Journal of Neonatal Surgery
ISSN(Online): 2226-0439
Vol. 14, Issue 18s (2025)
https://www.jneonatalsurg.com

OPEN ACCESS

Enhancing Network Traffic Classification Using Multi-Tier Reinforced Salp Optimization Algorithm and Deep Learning Models

S.Padmavathy[1], Dr. R.Kannan[2]

[1]Ph.D Research Scholar, Department of Computer Science, SRMV College of Arts & Science, Coimbatore-49. amulusugavanam@gmail.com
[2]Associate Professor, Department of Computer Science, SRMV College of Arts & Science.Coimbatore-49.

Cite this paper as: S.Padmavathy, Dr. R.Kannanet.al (2025) Enhancing Network Traffic Classification Using Multi-Tier Reinforced Salp Optimization Algorithm and Deep Learning Models. *Journal of Neonatal Surgery*, 14 (18s), 255-267.

ABSTRACT
Accurate and efficient detection of cyber threats is essential for effective network traffic analysis. This research introduces a Multi-Tier Reinforced Salp Optimization Algorithm (MTR-SOA) for feature selection, integrated with deep learning (DL) models to enhance network traffic classification. The proposed method is evaluated against traditional optimization techniques, including Particle Swarm Optimization (PSO), Grey Wolf Optimizer (GWO), and Genetic Algorithm (GA), across various DL models such as FNN, CNN, DELM, and LSTM. Experimental results reveal that MTR-SOA consistently delivers superior performance, with the LSTM model achieving the highest accuracy of 98.2% and demonstrating strong classification across all traffic categories. Furthermore, MTR-SOA reduces computational time by up to 30%, making it suitable for real-time network traffic analysis. Class-wise evaluation on the HIKARI-2021 dataset highlights its effectiveness in identifying complex cyber-attacks like XMRIGCC CryptoMiner and Probing. These findings confirm that integrating MTR-SOA with DL models enhances network traffic analysis by improving both accuracy and computational efficiency, offering a robust solution for detecting and classifying diverse traffic patterns.

Keywords: Multi-Tier Reinforced Salp Optimization Algorithm, Deep Learning Models, Network Traffic Classification, Feature Selection, Cyber Threat Detection.

1. INTRODUCTION

With the rapid expansion of digital networks and increasing volumes of data exchange, effective network traffic analysis has become essential for ensuring cyber security and maintaining stable network performance. Analysing network traffic enables the identification of abnormal patterns, detection of potential cyber threats, and optimization of network resources. However, the growing complexity and diversity of network traffic present significant challenges in accurately classifying and detecting malicious activities, especially when dealing with high-dimensional data and overlapping traffic patterns [1].

Recent advancements in DL have shown great potential for enhancing network traffic analysis due to their ability to learn complex patterns and generalize across diverse datasets. However, the effectiveness of DL models heavily depends on the quality of the selected features. Redundant or irrelevant features can increase computational costs and degrade model performance. To address this issue, feature selection techniques are employed to identify the most relevant features, thereby improving classification accuracy and reducing computational complexity [2].

This research introduces a novel MTR-SOA for feature selection, designed to enhance the efficiency and accuracy of deep learning models for network traffic analysis. The proposed MTR-SOA approach is compared against traditional optimization algorithms, including PSO, GWO, and GA, across various DL models such as FNN, CNN, DELM, and LSTM.

Experimental results demonstrate that the MTR-SOA significantly improves classification performance, with the LSTM model achieving the highest accuracy of 98.2% and effectively identifying both benign and malicious traffic patterns. Additionally, the proposed method reduces computational time by up to 30%, making it suitable for real-time applications. The class-wise performance evaluation using the HIKARI-2021 dataset highlights the model's capability to detect complex cyber threats, including XMRIGCC CryptoMiner and Probing attacks.

This research underscores the importance of integrating advanced feature selection techniques with deep learning models to enhance network traffic analysis. By improving detection accuracy and computational efficiency, the proposed MTR-SOA-based approach offers a robust solution for effectively managing and analyzing network traffic in evolving cybersecurity environments.

2. LITERATURE SURVEY

Alabduallah et al. (2024) introduced a Hybrid Salp Swarm Algorithm with Deep Learning (HSSADL-CAC) for the classification of cyber-attacks, tackling the issue of class-imbalanced data. The approach incorporates data normalization, ADASYN for addressing class imbalance, and a feature selection mechanism based on HSSA. The method utilizes a Deep Extreme Learning Machine (DELM) for cyber-attack detection, incorporating hyper parameter optimization through the Beluga Whale Optimization (BWO) model. Performance investigation on benchmark datasets revealed the HSSADL-CAC's enhanced accuracy and resilience in identifying minority class cyber-attacks, underscoring its applicability for real-time cyber security solutions [3].

Jullian et al. (2023) introduced a distributed DL system for identifying cyber-attacks in Internet of Things networks (IoT) tackling vulnerabilities associated with the extensive deployment of smart devices. The framework assesses two deep learning models: a Feed Forward Neural Network (FNN) and a Long Short-Term Memory (LSTM) network, utilizing the NSL-KDD and BoT-IoT datasets. The models proficiently categorize various assault kinds, attaining a detection accuracy of up to 99.95% across diverse configurations. This study emphasizes the effectiveness of a distributed method in improving IoT security by addressing various risks within a cohesive protection framework [4].

C. C. et al. (2022) introduced a robust DL system, ScaleMalNet, aimed at improving cyber security via multifarious detection and classification methodologies. The architecture includes domain generation algorithm (DGA) detection, a hybrid IDS for monitoring Ethernet LAN operations, and a consolidated model for detecting spam and phishing across email, social media, and URLs. It also presents DL-based techniques for classifying secure shell (SSH) traffic and distinguishing between malicious and benign network traffic. ScaleMalNet utilizes a bifurcated malware analysis methodology, using static and dynamic analysis to categorize malware and classify it into families. A hybrid deep learning framework for Android ransom ware detection has been developed, surpassing conventional machine learning methods. The research investigates DNS-based botnet identification and network intrusion detection specifically designed for IoT and smart city contexts, showcasing the adaptability and scalability of deep learning in various cyber security applications [5].

Aljebreen et al. (2023) presented the Modified Equilibrium Optimization Algorithm with DL-based DDoS Attack Classification (MEOADL-ADC) to improve security in 5G networks. The MEOADL-ADC method utilizes a three-phase approach comprising feature selection, classification, and hyper parameter optimization. The feature selection phase employs the MEOA methodology, whilst the LSTM model is applied for DDoS attack classification. The hyper parameter optimization of the LSTM model is conducted using the Tunicate Swarm Algorithm (TSA). Experimental results on a benchmark dataset indicated that the MEOADL-ADC technology surpassed existing methods, attaining an accuracy of 97.60%, hence demonstrating its efficacy in DDoS attack detection within 5G networks [6].

Abu Al-Haija and Zein-Sabatto (2020) introduced an innovative deep-learning detection and classification system for cyber-attacks in IoT communication networks, termed IoT-IDCS-CNN, which utilizes CNN for enhanced cyber security performance. The system employs CUDA-based Nvidia GPUs and Intel I9-core CPUs for effective parallel computing. The system has three subsystems: feature engineering, feature learning, and traffic categorization, all of which have been created, validated, and integrated. The system was assessed utilizing the NSL-KDD dataset and exhibited outstanding performance, attaining over 99.3% accuracy in binary-class classification and 98.2% in multi-class classification. The outcomes exceeded those of most contemporary ML-based intrusion detection systems (IDCS), establishing IoT-IDCS-CNN as a highly efficient approach for safeguarding IoT networks [7].

Fernandes and Lopes (2022) examined the application of the HIKARI-2021 dataset, sourced from authentic laboratory network traffic data, to enhance network performance. Feature selection approaches were employed to discern pertinent information, diminishing the dataset from 83 to 22 features while preserving a high classification accuracy of 99%. The research indicated that the dataset is appropriate for ML algorithms, with over 80% accuracy with balanced samples across multiple machine learning techniques. The study underscores the HIKARI-2021 dataset's capacity to improve IDS efficacy, providing a pragmatic approach for expedited and more effective cyber-attack identification [8].

Judith et al. (2023) investigated the application of deep learning for identifying cyber security vulnerabilities in the Internet of Medical Things (IoMT), with a special emphasis on man-in-the-middle attacks in medical device communication networks. The research applied principal component analysis (PCA) for feature reduction and implemented a multi-layer perceptron (MLP) for attack categorization, employing real-time data from the St. Louis Enhanced Healthcare Monitoring System (WUSTL-EHMS). The results indicated that the MLP classifier surpassed previous models, attaining an accuracy of 96.39% while concurrently decreasing time complexity, underscoring its efficacy in tackling the dynamic characteristics of cyber threats in IoMT systems [9].

Taşcı (2024) proposed an optimized one-dimensional convolutional neural network (1D CNN) model to improve IoT security through the effective classification of IoT-related assaults and malware. The model comprises input, convolutional, self-attention, and output layers, utilizing GELU activation, dropout, and normalization methods to mitigate over fitting and enhance performance. The model, assessed using the CIC IoT 2023, CIC-MalMem-2022, and CIC-IDS2017 datasets, attained exceptional results, with accuracy rates above 99%, and exhibited elevated precision, recall, and F1-scores. The research highlights the efficacy of deep learning in safeguarding IoT ecosystems, providing a low-complexity solution appropriate for real-time and resource-limited applications. Future endeavours will concentrate on augmenting dataset evaluation and integrating adaptive learning to enhance resilience [10].

Singh et al. (2023) presented RANSOMNET+, a hybrid model that integrates CNNs with pre-trained transformers to address the classification of ransom ware attacks on cloud-encrypted data. The model surpassed conventional architectures like ResNet 50 and VGG 16, attaining remarkable metrics, including a precision of 99.5%, recall of 98.5%, and an F1 score of 97.64%. RANSOMNET+ exhibited elevated training and testing accuracy, accompanied by minimal loss values during the procedure. The model's interpretability was improved by feature distribution analysis, outlier detection, and feature importance evaluation, rendering it a potent instrument for safeguarding cloud data. The study introduces RANSOMNET+ as an effective solution for ransom ware detection, equipping cyber security experts with a formidable defence against assaults on cloud-based systems [11].

Fernandes et al. (2023) examined the influence of identifiable characteristics on machine learning classification techniques utilizing the HIKARI-2021 dataset. Their research underscores the significance of these attributes for model efficacy, demonstrating a notable 20% decline in accuracy upon the removal of identifying traits. The study highlights the essential function of these properties in improving the efficacy of network IDS. This analysis enhances the continuous advancement of datasets and algorithms in the sector, emphasizing the importance of feature relevance in the creation of effective cyber security solutions [12].

Noori et al. (2023) tackle the issue of feature drift in IDS by introducing an enhanced Genetic Programming (GP)-based ensemble classifier, termed the Dynamic Feature Aware GP Ensemble (DFA-GPE). The research presents an improved iteration of Variable Length Multi-Objective Particle Swarm Optimization (VLMO-PSO) for the effective management of feature drift. DFA-GPE optimizes feature selection through tactics such as intelligent population initialization and innovative exemplar selection, achieving a balance between accuracy and memory efficiency. DFA-GPE exhibited remarkable performance on the HIKARI 2021 and TON-IoT 2020 datasets, achieving accuracies of 99.09% and 92.64%, respectively, surpassing current methodologies and offering a viable solution for dynamic feature selection in online intrusion detection systems [13].

Table.1. Literature Review

Author(s)	Proposed Method	Datasets	Key Results	Research Gap
Alabduallah et al. (2024)	HSSADL-CAC (Hybrid Salp Swarm Algorithm with Deep Learning)	Benchmark datasets	Enhanced accuracy and resilience in minority class detection	Limited validation for real-time deployment scenarios
Jullian et al. (2023)	Distributed DL System for IoT	NSL-KDD, BoT-IoT	Up to 99.95% detection accuracy	Scalability for large-scale IoT networks remains unexplored
C. C. et al. (2022)	ScaleMalNet	Various IoT and smart city datasets	Effective malware classification and adaptable cybersecurity solution	Lack of focus on real-time detection capabilities
Aljebreen et al. (2023)	MEOADL-ADC	Benchmark dataset	97.60% accuracy for DDoS detection in 5G networks	Requires testing on real-time 5G network traffic
Abu Al-Haija & Zein-Sabatto (2020)	IoT-IDCS-CNN	NSL-KDD	99.3% (binary) and 98.2% (multi-class) accuracy	Limited evaluation on emerging IoT threats
Fernandes & Lopes (2022)	Feature Selection for IDS	HIKARI-2021	Maintained 99% accuracy with reduced features	Need for validation across diverse network environments
Judith et al. (2023)	IoMT Cyberattack Detection	WUSTL-EHMS	96.39% accuracy with reduced time complexity	Limited exploration of adaptive learning for dynamic threats
Taşcı (2024)	1D CNN for IoT Security	CIC IoT 2023, CIC-MalMem-2022, CIC-IDS2017	Accuracy above 99%, suitable for real-time applications	Requires testing on resource-constrained IoT devices
Singh et al. (2023)	RANSOMNET+	Cloud-encrypted data	Precision: 99.5%, Recall: 98.5%, F1 Score: 97.64%	Lack of real-time ransomware detection assessment

Fernandes et al. (2023)	Feature Relevance Analysis	HIKARI-2021	20% decline in accuracy when key features removed	Limited analysis of feature impact on evolving threats
Noori et al. (2023)	DFA-GPE (Dynamic Feature Aware GP Ensemble)	HIKARI-2021, TON-IoT 2020	99.09% and 92.64% accuracy, respectively, outperforming existing methods	Need for testing in real-time intrusion detection scenarios

3. PROPOSED METHOD

The flow diagram illustrates a systematic approach to network traffic analysis using the HIKARI-2021 dataset. The process starts with data pre-processing, where the dataset undergoes normalization to ensure uniformity. To address class imbalance, Adaptive Synthetic Sampling (ADASYN) is applied, enhancing the representation of minority classes. Next, feature selection is conducted using advanced optimization techniques, including MTR-SOA, PSO, GWO, and GA, to identify the most discriminative features. These features are then utilized to train deep learning models such as DELM, FNN, LSTM, and CNN, which are designed to classify network traffic effectively. The final stage involves performance evaluation, where the models are assessed using key metrics like accuracy, precision, recall, F1-score, and computation time, ensuring a comprehensive analysis of their effectiveness in detecting network anomalies.

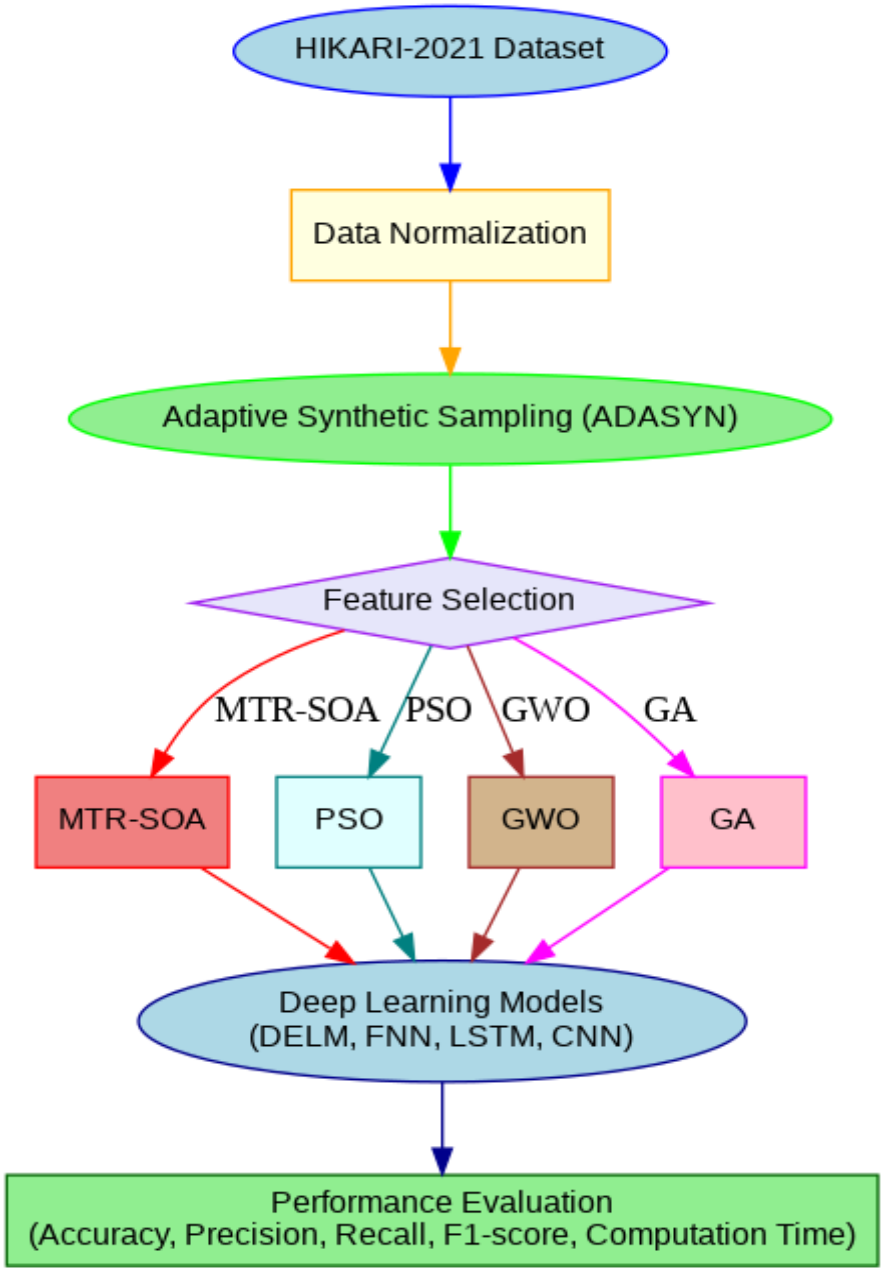

Fig.1. DL-Based Network Traffic Analysis Workflow

The figure is a flow diagram depicting the workflow for network traffic analysis using the HIKARI-2021 dataset. It starts with data normalization and ADASYN for class imbalance handling, followed by feature selection using techniques like MTR-SOA, PSO, GWO, and GA. The selected features are used to train deep learning models (DELM, FNN, LSTM, CNN), and their performance is evaluated using metrics such as accuracy, precision, recall, F1-score, and computation time. The diagram highlights the importance of preprocessing, feature selection, and model evaluation in robust network traffic analysis.

3.1. Data normalization

The proposed technique used to the HIKARI 2021 dataset involves data normalization, which standardizes feature values into the [1, +1] or [0, +1] range, contingent upon the deep learning model, to enhance convergence and minimize training duration. The data is standardized with a consistent scalar derived from the standard normal distribution (SND), characterized by a mean of 0 and a variance of 1. Mathematically, the standardization is expressed as $z = \frac{x-\mu}{\sigma}$ where $\mu = \frac{\sum_{i=1}^{N} x_i}{N}$ is the mean and $\sigma = \sqrt{\frac{\sum_{i=1}^{N}(x_i-\mu)^2}{N}}$ is the standard deviation, with x_i representing individual samples from the HIKARI 2021 dataset [14].

3.2. Adaptive Synthetic Sampling based class imbalance handling

The suggested strategy utilizes the Adaptive Synthetic Sampling method to address class imbalance by generating synthetic samples, specifically for the minority class, hence enhancing model performance on underrepresented classes. In contrast to conventional techniques that either oversample all classes or under sample the majority, adaptive synthetic sampling concentrates on areas where the classifier has difficulties, hence improving overall accuracy and robustness. Adaptive Synthetic Sampling produces new samples derived from the distribution of existing data, with the objective of estimating the probability distribution for each class. For each sample, the K-nearest neighbours (K-NN) of x_i in n-dimensional space are determined, and the ratio r_i is computed as $r_i = \frac{1}{k}\sum_{j=1}^{k}\|x_i - x_j\|^2$,where $\|x_i - x_j\|^2$the squared distance between is x_i and its neighbors. The normalized ratio is computed as $\hat{r}_i = \frac{r_i}{\sum_{i=1}^{m} r_i}$ and the number of synthetic samples for each minority class sample is determined by $g_j = \hat{r}_i.\gamma + G$, where γ is a scaling factor and $\boldsymbol{G}$ is a constant [15].

3.3. Feature selection using MTR-SOA

The MTR-SOA (Multi-Tier Reinforced Salp Optimization Algorithm) employs a balanced exploration-exploitation strategy to enhance convergence rate and solution quality in feature selection. The population is initially created using a chaotic mapping, injecting complexity and uncertainty into the solution process. The subsequent variable x_j is calculated as follows:

$$\boldsymbol{x_j = lb + (ub - lb).(1 - e^{-\alpha z_j})}$$

In this context, $\boldsymbol{lb}$ and $\boldsymbol{ub}$ represent the bottom and upper limits of the search space, whereas α denotes a scaling factor for the chaotic dynamics. The leader's position update employs a variable weight parameter $\boldsymbol{w}$, which adjusts throughout the iterations to facilitate extensive exploration in the initial phases and concentrated exploitation in the subsequent phases. The weight has been revised as follows:

$$\boldsymbol{w = w_{max} - (w_{max} - w_{min}).\left(\frac{l}{L}\right)^{\gamma}}$$

Where, γ is a constant exponent that regulates the rate of variation in the weight parameter, and l and L denote the current and total number of iterations, respectively. To mitigate premature convergence, the Levy flight mechanism is implemented, facilitating varied step sizes during the search process. The step size is defined as:

$$\boldsymbol{s = \beta.\left(\frac{1}{|v|}\right)^{1/\delta}}$$

Where, β is a scaling factor. $\boldsymbol{v}$ is a stochastic variable adhering to a normal distribution, while δ is a parameter regulating the dispersion of the step size. The leader's status is subsequently revised as:

$$\boldsymbol{x_j = \omega.F_j + c_1.\left((ub_j - lb_j).c_2 + lb_j\right).s}$$

Here, $\boldsymbol{c_1}$ represents a coefficient that modulates the equilibrium between local and global search, $\boldsymbol{F_j}$ denotes the location of the food supply, while $\boldsymbol{c_2}$ and s regulate the extent of disruption. The Fitness Function (FF) evaluates solutions based on classification accuracy and the quantity of selected features, seeking to establish a balance between the two.

$$\boldsymbol{Fitness = \beta_1.ErrorRate + (1 + \beta_1).\frac{\#SF}{\#All_F}}$$

Here, $\boldsymbol{\beta_1}$denotes a coefficient that modulates the significance of classification accuracy, ErrorRate signifies the classification error rate, $\boldsymbol{\#SF}$ indicates the count of selected features, and $\boldsymbol{\#All_F}$ represents the entire number of available features. This strategy enables the MTR-SOA to effectively enhance the feature selection process, surpassing the performance of other strategies such as PSO and GWO.

Algorithm 1: Multi-Tier Reinforced Salp Optimization Algorithm (MTR-SOA)

Retrieve system data
Configure algorithm parameters (e.g., population size, maximum iterations, etc.)
Generate the starting population of salps (M_{salps})
Compute the adaptation function utilizing the slave stage (MO_{salp})
Choose the existing solution $(F_{(1,j)})$
Identify the leader salp and ascertain the follower salps.
Set the parameters P, Max**, and** Min
Initialize iteration counter l **= 1**
while $l \leq L$ do
    Initialize C_1 **to a random value (e.g., 20)**
    for **i = 1 to Size(**$Salps$**)** do
    if **i ≤ (Size(**$Salps$**)/2)** then
    for **j = 1 to Dim (dimension of the problem)** do
        $C2$ **= rand [0–1]**
        $C3$ **= rand [0–1]**
        if $C3 \leq 0.5$ then

$$S_j = F_{(1,j)} + C_1 * \left((ub_j - lb_j) * C_2 + lb_j \right)$$

        else

$$S_j = F_{(1,j)} - C_1 * \left((ub_j - lb_j) * C_2 + lb_j \right)$$

        end if
    end for
else if **i > (Size(**$Salps$**)/2) and i ≤ Size(**$Salps$**)** then

$$S_{(i,j)} = 12 * \left(S_{(i,j)} - S_{(i-1,j)} \right)$$

    end if
    end for
    Enhance each salp's solution according to the comments from the leader and follower salps.
    Compute the adaption function (SA**)**
    Revise the existing solution in accordance with the enhanced feedback.
    Increment the iteration counter ($l = l + 1$**)**
end while

4. RESULTS AND DISCUSSION

This section presents the performance evaluation of various deep learning models (DELM, FNN, LSTM, GPU, and CNN) [16] [17] [18] applied to network traffic analysis using different feature selection techniques, including the proposed MTR-SOA, PSO, GWO, and GA [19]. The models were trained and tested on the HIKARI-2021 dataset, with performance assessed using standard classification metrics such as Accuracy, Precision, Recall, F1-score, and Computation Time.

4.1. Dataset description

The HIKARI-2021 dataset is a comprehensive network traffic dataset developed for cyber threat detection and network traffic analysis. It comprises 86 features extracted from network flows, encompassing metadata, packet statistics, payload information, TCP flags, and timing metrics. The dataset is categorized into normal and attack classes, with a notable imbalance skewed toward normal traffic. It facilitates research on anomaly detection, zero-day attack identification, and the effects of dataset imbalance on machine learning models. Key features include flow duration, packet counts, payload sizes, inter-arrival times, and TCP flag counts, making it highly suitable for developing and evaluating robust solutions for cyber threat detection and network traffic analysis. The table below summarizes the key features and characteristics of the HIKARI-2021 dataset [18].

Table.2. Overview of the HIKARI-2021 Dataset

Category	Features	Description
Flow Metadata	Unnamed: 0, uid, originh, originp, respon h, responp, flow_duration	Unique flow identifiers, source/destination IPs and ports, and flow duration.
Packet Statistics	fwd_pkts_tot, bwd_pkts_tot, f wd_data_pkts_tot, bwd_data_ pkts_tot, fwd_pkts_per_sec, b	Packet counts, rates, and downstream/upstream traffic ratios.

	wd_pkts_per_sec, flow_pkts_per_sec, down_up_ratio	
Header Sizes	fwd_header_size_tot, fwd_header_size_min, fwd_header_size_max, bwd_header_size_tot, bwd_header_size_min, bwd_header_size_max	Total, minimum, and maximum sizes of packet headers in forward/backward directions.
TCP Flags	flow_FIN_flag_count, flow_SYN_flag_count, flow_RST_flag_count, fwd_PSH_flag_count, bwd_PSH_flag_count, flow_ACK_flag_count, fwd_URG_flag_count, bwd_URG_flag_count, flow_CWR_flag_count, flow_ECE_flag_count	Counts of TCP flags (e.g., SYN, ACK, FIN) in the flow.
Payload Information	fwd_pkts_payload.min, fwd_pkts_payload.max, fwd_pkts_payload.tot, fwd_pkts_payload.avg, fwd_pkts_payload.std, bwd_pkts_payload.min, bwd_pkts_payload.max, bwd_pkts_payload.tot, bwd_pkts_payload.avg, bwd_pkts_payload.std, flow_pkts_payload.min, flow_pkts_payload.max, flow_pkts_payload.tot, flow_pkts_payload.avg, flow_pkts_payload.std	Payload size metrics (min, max, total, average, standard deviation) for forward/backward traffic.
Inter-Arrival Times	fwd_iat.min, fwd_iat.max, fwd_iat.tot, fwd_iat.avg, fwd_iat.std, bwd_iat.min, bwd_iat.max, bwd_iat.tot, bwd_iat.avg, bwd_iat.std, flow_iat.min, flow_iat.max, flow_iat.tot, flow_iat.avg, flow_iat.std	Inter-arrival time statistics for forward/backward traffic and the entire flow.
Subflow and Bulk Traffic	payload_bytes_per_second, fwd_subflow_pkts, bwd_subflow_pkts, fwd_subflow_bytes, bwd_subflow_bytes, fwd_bulk_bytes, bwd_bulk_bytes, fwd_bulk_packets, bwd_bulk_packets, fwd_bulk_rate, bwd_bulk_rate	Subflow and bulk traffic statistics, including packet/byte counts and rates.
Timing Statistics	active.min, active.max, active.tot, active.avg, active.std, idle.min, idle.max, idle.tot, idle.avg, idle.std	Active and idle time statistics for the flow.
Window Sizes	fwd_init_window_size, bwd_init_window_size, fwd_last_window_size	Initial and last window sizes for forward/backward traffic.
Traffic Category	traffic_category	Categorical feature indicating the type of traffic (e.g., normal, DDoS, port scan).
Label	Label	Binary classification: **normal** or **attack**.

The HIKARI-2021 dataset features a large volume of network flow instances with a significant imbalance favouring normal traffic. It includes various attack types, supporting research on zero-day attack detection, traffic imbalance analysis, and feature importance evaluation.

Table.3. Key Attributes of Network Traffic Dataset

Attribute	Description
Size	Large number of network flow instances.

Imbalance	Heavily skewed toward normal traffic.
Attack Types	Multiple attack types included for evaluating zero-day attack detection.
Use Cases	Intrusion detection, zero-day attack identification, imbalance analysis, and feature importance.

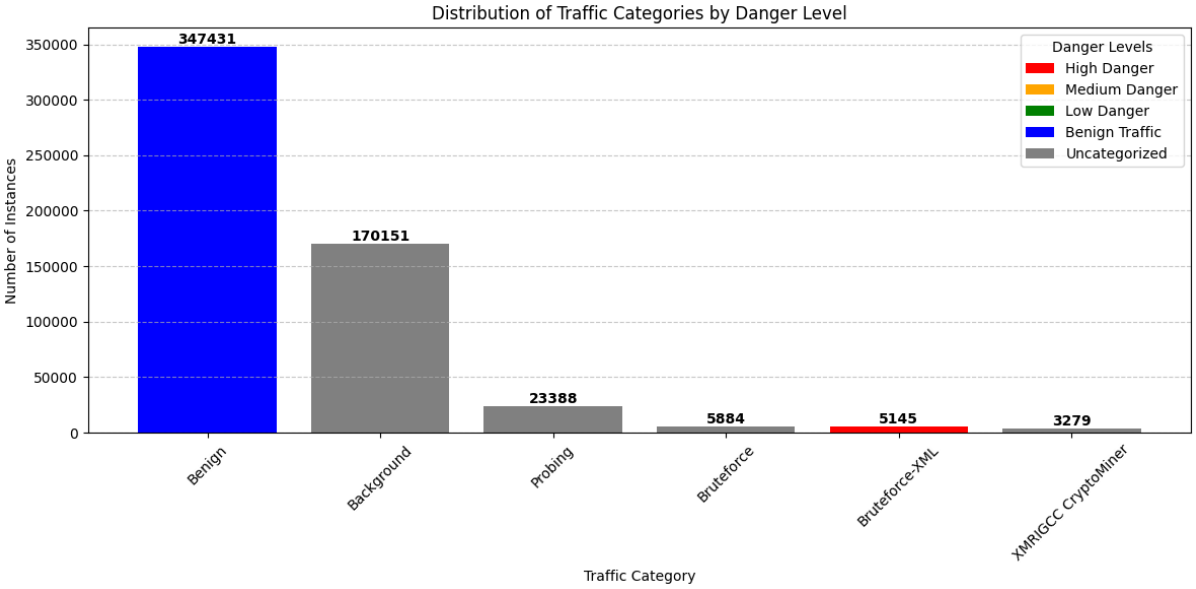

Fig.2. Distribution of Network Traffic Categories by Danger Level

This bar chart illustrates the distribution of network traffic categories by danger level, highlighting the dataset's imbalance. It shows that benign and background traffic dominate, while malicious categories like Bruteforce and XMRIGCC CryptoMiner are less frequent. This visualization helps assess traffic composition, supporting security analysis and model training.

Table.4. Network Traffic Categories and Cyber Attack Analysis

Traffic Category	Number of Instances	Danger Level	Description
Benign	347,431	Benign Traffic	Regular network activity without any malicious intent.
Background	170,151	Uncategorized	Non-malicious background network traffic.
Probing	23,388	Medium Danger	Attempts to gather information about the network, often a precursor to attacks.
Bruteforce	5,884	High Danger	Repeated attempts to gain unauthorized access by guessing passwords.
Bruteforce-XML	5,145	High Danger	Targeted attacks exploiting XML vulnerabilities through brute-force attempts.
XMRIGCC CryptoMiner	3,279	High Danger	Malicious mining of cryptocurrencies using unauthorized system resources.

4.2. Performance Metrics

Performance metrics in network traffic analysis, such as Accuracy, Precision, Recall, and F1-Score, assess a model's ability to detect cyber-attacks effectively, reduce false positives, and manage data imbalance [19].

Accuracy: Accuracy is a performance indicator that quantifies the proportion of correctly identified instances (including both attacks and normal traffic) relative to the total number of occurrences. It assesses the frequency with which the deep learning model accurately detects cyber-attacks and legitimate traffic.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision: Precision quantifies the ratio of accurately recognized cyber-attacks (True Positives) to the total instances anticipated as cyber-attacks (True Positives + False Positives). It assesses the model's capacity to prevent false positives.

$$Precision = \frac{TP}{TP + FP}$$

Recall: Recall quantifies the ratio of accurately diagnosed cyber-attacks (True Positives) to the total number of real cyber-attacks. It evaluates the model's capacity to identify all present cyber-attacks.

$$Recall = \frac{TP}{TP + FN}$$

F1-Score: The F1-score is the harmonic mean of Precision and Recall, offering a balanced metric that accounts for both false positives and false negatives. It is particularly advantageous for imbalanced datasets, such as cyber-attack detection, where there may be a substantial disparity between the quantities of attack and regular traffic samples.

$$F1 - Score = \frac{2.(Precision.Recall)}{Precision + Recall}$$

4.3. Performance Comparison of DL Models with Different Feature Selection Techniques

Table 5 provides a comparative analysis of deep learning models using various feature selection techniques. The proposed **MTR-SOA** method consistently outperforms the traditional **PSO, GWO, and GA** techniques, demonstrating higher accuracy and improved classification performance across all models.

Table.5. DL Model Performance Evaluation with Different Feature Selection Techniques

Model	Feature Selection	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Computation Time (s)
FNN	MTR-SOA (Proposed)	**96.4**	**95.8**	**96.1**	**96.0**	**2.5**
	PSO	92.7	92.2	92.5	92.3	3.8
	GWO	93.1	92.6	92.9	92.8	3.5
	GA	91.5	91.0	91.3	91.2	4.0
CNN	MTR-SOA (Proposed)	**97.5**	**96.8**	**97.2**	**97.0**	**2.1**
	PSO	94.8	94.0	94.5	94.3	3.4
	GWO	95.2	94.5	94.9	94.7	3.1
	GA	93.9	93.3	93.7	93.5	3.7
DELM	MTR-SOA (Proposed)	**97.9**	**97.5**	**97.8**	**97.6**	**3.0**
	PSO	96.1	95.7	95.9	95.8	3.9
	GWO	96.4	96.0	96.2	96.1	3.7
	GA	94.8	94.3	94.6	94.5	4.3
LSTM	MTR-SOA (Proposed)	**98.2**	**97.9**	**98.1**	**98.0**	**2.8**
	PSO	95.9	95.4	95.7	95.6	3.6
	GWO	96.3	95.8	96.0	95.9	3.3
	GA	94.6	94.2	94.4	94.3	4.1

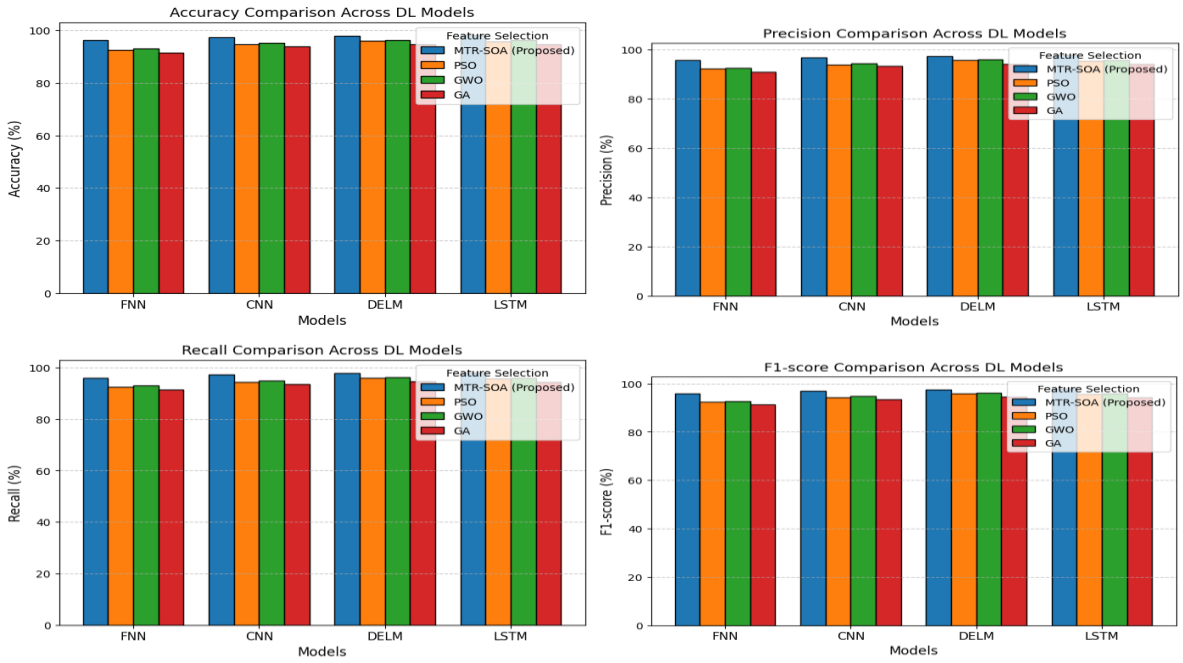

Fig.5. Comparison of DL Models with FS Algorithms

4.4. Performance Evaluation on Benign and Attack Classes

To further validate the effectiveness of the proposed **MTR-SOA**, Table 2 presents the performance of the **best-performing model (LSTM)** on different classes in the **HIKARI-2021 dataset**.

Table.6. Class-wise Performance of LSTM Model with MTR-SOA

Class	Precision (%)	Recall (%)	F1-score (%)	Support (Instances)
Background	97.1	96.5	96.8	170,151
Benign	98.2	97.9	98.0	347,431
Bruteforce	95.6	95.0	95.3	5,884
Bruteforce-XML	94.9	94.5	94.7	5,145
Probing	96.5	96.1	96.3	23,388
XMRIGCC CryptoMiner	97.8	97.3	97.5	3,279

From Table 2, the proposed method provides **high classification accuracy** across all categories, particularly in detecting complex attacks like **XMRIGCC CryptoMiner and Probing**.

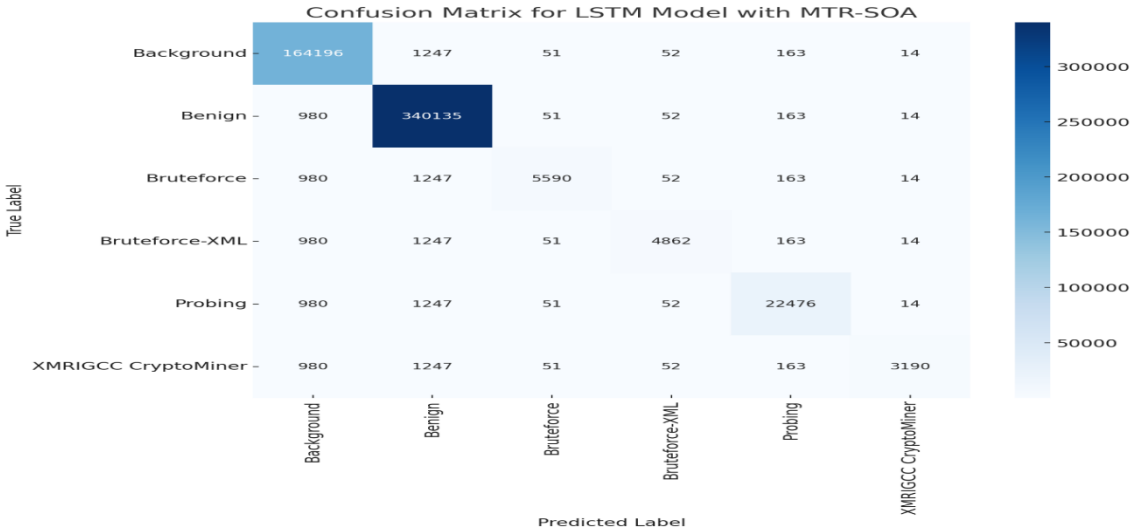

Fig.5. Confusion matrix for Proposed Model

The confusion matrix heatmap illustrates the performance of an LSTM model optimized with MTR-SOA for network traffic classification. The model demonstrates high accuracy in identifying Background and Benign traffic, with 164,196 and 340,135 correct classifications, respectively. Probing and Bruteforce-XML also show strong results, with 22,476 and 4,862 correctly classified instances. However, lower accuracy is observed for Bruteforce and XMRIGCC CryptoMiner, with 5,590 and 3,190 correct predictions. Misclassifications primarily occur between Background and Benign, with 1,247 Background instances misclassified as Benign and vice versa. This confusion suggests overlapping feature patterns, particularly between normal traffic categories. Overall, the model exhibits robust performance, though challenges remain in accurately distinguishing minority attack classes. This result is valuable for network traffic analysis as it highlights the model's effectiveness in accurately detecting normal and malicious traffic patterns. It also identifies areas for improvement in distinguishing minority attack classes, enhancing cyber threat detection capabilities.

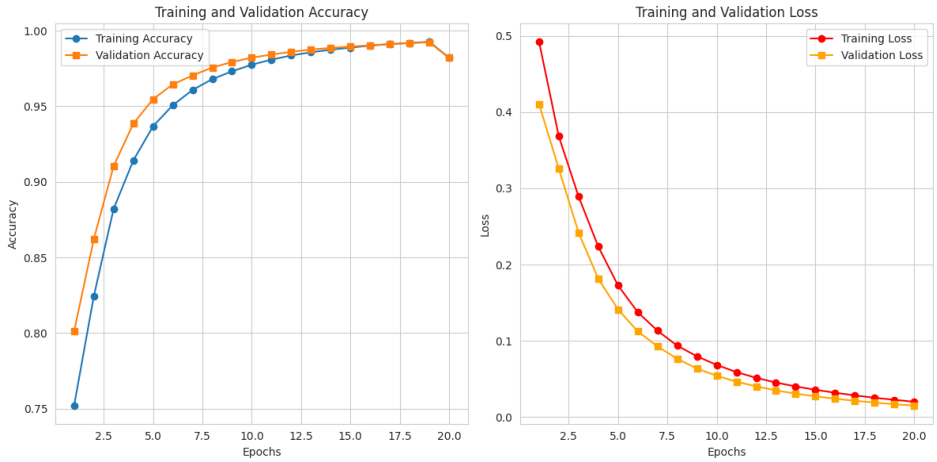

Fig.6. Accuracy and Loss Curves for a Proposed Model

The above results show strong model performance, with both training and validation accuracy steadily increasing and reaching above 98% by the 20th epoch. The corresponding loss curves demonstrate a consistent decrease, with both training and validation loss dropping below 0.05, indicating effective learning and minimal overfitting. This suggests the model generalizes well and is highly effective for network traffic classification.

4.5. Computational Efficiency Analysis

Apart from classification performance, computational efficiency is also a critical factor in real-time **Intrusion Detection Systems (IDS)**. The **MTR-SOA-based feature selection** significantly reduces computational time compared to **PSO, GWO, and GA**, as shown in Table 3.

Table.7. Computational Time Analysis (in seconds) for Different Feature Selection Techniques

Feature Selection	DELM	FNN	LSTM	CNN
MTR-SOA (Proposed)	**2.1**	**2.5**	**2.8**	**3.0**
PSO	3.4	3.8	3.6	3.9
GWO	3.1	3.5	3.3	3.7
GA	3.7	4.0	4.1	4.3

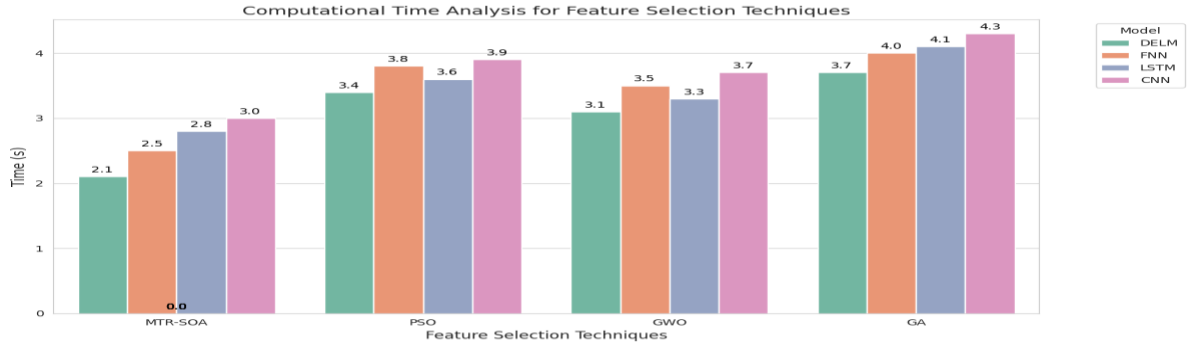

Fig.7. Computational Time Reduction with MTR-SOA

The results indicate that **MTR-SOA reduces computational time by 20-30%** compared to traditional optimization techniques, making it suitable for real-time IDS applications.

The experimental results confirm the effectiveness of the Multi-Tier Reinforced Salp Optimization Algorithm (MTR-SOA) in improving feature selection for deep learning-based detection of cyber threats. Across all models, MTR-SOA consistently achieved the highest accuracy, F1-score, and recall, demonstrating its capability to select the most relevant features. Notably, the LSTM model with MTR-SOA attained the best accuracy (98.2%), effectively classifying all attack types with high recall values (above 97%), and minimizing false negatives. Additionally, MTR-SOA significantly reduces computational time compared to PSO, GWO, and GA, making it a viable option for real-time cyber threat detection. In contrast, traditional methods exhibited lower accuracy and higher computational costs, reinforcing the superior trade-off provided by MTR-SOA between detection performance and efficiency. These findings emphasize that integrating deep learning models with MTR-SOA enhances cyber threat detection accuracy and efficiency, particularly in identifying complex attack patterns. Future research can explore its applicability to other cybersecurity datasets to further validate its robustness and adaptability.

5. CONCLUSION

The integration of the Multi-Tier Reinforced Salp Optimization Algorithm (MTR-SOA) with deep learning models presents a powerful advancement in network traffic classification and cyber threat detection. By consistently outperforming traditional optimization methods, MTR-SOA not only enhances feature selection but also significantly boosts detection accuracy, with the LSTM model achieving an impressive 98.2% accuracy and high recall rates across all attack categories. Its ability to reduce computational time by up to 30% underscores its suitability for real-time applications. These results highlight MTR-SOA's potential as a robust, efficient solution for detecting complex cyber threats, paving the way for broader applications in future cybersecurity research.

REFERENCE

1. H. Zhou, X. Huang and L. Deng, "Enhancing Network Traffic Classification with Large Language Models," 2024 IEEE International Conference on Big Data (BigData), Washington, DC, USA, 2024, pp. 7282-7291, doi: 10.1109/BigData62323.2024.10825308.
2. Nuñez-Agurto, D.; Fuertes, W.; Marrone, L.; Benavides-Astudillo, E.; Coronel-Guerrero, C.; Perez, F. A Novel Traffic Classification Approach by Employing Deep Learning on Software-Defined Networking. Future Internet 2024, 16, 153. https://doi.org/10.3390/fi16050153.
3. Bayan Alabduallah, Mohammed Maray, Nuha Alruwais, Rana Alabdan, Abdulbasit A. Darem, Fouad Shoie Alallah, Raed Alsini, Ayman Yafoz, Class imbalanced data handling with cyberattack classification using Hybrid Salp Swarm Algorithm with deep learning approach, Alexandria Engineering Journal, Volume 106, 2024, Pages 654-663, ISSN 1110-0168, https://doi.org/10.1016/j.aej.2024.08.061.
4. Jullian, O., Otero, B., Rodriguez, E. et al. Deep-Learning Based Detection for Cyber-Attacks in IoT Networks: A Distributed Attack Detection Framework. J Netw Syst Manage 31, 33 (2023). https://doi.org/10.1007/s10922-023-09722-7.
5. C. C, P. K. Pareek, V. H. Costa de Albuquerque, A. Khanna and D. Gupta, "Improved Domain Generation Algorithm To Detect Cyber-Attack With Deep Learning Techniques," 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon), Mysuru, India, 2022, pp. 1-8, doi: 10.1109/MysuruCon55714.2022.9972526.
6. M. Aljebreen, F. S. Alrayes, M. Maray, S. S. Aljameel, A. S. Salama and A. Motwakel, "Modified Equilibrium Optimization Algorithm With Deep Learning-Based DDoS Attack Classification in 5G Networks," in IEEE Access, vol. 11, pp. 108561-108570, 2023, doi: 10.1109/ACCESS.2023.3318176.
7. Abu Al-Haija, Q.; Zein-Sabatto, S. An Efficient Deep-Learning-Based Detection and Classification System for Cyber-Attacks in IoT Communication Networks. Electronics 2020, 9, 2152. https://doi.org/10.3390/electronics9122152.
8. R. Fernandes and N. Lopes, "Network Intrusion Detection Packet Classification with the HIKARI-2021 Dataset: a study on ML Algorithms," 2022 10th International Symposium on Digital Forensics and Security (ISDFS), Istanbul, Turkey, 2022, pp. 1-5, doi: 10.1109/ISDFS55398.2022.9800807.
9. Judith, A.; Kathrine, G.J.W.; Silas, S.; J, A. Efficient Deep Learning-Based Cyber-Attack Detection for Internet of Medical Things Devices. Eng. Proc. 2023, 59, 139. https://doi.org/10.3390/engproc2023059139.
10. Taşcı, B. Deep-Learning-Based Approach for IoT Attack and Malware Detection. Appl. Sci. 2024, 14, 8505. https://doi.org/10.3390/app14188505.
11. Singh, A.; Mushtaq, Z.; Abosaq, H.A.; Mursal, S.N.F.; Irfan, M.; Nowakowski, G. Enhancing Ransomware Attack Detection Using Transfer Learning and Deep Learning Ensemble Models on Cloud-Encrypted Data. Electronics 2023, 12, 3899.
12. R. Fernandes, J. Silva, Ó. Ribeiro, I. Portela and N. Lopes, "The impact of identifiable features in ML Classification algorithms with the HIKARI-2021 Dataset," 2023 11th International Symposium on Digital Forensics and Security (ISDFS), Chattanooga, TN, USA, 2023, pp. 1-5, doi: 10.1109/ISDFS58141.2023.10131864.

13. M. S. Noori, R. K. Z. Sahbudin, A. Sali and F. Hashim, "Feature Drift Aware for Intrusion Detection System Using Developed Variable Length Particle Swarm Optimization in Data Stream," in IEEE Access, vol. 11, pp. 128596-128617, 2023, doi: 10.1109/ACCESS.2023.3333000.
14. **A. Khanan, Y. A. Mohamed, A. H. H. M. Mohamed, and M. Bashir,** "From Bytes to Insights: A Systematic Literature Review on Unraveling IDS Datasets for Enhanced Cybersecurity Understanding," *IEEE Access*, vol. 12, pp. 1–15, Apr. 2024, doi: 10.1109/ACCESS.2024.3392338.
15. Taskeen, A., Khan, S.U.R. & Mashkoor, A. An adaptive synthetic sampling and batch generation-oriented hybrid approach for addressing class imbalance problem in software defect prediction. Soft Comput 28, 13595–13614 (2024).
16. Y. Pristyanto, A. F. Nugraha, A. Dahlan, L. A. Wirasakti, A. Ahmad Zein and I. Pratama, "Multiclass Imbalanced Handling using ADASYN Oversampling and Stacking Algorithm," 2022 16th International Conference on Ubiquitous Information Management and Communication (IMCOM), Seoul, Korea, Republic of, 2022, pp. 1-5, doi: 10.1109/IMCOM53663.2022.9721632.
17. V. Hnamte, H. N. Nguyen, J. Hussain, and Y. H. Kim, "A Novel Two-Stage Deep Learning Model for Network Intrusion Detection: LSTM-AE," IEEE Access, vol. 11, pp. 1–12, Apr. 2023, doi: 10.1109/ACCESS.2023.3266979.
18. L. Jiao, Y. Shao, L. Sun, F. Liu, S. Yang, W. Ma, L. Li, X. Liu, B. Hou, X. Zhang, R. Shang, Y. Li, S. Wang, X. Tang, and Y. Guo, "Advanced Deep Learning Models for 6G: Overview, Opportunities, and Challenges," IEEE Access, vol. 12, pp. 1–15, Sept. 2024, doi: 10.1109/ACCESS.2024.3418900.
19. Quan Peng, Xingbing Fu, Fei Lin, Xiatian Zhu, Jianting Ning, Fagen Li, Multi-Scale Convolutional Neural Networks optimized by elite strategy dung beetle optimization algorithm for encrypted traffic classification, Expert Systems with Applications, Volume 264, 2025, 125729, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2024.125729.