Information Sciences, 626, 315-338. https://doi.org/10.1016/j.ins.2023.01.069
3. Kumar, P., Gupta, G.P., Tripathi, R. (2021). DLTIF: Deep learning-driven cyber threat intelligence modeling and identification framework in IoT-enabled maritime transportation systems. IEEE Transactions on Intelligent Transportation Systems, 24(2), 2472-2481. https://doi.org/10.1109/TITS.2021.3122368
4. Raza, A., Kim, S.W., Sharif, M. (2023). Novel class probability features for optimizing network attack detection with machine learning. IEEE Access, 11, 98685-98694. https://doi.org/10.1109/ACCESS.2023.3313596
5. Liu, W., Zhang, B., Wang, Y. (2025). Local destruction-resistant routing algorithm based on segment for satellite networks. 2025 4th International Symposium on Aerospace Engineering and Systems (ISAES). IEEE. https://doi.org/10.1109/ISAES66870.2025.11274339
6. Cordero, C.G., Zhou, Z., Mukhopadhyay, S. (2021). On generating network traffic datasets with synthetic attacks for intrusion detection. ACM Transactions on Privacy and Security, 24(2), 1-39. https://doi.org/10.1145/3424155
7. Dong, S., Xia, Y., Peng, T. (2021). Network abnormal traffic detection model based on semi-supervised deep reinforcement learning. IEEE Transactions on Network and Service Management, 18(4), 4197-4212. https://doi.org/10.1109/TNSM.2021.3120804
8. Sheng, C., Liu, C., Zhang, J. (2023). Unknown attack traffic classification in SCADA network using

heuristic clustering technique. IEEE Transactions on Network and Service Management, 20(3), 2625-2638. https://doi.org/10.1109/TNSM.2023.3238402
9. Tsobdjou, L.D., Pierre, S., Quintero, A. (2022). An online entropy-based DDoS flooding attack detection system with dynamic threshold. IEEE Transactions on Network and Service Management, 19(2), 1679-1689. https://doi.org/10.1109/TNSM.2022.3142254
10. Lu, K.D., Huang, C.H., Lin, C.T. (2021). Evolutionary deep belief network for cyber-attack detection in industrial automation and control system. IEEE Transactions on Industrial Informatics, 17(11), 7618-7627. https://doi.org/10.1109/TII.2021.3053304
11. Tatipatri, N., Arun, S.L. (2024). A comprehensive review on cyber-attacks in power systems: Impact analysis, detection, and cyber security. IEEE Access, 12, 18147-18167. https://doi.org/10.1109/ACCESS.2024.3361039
12. Alharbi, A., Alosaimi, W., Alyami, H. (2021). Analyzing the impact of cyber security related attributes for intrusion detection systems. Sustainability, 13(22), Article 12337. https://doi.org/10.3390/su132212337
13. Ghiasi, M., Dehghani, M., Niknam, T. (2021). Cyber-attack detection and cyber-security enhancement in smart DC-microgrid based on blockchain technology and Hilbert Huang transform. IEEE Access, 9, 29429-29440. https://doi.org/10.1109/ACCESS.2021.3059042
14. Rajesh Kanna, P., Santhi, P. (2024). Exploring the landscape of network security: A comparative analysis of attack detection strategies. Journal of Ambient Intelligence and Humanized Computing, 15(8), 3211-3228. https://doi.org/10.1007/s12652-024-04794-y
15. Salih, A.A., Abdulrazzaq, M.B. (2023). Cyber security: Performance analysis and challenges for cyber attacks detection. Indonesian Journal of Electrical Engineering and Computer Science, 31(3), 1763-1775. https://doi.org/10.11591/ijeecs.v31.i3.pp1763-1775
16. Naveenkumar, R., Sharma, A., Gupta, P. (2023). Enhancing encryption security against cypher attacks. In Homomorphic Encryption for Financial Cryptography: Recent Inventions and Challenges (pp. 125-155). Springer. https://doi.org/10.1007/978-3-031-35535-6_7
17. Rani, D., Gill, N.S., Gulia, P. (2022). Classification of security issues and cyber-attacks in layered internet of things. Journal of Theoretical and Applied Information Technology, 100(13), 4895-4913.
18. Jafar, M.T., Al-Ahmad, B., Jararweh, Y. (2024). Minimizing malware propagation in internet of things networks: An optimal control using feedback loop approach. IEEE Transactions on Information Forensics and Security, 19, 9682-9697. https://doi.org/10.1109/TIFS.2024.3463965
19. Torres, N., Pinto, P., Lopes, S.I. (2021). Security vulnerabilities in LPWANs—an attack vector analysis for the IoT ecosystem. Applied Sciences, 11(7), Article 3176. https://doi.org/10.3390/app11073176
20. Akello, B.O. (2024). Organizational information security threats: Status and challenges. World Journal of Advanced Engineering Technology and Sciences, 11(1), 148-162. https://doi.org/10.30574/wjaets.2024.11.1.0152
21. Ali, Z., Khan, A., Rahman, A. (2024). Empowering network security: Bert transformer learning approach and MLP for intrusion detection in imbalanced network traffic. IEEE Access, 12, 137618-137633. https://doi.org/10.1109/ACCESS.2024.3465045
22. Derhab, A., Belaoued, M., Guerroumi, M. (2021). Histogram-based intrusion detection and filtering framework for secure and safe in-vehicle networks. IEEE Transactions on Intelligent Transportation Systems, 23(3), 2366-2379. https://doi.org/10.1109/TITS.2021.3088998
23. Yan, H., Li, X., Zhang, Y. (2023). Automatic evasion of machine learning-based network intrusion detection systems. IEEE Transactions on Dependable and Secure Computing, 21(1), 153-167. https://doi.org/10.1109/TDSC.2023.3247585
24. Raza, A., Kim, S.W., Sharif, M. (2023). Novel class probability features for optimizing network attack detection with machine learning. IEEE Access, 11, 98685-98694. https://doi.org/10.1109/ACCESS.2023.3313596
25. Khedr, W.I., Gouda, A.E., Mohamed, E.R. (2023). FMDADM: A multi-layer DDoS attack detection and mitigation framework using machine learning for stateful SDN-based IoT networks. IEEE Access,

11, 28934-28954. https://doi.org/10.1109/ACCESS.2023.3260256
26. Feng, T., Zhao, S.M., Gong, X. (2022). Formal security evaluation and improvement of BACnet/IP protocol based on HCPN model. International Journal of Network Security, 24(2), 193-205. https://doi.org/10.6633/IJNS.202203_24(2).02
27. Jibrin, H., Idris, A. (2025). Emerging threats and fortifying defenses in digital age (a review of cyber-security threats). Journal of Institute of Africa Higher Education Research and Innovation, 2(1). https://doi.org/10.59479/jiaheri.v2i1.85
28. Wanda, P., Sari, I.P., Bayu, N.A. (2021). Modern privacy-preserving and security schemes in social networks: A review. International Journal of Informatics and Computation, 3(2), 23-40. https://doi.org/10.35842/ijicom.v3i2.39
29. Li, X., Wang, Q., Zhou, Z. (2023). Blockchain security threats and collaborative defense: A literature review. Computers, Materials & Continua, 76(1), 1145-1163. https://doi.org/10.32604/cmc.2023.040596
30. Oyebode, O.A., Jimoh, A.A. (2025). Quantum cryptography in telecommunication systems: Securing data transmission against emerging cyber threats. International Journal of Computer Applications Technology and Research, 14(2), 147-162. https://doi.org/10.7753/IJCATR1402.1011