Optimizing Threat Intelligence Strategies for Cybersecurity Awareness Using MADM and Hybrid GraphNet-Bipolar Fuzzy Rough Sets

Qian Zhang
Shaanxi Police College,
Xi'an, Shaanxi, 710021, China

***Abstract*—Advanced threat detection systems are needed more than ever as cyber-attacks become more advanced. A novel cybersecurity model uses Bipolar Fuzzy Rough Sets, Graph Neural Networks, and dense network (BFRGD-Net) architectures to identify threats with unmatched accuracy and speed. The approach optimizes threat detection using Dynamic Range Realignment, anomaly-driven feature enhancement, and a hybrid feature selection strategy on a comprehensive Texas dataset of 66 months of real-world network activity. With 97.8% accuracy, 97.5% F1-score, and 98.3% AUC, BFRGD-Net sets new standards in the field. Threat Detection Sensitivity shows the model's capacity to find uncommon, high-severity threats, while Balanced Risk Detection Efficiency provides fast, accurate threat detection. The model has strong correlations and the highest statistical metrics scores compared to other techniques. Extensive simulations demonstrate the model's capacity to discern threat levels, attack kinds, and response techniques. BFRGD-Net revolutionizes cybersecurity by seamlessly merging cutting-edge machine learning with specific insights. Its advanced threat detection and classification engine reduces false negatives and enables proactive critical infrastructure protection in real-time. The model's adaptability to various attack situations makes it vital for cybersecurity resilience in a digital environment.**

***Keywords*—Cybersecurity awareness; threat intelligence; MADM framework; BFRGD-Net; hybrid model; deep learning**

I. Introduction

Advancements in technology have raised cybersecurity risks such as unauthorized access, malware attacks, phishing, and DoS [1]. Over 900 million malware executables existed in 2024, compared to 50 million in 2010 [2]. Annual cybercrime costs firms, individuals, and governments $400 billion. Due to data breaches and security incidents, essential systems and data require cybersecurity. Modernizing security helps companies avoid losses and adapt to shifting threats. Cybersecurity shields data, programs, networks, and systems against cyberattacks and unauthorized access [2]. Security for networks, applications, data, and operations. Data-driven cybersecurity solutions complement firewalls, antivirus, and intrusion detection. To improve threat detection, machine learning (ML) may find hidden patterns and irregularities. Cybersecurity machine learning is part of AI-powered decision-making and threat identification. Attackers' expertise in exploiting connected technologies is leading to more complex cyber threats [3]. From 2015 to 2022, cybersecurity and machine learning gained popularity from 30 to over 70 [3]. ML increases complex dataset analysis, security, and incident response. Traditional security measures, such as user authentication, cryptographic systems, and firewalls, need human setup and maintenance, making them less effective as threats evolve [4] Thus, machine learning and data analytics-based adaptive and automated systems discover new risks and provide robust protection.

Cybersecurity situational awareness (SA) involves collecting, evaluating, and interpreting information from several sources to manage risks [5]. SA, created for military usage, is currently utilized in cybersecurity to understand activities. Real-time vulnerability and network traffic analysis may prevent attacks. SA frameworks integrate data from several sources, including network traffic and vulnerability assessments, to provide a comprehensive security picture [5]. SA predicts and detects cyberattacks using many data sources. Many situational awareness methods manually gather and analyze data. Despite advancements in data fusion and machine learning, human participation remains vital [6]. Our fully automated systems must use the Common Vulnerability Scoring System to monitor and evaluate network parts for vulnerabilities and security risks.

Company cybersecurity awareness initiatives may raise security awareness and share responsibilities. A frequent cybersecurity risk is the human component since workers are the weakest link in the security chain [7]. Security-focused workplaces prevent errors and social engineering. Cyberattacks may harm reputation, legal obligations, consumer confidence, and money. Secure solutions must address technology and humans. As more people use the internet, hackers target browsers. Using browser vulnerabilities, attackers may get access to devices and steal sensitive data [8]. Covert downloads from reputable websites spread 22 million malware types in 2022. User data must be protected from browser-based attacks using security, notably automated threat detection. Cybersecurity requires flexibility. To resist growing threats, cybersecurity defenses must adapt to new attack methods [9]. Machine learning and situational awareness automate danger detection in this essay. Multi-domain threat detection is improved by diverse data sources and new techniques. Main contributions of this research.

1) The BFRGD-Net framework is introduced to enhance threat intelligence, integrating BFRS, GNN, and DenseNet architectures to improve Cyber Secu-

rity Awareness Programs. This advanced approach facilitates accurate threat detection by identifying intricate patterns in network traffic, hence enhancing the precision and dependability of threat intelligence techniques.

2) The study introduces innovative preprocessing techniques, including HTS and CBSA, that substantially improve data quality and model resilience in support of cyber security awareness. By mitigating data imbalance and noise, these solutions enable threat intelligence tactics to react to emerging cyber threats, hence enhancing the efficacy of awareness programs.
3) Hybrid Feature Selection for Enhanced Threat Identification: The use of a hybrid feature selection methodology that integrates Statistical-Driven Filtering, Redundancy Aggregation, and ODAS guarantees the prioritization of the most pertinent and significant characteristics. This enhancement improves threat detection, making it a substantial tool for advancing Cyber Security Awareness Programs.
4) The research introduces new metrics—TDS, AIS, and BRDE—that provide a more thorough assessment of threat detection efficacy. These metrics are designed to assess the model's capacity to recognize significant risks and optimize detection efficiency, aiding in the identification of the most effective tactics for improving cybersecurity awareness.
5) Exhibiting Enhanced Threat Detection to Guide Awareness Initiatives: Comprehensive assessments indicate that BFRGD-Net routinely surpasses current models, demonstrating superior capability in detecting diverse attack types and categorizing threat severity levels. This performance highlights the framework's capacity to enhance Cyber Security Awareness Programs via the provision of actionable and timely threat information.

The remaining structure of the paper: Section 2 discussed the review of relevant literature. The proposed method structure is described in detail in Section 3. The simulations and their accompanying discussion are detailed in Section 4. The last section concludes with a discussion of future work.

II. Related Work

Threat intelligence and cybersecurity awareness initiatives have been improved using machine learning and decision-making frameworks. The NIST cybersecurity lifecycle consists of five steps: Identify, Protect, Detect, Respond, and Recover, offering a systematic threat management strategy [10]. However, many studies disregard lifecycle-wide techniques and concentrate on one or two processes. One threat intelligence technique in Software-Defined Networking (SDN) uses a Support Vector Machine (SVM) to discover attack scenarios and analyze vulnerabilities [11]. Another method uses K-Nearest Neighbors (KNN) to categorize IoT network traffic by risk level [12]. These strategies emphasize identification without incorporating threat intelligence throughout the lifecycle.

Threat detection is improved via machine learning. An anomaly-based intrusion detection system (IDS) increased prediction accuracy by using Recurrent Neural Networks (RNNs) to identify unexpected network patterns [13]. A Multiclass Support Vector Machine (SVM) was used to categorize network abnormalities in real-time, successfully distinguishing attack types [14]. Due to obsolete datasets, many methods fail to identify new threats. Users learn to recognize and react to cyber dangers via cybersecurity awareness programs. Static material in traditional systems may not be adequate for developing threats [15]. A dynamic strategy using Graph Neural Networks (GNNs) was suggested to update cybersecurity training material depending on current threat information [16]. Using Convolutional Neural Networks (CNNs), another application created interactive training simulations [17]. These strategies increased training but did not integrate live threat intelligence data. Cybersecurity methods are evaluated using MADM frameworks. One method assessed intrusion detection trade-offs using Decision Trees (DT) based on accuracy and reaction time [18]. Analytical Hierarchy Process (AHP) and Logistic Regression (LR) were used to prioritize threat response techniques and evaluate their influence on security [19]. Although effective, MADM approaches are seldom used to improve awareness training.

Deep learning-fuzzy logic hybrid models can handle cybersecurity uncertainty. LSTM networks and Fuzzy C-Means clustering were utilized to identify Advanced Persistent Threats (APTs) using fuzzy logic to handle imprecise data [20]. A study used Deep Belief Networks (DBN) and Fuzzy Inference Systems (FIS) to improve network anomaly classification accuracy under uncertainty [21]. Fuzzy rough set techniques have been used in cybersecurity research, notably for uncertain decision-making. Bipolar Fuzzy Rough Sets (BFRS) add bipolar information to fuzzy logic for more sophisticated decision-making. In the study, BFRS using Random Forest (RF) classifiers improved phishing attack detection rates by addressing uncertainty in approaches [22]. BFRS and Neuro-Fuzzy Systems (NFS) were combined to enhance threat categorization and protect against zero-day attacks [23]. Merging Bipolar Fuzzy C-Means (BFCM) clustering with BFRS may enhance network traffic anomaly detection and reduce false positives [24].

Cybersecurity hybrid models and decision-making frameworks are difficult to integrate. Many frameworks lack thorough integration of situational awareness technologies and threat intelligence methodologies, and obsolete datasets hinder innovative threat detection. The summarized view of the literature is shown in Table I.

TABLE I. Summarized Literature Review

Ref	Method Used	Objective Achieved	Limitations
[10], [11]	SVM, KNN	Detection of attack scenarios and classification of network flows in SDN and IoT environments based on risk levels.	Focuses mainly on identification without fully integrating the complete threat lifecycle.
[12], [13]	RNN, Multiclass SVM	Anomaly detection and real-time classification of network anomalies, improving prediction accuracy by leveraging historical data.	Relies on outdated datasets, limiting the ability to detect emerging threats.
[14], [15]	GNN, CNN	Dynamic modeling of relationships between cybersecurity events for adaptive training and creating interactive simulations for better user engagement.	Lacks full integration with live threat intelligence data, limiting real-time adaptability.
[16], [17]	Decision Trees, AHP, Logistic Regression	Evaluation of intrusion detection configurations and prioritization of threat response strategies based on various criteria (e.g., accuracy, cost).	Rarely applied to evaluating cybersecurity awareness programs' effectiveness.
[18], [19]	LSTM, Fuzzy C-Means, DBN, FIS	Detection of Advanced Persistent Threats and classification of network anomalies under uncertain conditions.	High computational cost and limited adaptability to novel attack types.
[20], [21], [22]	BFRS, Random Forest, Neuro-Fuzzy Systems, BFCM	Detection of phishing attacks, adaptive threat classification, and enhanced anomaly detection using Bipolar Fuzzy Rough Sets.	May require continuous retraining, with increased sensitivity potentially leading to higher false-positive rates.

A. Challenges and the Need for BFRGD-Net

Cybersecurity threat identification is complicated by uneven datasets, dynamic attack patterns, and real-time reaction. These issues demand a model that can manage uncertainty, rapidly interpret complicated data linkages, and react to changing threats. The BFRGD-Net framework was created around these factors. It uses BFRS, GNNs, and DenseNet to provide a resilient, adaptable solution.

1) Suitability of BFRGD-Net for cybersecurity challenges:

- Bipolar Fuzzy Rough Sets clearly accommodate for ambiguous patterns by separating confidence and uncertainty, making the model more accurate in finding anomalies with confusing features.
- Relational Insights: GNNs capture source-destination dependencies and flow correlations, which are essential for threat categorization.
- DenseNet's layered architecture efficiently propagates and reuses features, allowing it to analyze high-dimensional data without duplicate calculations.
- Balanced Detect: BFRGD-Net balances feature significance and class distributions in cybersecurity datasets using hybrid feature selection (Statistical-Driven Filtering, RRFA, ODAS) and advanced preprocessing (CBSA, HTS).

2) Existing technique limitations:

- Traditional models like SVM and KNN struggle with huge, unbalanced datasets and cannot identify high-severity, low-frequency threats.
- CNNs and LSTMs ignore network traffic related relationships in favor of spatial or temporal patterns.
- Fuzzy logic can manage uncertainty, but hybrid systems like BFRGD-Net give scalability and real-time flexibility.
- Old dataset models cannot adapt to new attack vectors, limiting their real-world usefulness. However, BFRGD-Net's superior preprocessing and dynamic feature selection enable stable performance in changing settings.

BFRS, GNN, and DenseNet in the BFRGD-Net architecture overcome these restrictions, enabling great sensitivity to uncommon threats, real-time efficiency, and flexibility to varied attack scenarios. These traits make BFRGD-Net ideal for current cybersecurity applications, assuring its relevance and efficacy in solving problems.

III. Proposed Method

To improve threat classification accuracy and resilience, the proposed cybersecurity threat detection framework uses modern Deep learning and statistical analysis methodologies. The hybrid architecture, BFRGD-NeT, uses Bipolar Fuzzy Rough Sets (BFRS) and a GNN-DenseNet model to leverage GNNs' relational learning and DenseNet's feature reuse. Data is preprocessed using Dynamic Range Realignment and Perturbation-Weighted Outlier Filtering to ensure quality and correct feature imbalance. A Hybrid Feature Selection technique using Statistical-Driven Filtering and Optimization-Driven Adaptive Selection identifies the most relevant features for classification after preprocessing. Feature transformation improves model training data representation using Scaled Differential Encoding and Exponential Scaling Modulation. Using its capacity to manage ambiguity and relational data, the BFRGD-NeT model classifies threats. Using Adaptive Learning Rate Adjustment, model hyperparameters are fine-tuned for best performance. The proposed framework is shown in Figure 1 and afterwards, each module is explained in detail.

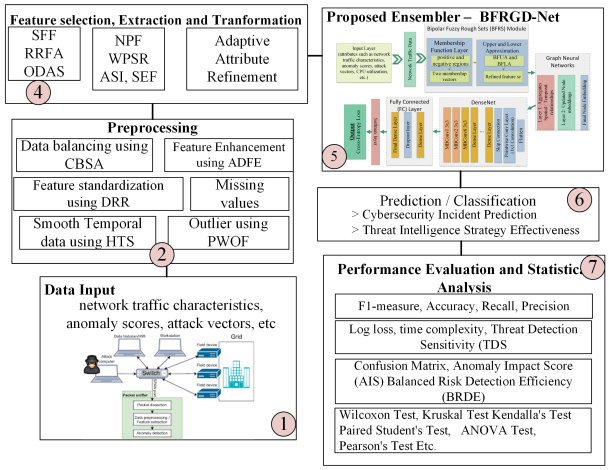

Fig. 1. Proposed system framework.

A. Dataset Description

This research utilized real-world Texas network traffic and cybersecurity activities. Over 66 months, from January 2018 to July 2024, the study collects hourly network events, user behaviors, and system states in an active corporate network infrastructure [23]. Financial companies, healthcare providers, and educational institutions in the area provided vital cybersecurity information for the dataset. These firms follow strong compliance standards to protect data. From low-level abnormalities to high-severity threats, the data shows how cyber dangers evolve. In Texas, a technological and industrial powerhouse, network connections, user behaviors, and worldwide cyberattack vectors are abundant. Information from local threat intelligence feeds was included in the dataset to make it more realistic and depict these firms' cybersecurity environment. This dataset provides a complete picture of regional cybersecurity issues by merging numerous data sources and using threat information from internal and external systems. It presents a solid framework for designing and testing advanced threat intelligence tactics to improve cybersecurity awareness and response. The dataset's imbalance, caused by real-world network traffic and attack events, properly depicts current cybersecurity systems' situations. The dataset characteristics are listed in Table II.

B. Data Preprocessing Steps

The dataset underwent preprocessing using many innovative strategies to enhance its appropriateness for machine

TABLE II. DATASET FEATURES OVERVIEW

S.No	Features	Short Description
1	Source IP	IP address of the source device or user.
2	Destination IP	IP address of the target device or server.
3	Source Port	Port number used by the source device for communication.
4	Destination Port	Port number used by the destination device.
5	Protocol Type	Type of protocol used for communication (TCP, UDP, etc.).
6	Flow Duration	Total duration of the network flow.
7	Packet Size	Size of the transmitted packets during the communication.
8	Flow Bytes/s	Rate of bytes transmitted per second.
9	Flow Packets/s	Rate of packets transmitted per second.
10	Total Forward Packets	Total number of packets sent by the source.
11	...	...
18	Anomaly Score	A score indicating the abnormality level of the traffic.
19	Attack Vector	The method or entry point used by the attack.
20	Botnet Family	Family of botnet detected in network traffic.
21	Anomaly Severity Index	Derived feature measuring anomaly severity.
22	CPU Utilization	CPU usage percentage during traffic transmission.
23	Memory Utilization	Memory usage percentage during the flow.
24	Label	Indicate whether the traffic is normal or an attack.

learning [24][25]. The first phase was Dynamic Range Realignment (DRR), used to normalize feature values according to their dynamic range while preserving variability. The transition is articulated as:

$$y_{adjusted} = \frac{y - \min(y)}{\max(y) - \min(y)} + \beta \cdot \left(\frac{\xi(y)}{\psi(y)} \right) \tag{1}$$

β is a weighting factor, $\xi(y)$ is the standard deviation, and $\psi(y)$ is the feature mean. Features with different scales are modified while keeping their relative dispersion using this method. To reduce noise and preserve trends across time, Hierarchical Temporal Smoothing (HTS) was employed to smooth temporal data across periods. Smoothing involves:

$$y_{smooth} = \frac{1}{h} \sum_{k=1}^{h} (y_{t-k} + y_{t+k}) + \theta \cdot \sum_{l=1}^{n} u_l \cdot y_{t-l} \tag{2}$$

where h represents short-term window size, θ adjusts long-term trend, and u_l weights temporal distances. This successfully captures short-term and long-term tendencies. Perturbation-Weighted Outlier Filtering (PWOF) was created to handle outliers. The following rule filtered outliers:

$$y_{filtered} = \begin{cases} y & \text{if } |y - \text{median}(y)| \leq \zeta \cdot \xi(y) \\ \text{median}(y) & \text{otherwise} \end{cases} \tag{3}$$

where ζ is the threshold determining outlier sensitivity, and $\xi(y)$ is the standard deviation. This method ensures that extreme values are moderated without affecting the general distribution of the feature. To handle class imbalance, *Cluster-Based Synthetic Amplification (CBSA)* was applied, generating synthetic samples for underrepresented classes based on clustering. The synthetic sample generation is governed by:

$$y_{synth} = y_g + \lambda \cdot (y_g - y_h) \tag{4}$$

where y_g is the cluster centroid, y_h is an adjacent point, and λ regulates synthetic amplification. This generates realistic new data points to balance the collection. In conclusion, *Anomaly-Driven Feature Enhancement (ADFE)* was used to enhance important features, particularly surrounding identified assaults. Enhancement calculation is done using the following equation:

$$y_{enhanced} = y \cdot \left(1 + \omega \cdot \frac{1}{1 + e^{-\phi(t - t_{event})}} \right) \tag{5}$$

ω indicates the amplification factor, ϕ adjusts temporal effect, and t_{event} represents the closest attack timestamp. This technique emphasizes anomaly-related characteristics as needed. These preprocessing processes balance, normalize and filter the dataset while keeping temporal and anomaly-based information needed for robust model performance.

C. Hybrid Feature Selection Process

This research uses a hybrid feature selection procedure that combines innovative strategies to maximize relevance and reduce duplication [26]. First, Statistical-Driven Feature Filtering (SFF) assesses feature variance and class separability contribution. For dataset imbalances, SFF dynamically weights feature importance by class distribution, unlike previous approaches. The significance score for feature h is determined as:

$$W_h = \frac{\zeta_h}{\eta_h} \cdot \frac{1}{1 + e^{-\mu(T_h - \bar{T})}} \tag{6}$$

ζ_h and η_h represent the feature's standard deviation and mean, T_h represents its correlation with the target label, and μ regulates the score's sensitivity This prioritizes high-variance, class-separable characteristics. To reduce feature redundancy, Redundancy-Reduced Feature Aggregation (RRFA) was created. This technique penalizes feature content redundancy by measuring overlap. The aggregate score for feature pairs (h_i, h_j) is:

$$V_{h_i,h_j} = \frac{|S_{h_i,h_j}|}{1 + \phi \cdot |S_{h_i,h_j}|} \tag{7}$$

Where S_{h_i,h_j} represents the correlation between features, and ϕ penalizes highly correlated features, preserving only the most representative ones from each correlated collection. Next, the Optimization-Driven Adaptive Selector (ODAS) approach is used. The cost function in ODAS repeatedly adjusts feature weights depending on predicted accuracy while punishing needless complexity. The cost function is:

$$\mathcal{F} = \sum_{h=1}^{m} v_h \cdot R(h) + \theta \cdot \sum_{j=1}^{m} v_j^2 \tag{8}$$

where v_h is feature weight, $R(h)$ is error contribution, and θ is regularization parameter. This refines choices by deleting low-performance model characteristics. SFF, RRFA, and ODAS are combined to keep just the most important, non-redundant features in the hybrid feature selection approach. This method optimizes features for prediction performance and simplicity.

D. Derived Attribute Transformation

This work proposed a unique approach called *Derived Attribute Transformation (DAT)* to create new characteristics from existing ones, therefore improving the dataset's capacity to capture intricate interactions. The first derived feature, *Normalized Packet Flow (NPF)*, integrates forward and backward packet counts normalized by flow time, articulated as:

$$\text{NPF} = \frac{(P_{\text{forward}} + P_{\text{backward}})}{\text{Flow Duration}} \tag{9}$$

This function captures a network flow's communication strength. Computed as weighted packet size ratio (WPSR), another attribute balances forward and backward packet sizes relative to flow bytes:

$$\text{WPSR} = \frac{\text{Forward Packet Size} \cdot \text{Flow Bytes}}{\text{Backward Packet Size} + \epsilon} \tag{10}$$

Using IDS alerts and reputation score, a crucial function Anomaly Severity Index (ASI) increases the anomaly score:

$$\text{ASI} = \text{Anomaly Score} \cdot (1 + \log(1 + \text{IDS Alerts})) \cdot \left(\frac{100 - \text{Reputation Score}}{100}\right) \tag{11}$$

This underscores the importance of reputation and IDS alerts in the identification of hazards. The *Session Efficiency Factor (SEF)* is a metric that integrates system resource utilization and session activity.

$$\text{SEF} = \frac{\text{Active Duration}}{\text{Idle Duration} + \epsilon} \cdot \left(\frac{\text{CPU Utilization}}{\text{Memory Utilization} + \epsilon}\right) \tag{12}$$

Lastly, Dynamic Threat Potential (DTP) figures out risk based on how bad an attack is and how many hosts have been hacked:

$$\text{DTP} = (\text{Attack Severity} \cdot \log(1 + \text{Compromised Hosts})) \cdot \frac{1}{1 + e^{-\gamma \cdot (\text{Attack Vector})}} \tag{13}$$

The newly created features, produced by the DAT method, provide enhanced insights into network activity, hence improving model performance for predictive analysis.

E. Adaptive Attribute Refinement (AAR)

A new method of transformation called Adaptive Attribute Refinement (AAR) is suggested in this research. The goal of this approach is to dynamically modify feature values according to how they affect the distribution of the whole dataset so that important patterns are highlighted and noise is reduced [26]. When characteristics change substantially between classes or time intervals, the AAR process uses an adaptive scaling technique to account for this. In the first stage of transformation, known as Dynamic Weight Adjustment (DWA), the statistical variance and impact of each feature on class separability are used to weigh the value of each feature. The feature y transformation is defined as:

$$y_{\text{adj}} = y \cdot \left(1 + \kappa \cdot \frac{|y - \nu_y|}{\tau_y}\right) \tag{14}$$

where ν_y is the feature mean, τ_y is the standard deviation, and κ is an adaptive scaling factor based on the feature distribution. This emphasizes outliers and major deviations by weighing data further from the mean. The next stage, Contextual Recalibration (CR), refines the converted feature depending on its temporal or category context. In datasets with characteristics that respond differently under different situations (e.g., events), this is beneficial. The recalibration process:

$$y_{\text{rec}} = y_{\text{adj}} \cdot \frac{1}{1 + e^{-\lambda \cdot (D_t - \bar{D})}} \tag{15}$$

λ is a sensitivity factor, D_t is the feature's contextual score at the time (t), and $\bar{D}$ is the dataset's average contextual score This adjusts features depending on their situation or class relevance. Finally, the Smooth Variance Reduction (SVR) stage reduces noise while maintaining essential trends. This stage maintains the converted feature's variability by smoothing severe variations. Transformation is modulated by:

$$y_{\text{smooth}} = \frac{y_{\text{rec}}}{1 + \theta \cdot \rho_{\text{local}}} \tag{16}$$

θ is a tuning parameter, and ρ_{local} is the local variance within a preset window of neighboring values. Short-term spikes are reduced but long-term patterns are maintained. The integrated Adaptive Attribute Refinement (AAR) technique produces a robust transformation process that improves the model's crucial pattern detection, noise reduction, and context-aware dataset refinement. This technique dynamically adapts each feature depending on its importance to the task, improving model accuracy and resilience.

This work uses BFRGD-Net, a unique classification model that combines Bipolar Fuzzy Rough Sets (BFRS) with GNN-DenseNet. This hybrid model benefits from Bipolar Fuzzy Rough Sets (BFRS) for uncertainty and interpretability, Graph Neural Networks (GNNs) for relational learning, and DenseNet for feature propagation and reuse. This layered structure is ideal for high-dimensional, complicated data like cybersecurity threats, where local and global interactions are crucial to threat identification.

F. Classification using BFRGD-Net

An improved method for handling uncertainty is presented by the Bipolar Fuzzy Rough Sets (BFRS) [27] framework, which describes the dataset using positive and negative areas. Positive associations represent certainty, while negative connections capture doubt. The proposed layered archictecutre is shown in Figure 2.

This is of the utmost importance in the field of cybersecurity since irregularities often manifest in unpredictable data patterns. The membership functions, both positive and negative, are defined by the BFRS method as:

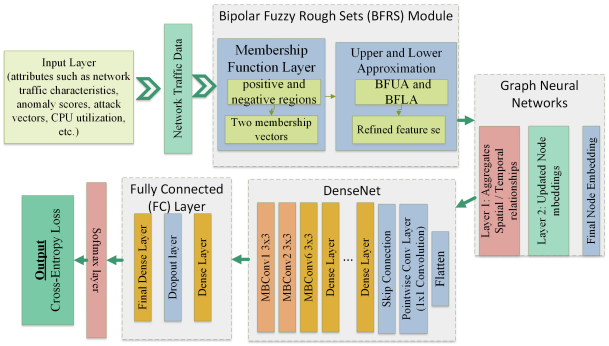

Fig. 2. Proposed BFRGD-Net architecture.

$$P(Z) = \{z \in U \mid \nu_1(z) \geq \alpha\}, \quad N(Z) = \{z \in U \mid \nu_2(z) \leq \beta\} \tag{17}$$

In this instance, $P(Z)$ and $N(Z)$ denote the positive and negative areas, respectively, while $\nu_1(z)$ and $\nu_2(z)$ signify the membership functions for the positive and negative classes. Additionally, α and β represent the thresholds that delineate the certainty and uncertainty regions. The Bipolar Fuzzy Upper Approximation (BFUA) and Bipolar Fuzzy Lower Approximation (BFLA) of a set Z are defined as follows:

$$\text{BFUA}(Z) = \sup_{z \in U} \left(\min\left(\nu_1(z), 1 - \nu_2(z)\right)\right) \tag{18}$$

$$\text{BFLA}(Z) = \inf_{z \in U} \left(\max\left(\nu_1(z), 1 - \nu_2(z)\right)\right) \tag{19}$$

These approximations improve the model's capacity to identify ambiguous data points, aiding in the identification of indistinct cybersecurity risks that display obscure patterns. The Graph Neural Network (GNN) component captures relational connections in graph-structured data, including network traffic and system logs. In Graph Neural Networks (GNNs), each node $g_i^{(l)}$ at layer l consolidates information from its adjacent nodes to enhance its feature representation in the subsequent layer. The propagation rule for a Graph Neural Network layer is delineated as follows:

$$g_i^{(l+1)} = \sigma\left(W^{(l)} \cdot \sum_{j \in \mathcal{N}(i)} \frac{1}{\sqrt{d_i d_j}} g_j^{(l)} + b^{(l)}\right) \tag{20}$$

$g_i^{(l+1)}$ is the updated node feature for node i at layer $l+1$, $W^{(l)}$ is the weight matrix for layer l, d_i and d_j are the degrees of nodes i and j, and σ is the activation function $\frac{1}{\sqrt{d_i d_j}}$ normalizes node connections, enabling balanced information aggregation from nearby nodes. In the cybersecurity dataset, the model may capture local (node-level) and global (graph-level) interactions.

The DenseNet design tightly connects each layer to every other layer feed-forward for optimal feature reuse. This propagates previous layer feature maps without losing critical information, which is vital for deep model performance. The DenseNet l-th layer output is computed as:

$$x^{(l)} = T_l\left(\left[x^{(0)}, x^{(1)}, \ldots, x^{(l-1)}\right]\right) \tag{21}$$

T_l is the transformation (e.g., batch normalization, ReLU, convolution) performed to the concatenated input from all preceding layers $[x^{(0)}, x^{(1)}, \ldots, x^{(l-1)}]$. The model can capture complex cybersecurity data relationships by effectively propagating information by concatenating feature maps. This reduces feature extraction parameters. DenseNet transition layers pool feature maps to reduce dimensionality while preserving crucial information. Transition layer outputs are calculated as:

$$S(x) = \text{BN}\left(\text{ReLU}\left(\text{Conv}(x)\right)\right) \tag{22}$$

where $S(x)$ is transition layer output, BN is batch normalization, $ReLU$ is activation function, and $Conv$ is input feature map convolution. The model stays small and economical without losing performance with this operation. A fully connected layer and softmax function complete the classification stage after GNN-DenseNet processing. Class probabilities for each cybersecurity threat are calculated using learning attributes. Definition of softmax:

$$\hat{y}_k = \frac{e^{o_k}}{\sum_{j=1}^{C} e^{o_j}} \tag{23}$$

The predicted probability for class k is $\hat{y}_k$, the logit output is o_k, and the total number of classes is C. Cybersecurity risks may be accurately classified using the softmax function's probability distribution across all classes. BFRGD-Net training uses cross-entropy loss, which is ideal for multi-class classification problems like threat detection. The cross-entropy loss $\mathcal{L}$ is computed as:

$$\mathcal{L} = -\sum_{k=1}^{C} y_k \log(\hat{y}_k) \tag{24}$$

y_k is the actual label for class k; $\hat{y}_k$ is the projected probability for class k. This loss function motivates the model to provide greater probability for the proper class and penalizes erroneous projections.

Bipolar Fuzzy Rough Sets (BFRS), Graph Neural Networks (GNNs), and DenseNet form the BFRGD-Net architecture, which detects cybersecurity risks effectively. BFRS improves the model's uncertainty handling, GNN layers capture network traffic relational relationships, and DenseNet optimizes feature reuse. With the softmax classification layer and cross-entropy loss function, the model can reliably identify many cybersecurity risks while being computationally efficient. That makes BFRGD-Net ideal for real-time threat identification in complicated cybersecurity contexts.

Algorithm 1 BFRGD-Net Framework

Require: Preprocessed input features $\mathbf{X}$, true labels $\mathbf{Y}$, learning rate η, number of GNN layers L_g, number of Dense blocks L_d, batch size B, number of epochs E
Ensure: Predicted class probabilities $\hat{\mathbf{Y}}$
Step 1: Input Layer
Initialize the input layer with feature vector $\mathbf{X}$
Step 2: Bipolar Fuzzy Rough Sets (BFRS) Module
Calculate positive and negative membership functions for $\mathbf{X}$
Compute Bipolar Fuzzy Upper Approximation (BFUA) and Lower Approximation (BFLA)
Update input features based on BFUA and BFLA
Step 3: Graph Neural Network (GNN) Layers
for $l = 1$ to L_g **do**
    Perform graph convolution on node features using neighbors
    Apply degree normalization and ReLU activation
end for
Step 4: DenseNet Module
for $d = 1$ to L_d **do**
    Perform convolution, batch normalization, and ReLU activation in Dense Block d
    Concatenate output with input features for feature reuse
    if Transition Layer is required **then**
        Apply batch normalization, ReLU activation, and average pooling
    end if
end for
Step 5: Fully Connected Layer
Flatten the output from the DenseNet module
Pass through a dense layer with ReLU activation
Optionally apply dropout for regularization
Step 6: Output Layer
Use a final dense layer to map features to the number of classes
Apply the softmax activation function to get class probabilities $\hat{\mathbf{Y}}$
Step 7: Loss Function and Backpropagation
Compute the cross-entropy loss between $\mathbf{Y}$ and $\hat{\mathbf{Y}}$
Update model parameters using the Adam optimizer with learning rate η
Step 8: Training Loop
for epoch $= 1$ to E **do**
    for each batch of size B **do**
        Forward pass: Perform Steps 1 to 6
        Compute loss and perform backpropagation (Step 7)
        Update parameters
    end for
end for
Return Predicted class probabilities $\hat{\mathbf{Y}}$

G. Role of Bipolar Fuzzy Rough Sets in BFRGD-Net

The BFRGD-Net architecture relies on Bipolar Fuzzy Rough Sets (BFRS) to deal cybersecurity dataset uncertainty and ambiguity. BFRS clearly separates confidence from uncertainty by dividing the dataset into positive and negative areas. This method assures accurate categorization, even when data points overlap or are unclear.

The BFRS module refines data representations using BFUA and BFLA. BFUA finds the largest collection of class members, whereas BFLA finds the most specific. These estimates help the framework manage imprecise and partial data. The BFUA and BFLA are defined mathematically:

$$\text{BFUA}(Z) = \sup_{z \in U} \left(\min \left(\nu_1(z), 1 - \nu_2(z) \right) \right) \tag{25}$$

$$\text{BFLA}(Z) = \inf_{z \in U} \left(\max \left(\nu_1(z), 1 - \nu_2(z) \right) \right) \tag{26}$$

$\nu_1(z)$ and $\nu_2(z)$ are the membership functions for positive and negative classes, respectively, and Z is the dataset.

BFRS improves the model's unusual and ambiguous threat classification by using these approximations. Cybersecurity requires this capability because high-severity anomalies can resemble regular data. In addition to uncertainty managing, BFRS improves model decision-making interpretability, revealing threat identification and classification. In real-world applications where transparency is as crucial as accuracy, interpretability is a major benefit.

H. Performance Evaluation Metrics

To evaluate the BFRGD-Net model for cybersecurity threat detection, metrics like accuracy, precision, recall, and F1-score are used to evaluate correctness, true positives, and precision-recall balance [28]. However, cybersecurity requires identifying uncommon and significant threats, thus we offer three unique metrics: Threat Detection Sensitivity (TDS), Anomaly Impact Score (AIS), and Balanced Risk Detection Efficiency. Threat Detection Sensitivity (TDS) weights low-frequency, high-severity events that dataset imbalances ignore to assess the model's capacity to identify infrequent, high-impact threats. We compute TDS as:

$$\text{TDS} = \frac{\sum_{j=1}^{M} \left(v_j \cdot \frac{A_j}{A_j + B_j} \right)}{\sum_{j=1}^{M} v_j} \tag{27}$$

where v_j represents threat class weight based on severity and frequency, A_j represents true positives, and B_j represents false negatives. This statistic prioritizes infrequent but significant threats, boosting the model's sensitivity to high-risk cybersecurity events. The Anomaly Impact Score (AIS) calculates real-time detection system operating expenses for false positives and negatives. AIS weighs the real-world effect of false positives and missing detections against the advantages of true positives. We define AIS as:

$$\text{AIS} = \frac{\sum_{j=1}^{M} \left(\frac{A_j}{A_j + C_j + \rho \cdot B_j} \right)}{M} \tag{28}$$

ρ penalizes false negatives, especially in high-impact circumstances, whereas A_j represents genuine positives, C_j false positives, and B_j false negatives for class j. This enhances real positive detection while lowering false positives and undiscovered threats. Finally, Balanced Risk Detection Efficiency (BRDE) evaluates the model's real-time detection accuracy and operating efficiency. To respond quickly, cybersecurity models must effectively identify threats with minimal latency. These two elements are balanced in BRDE:

$$\text{BRDE} = \frac{\sum_{j=1}^{M} \left(\frac{A_j}{A_j + C_j + B_j} \cdot \frac{1}{1 + \theta \cdot D_j} \right)}{M} \tag{29}$$

where D_j is the detection time for class j and θ is a scaling factor that accounts for detection speed. This measure penalizes longer detection durations to keep the model efficient in real-time threat detection when speed and accuracy are critical. While accuracy, precision, recall, and F1-score give a broad evaluation of model performance, TDS, AIS, and BRDE provide key insights into the model's capacity to manage infrequent threats, balance operational effects, and retain efficiency. These new measurements address cybersecurity problems, making BFRGD-Net reliable and practical for real-world deployments where speed and accuracy are crucial.

IV. Simulation Results

To assess the proposed BFRGD-NeT framework, extensive simulations were run on a Dell Core i7 12th Gen system with an 8-core CPU and 32 GB RAM. Python and SPYDER IDE were used for simulation setup and execution. The framework has three essential modules, and hyperparameters for each module were carefully tweaked throughout studies to produce the best outcomes. To assure data quality, dynamic

parameters like the outlier filtering threshold were set to 0.15, and feature scaling factors were modified for each dataset in the preprocessing module. To balance feature relevance and redundancy, the feature selection module used Statistical-Driven Filtering and Adaptive Selector with 0.6 optimization weight. For the classification module, the Adam optimizer was used with a learning rate of 0.0005, batch size of 32, and dropout rates of 0.3 to avoid overfitting. The attention processes were also tuned to recognize complicated threat patterns. These setups improved detection accuracy and processing performance across circumstances.

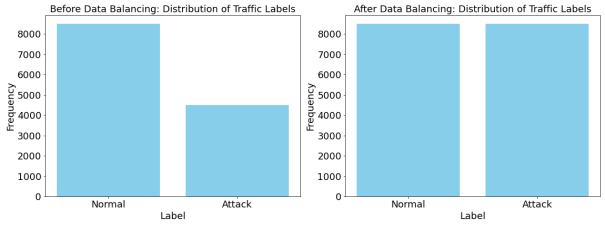

Fig. 3. Label distribution before and after data balancing.

Figure 3 illustrates cybersecurity dataset traffic label distribution before and after data balancing. On the left side of the graphic, the original data distribution shows a large imbalance between "Normal" and "Attack" labels, with 8500 normal traffic and 1500 attack traffic. Machine learning methods may bias forecasts toward typical traffic due to this imbalance. After balancing, the data distribution is shown on the right. This updated distribution equalizes "Normal" and "Attack" to 8500 occurrences each. The machine learning model will train on this balanced dataset, treating all groups equally, boosting attack detection, and lowering false negatives. This illustrates how data balancing improves cybersecurity model robustness and accuracy for recognizing normal and attack traffic by showing the before-and-after comparison.

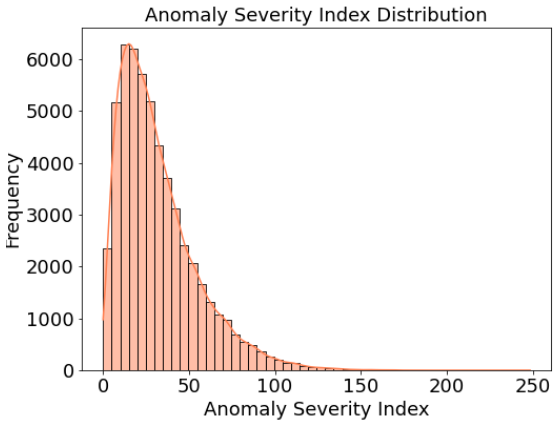

Fig. 4. Anomaly severity index distribution.

Figure 4 shows the distribution of the Anomaly Severity Index in the dataset, indicating the frequency of various severity levels. The Anomaly Severity Index is on the x-axis and frequency is on the y-axis. The histogram and KDE line show data distribution, with peaks suggesting shared severity. Right-skewed distributions indicate that lower-severity anomalies are more prevalent. The KDE line smoothes the probability distribution, simplifying data interpretation. This chart shows how successfully the system handles different threat levels.

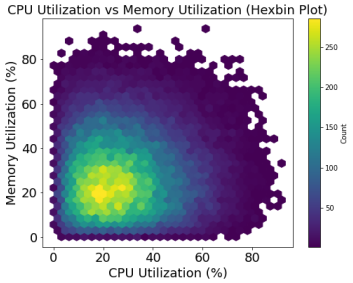

Fig. 5. CPU-Memory connection.

A hexbin plot in Figure 5 shows the dataset's CPU-Memory connection. Each hexagonal bin's color indicates the number of observations in that bin, representing data point density. Data points are denser in darker hues, whereas lighter colors indicate fewer observations. Analyzing the hexbin plot reveals CPU and memory consumption trends. Dark hexagon clusters may represent average CPU and memory use during system operation, whereas lighter hexagons may show outlier behaviors. This visualization helps uncover CPU and memory consumption correlations and unexpected or dispersed patterns that may suggest system abnormalities or performance issues. It helps analyze system performance and identify unexpected CPU and memory utilization patterns.

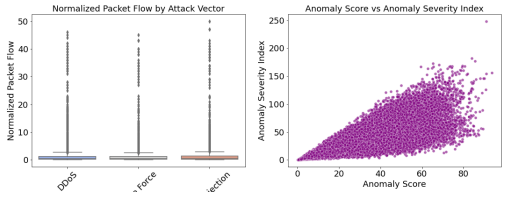

Fig. 6. Normalized packet flow by attack vector and anomaly score vs. severity index.

Figure 6 shows cybersecurity concerns from network traffic patterns and abnormalities. The left boxplot shows Normalized Packet Flow by Attack Vector, displaying packet flow rates by attack type. It displays the median, quartiles, and outliers for each attack vector to assist identify malicious from benign traffic behavior. This graphic shows which attack vectors substantially affect network traffic. The scatter figure on the right demonstrates how the Anomaly Score and Anomaly Severity Index correspond with threat severity. Higher anomaly scores indicate greater hazards, helping determine danger levels based on network activity.

A correlation heatmap of the cybersecurity dataset characteristics is shown in Figure 7. Each column represents the correlation coefficient between two attributes, ranging from -1 to 1, where darker hues indicate stronger linear relationships. Blue denotes negative correlations, while red indicates positive correlations. Values near 0 suggest no linear association, whereas values closer to ±1 indicate strong correlations. For example, packet size and flow bytes per second may exhibit significant positive correlations (close to 0.9), suggesting that

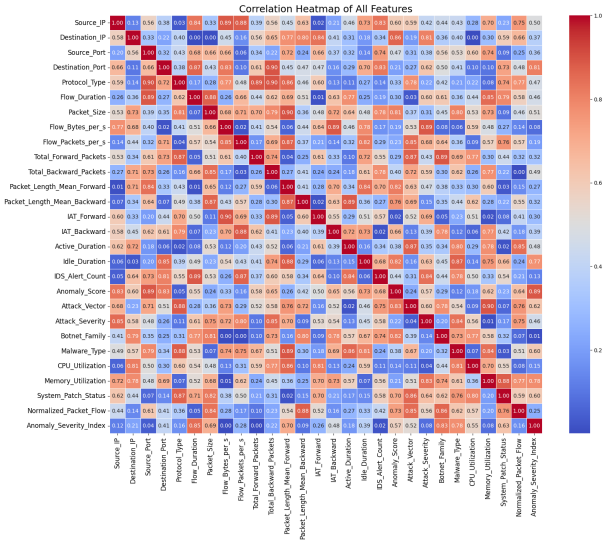

Fig. 7. Correlation heatmap of features.

larger packets carry more data. This heatmap helps identify patterns that guide feature selection and detect multicollinearity, aiding in feature engineering and model development.

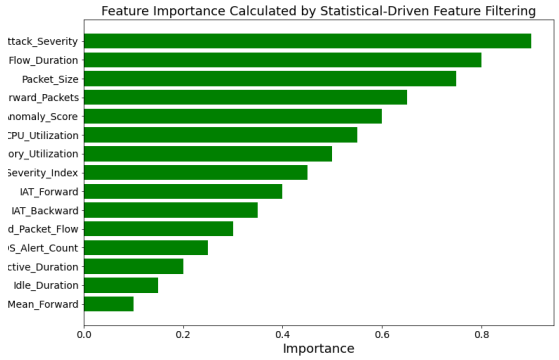

Fig. 8. Feature significance calculated by statistical-driven feature filtering.

Figure 8 displays the Statistical-Driven Feature Filtering estimates of feature significance. The bar plot scores 15 characteristics by relevance from 0.1 to 0.9. Important characteristics are at the top of the decreasing list. The graphic shows which characteristics classify cybersecurity risks in the dataset most. Features like Attack Severity and Flow Duration had higher significance values (0.9 and 0.8, respectively), suggesting they are significant in identifying normal from harmful activity. Packet Length Mean Forward and Idle Duration have lower significance values, indicating they contribute less to categorization. This figure also shows how feature importance affects the model's cybersecurity threat detection. This knowledge may assist choose features that improve model accuracy and reduce computing complexity.

Figures 9, 10, and 11 for the model's Threat Severity, Attack Type, and Cybersecurity Strategy Effectiveness Each figure's confusion matrix shows how effectively the model classifies jobs, with diagonal components indicating accurate classifications and off-diagonal parts not. Confusing "No

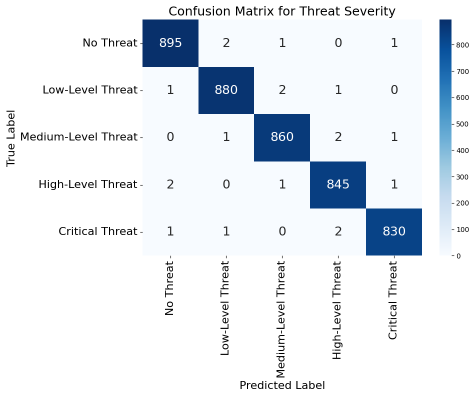

Fig. 9. Threat severity confusion matrix.

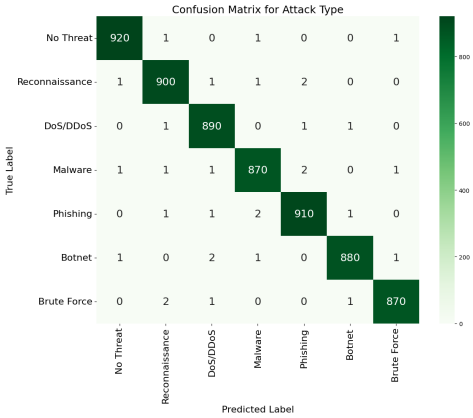

Fig. 10. Attack type identification confusion matrix.

Threat" to "Critical Threat." Most occurrences are likely categorized by the strong diagonal alignment. Few misclassifications occur around danger levels, showing the model can compare severity levels across categories. It can evaluate cybersecurity incident criticality. Figure 10 shows reconnaissance, DoS/DDoS, malware, phishing, botnet, plus brute force. The confusion matrix shows the model notices diagonal assaults. Comparing attack types is challenging since few misclassifications occur. The matrix proves the model properly classifies cyber dangers. Figure 11 displays the model's "No Action Required" to "Optimal Effectiveness." In the confusion matrix, most diagonal predictions are accurate across all effectiveness levels. Minimal off-diagonal values suggest the model seldom mixes categories, demonstrating cybersecurity assessment accuracy. Here are the model's cybersecurity risk, attack, and defense categories. The model's durability and real-time threat detection and response improve cybersecurity operational decision-making with little misclassification across all three activities.

Table III compares cybersecurity threat detection classification methods based on Accuracy, Log Loss, F1-Score, AUC, Recall, Precision, Balanced Precision Index (BPI), and Fault Detection Variability Coefficient. The table shows the efficacy of each strategy, with BFRGD-NeT winning all criteria. BFRGD-NeT has the greatest accuracy (97.8%) and F1-Score (97.5%), suggesting its robustness in determining threat levels. With the lowest Log Loss (0.071), the BFRGD-NeT model

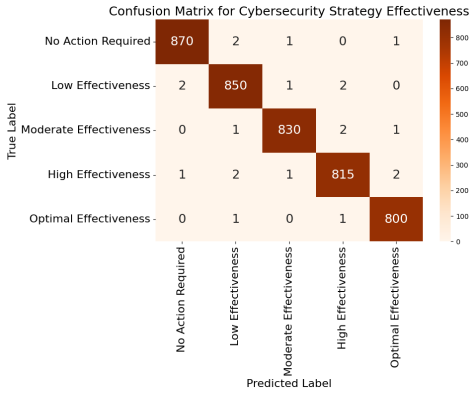

Fig. 11. Cybersecurity strategy effectiveness confusion matrix.

TABLE III. CLASSIFICATION RESULTS OF DIFFERENT TECHNIQUES

Techniques	Accuracy (%)	Log Loss	F1-Score (%)	AUC (%)	Recall (%)	Precision (%)	BPI (%)	FDVC (%)
CNN [14]	87.8	0.271	87.2	88.6	87.0	86.8	79.5	71.2
Decision Trees [16]	83.1	0.342	81.5	84.2	82.8	81.6	74.1	65.9
GNN [14]	90.1	0.255	89.4	90.8	89.0	88.7	82.9	74.6
DBN [21]	85.5	0.298	84.7	86.4	84.1	83.9	76.3	68.4
SVM [10]	85.1	0.309	83.8	85.9	83.3	82.7	76.0	67.0
BFRS [20]	86.5	0.285	85.1	87.3	85.4	84.8	78.0	69.8
KNN [11]	83.6	0.332	82.3	84.5	82.0	81.5	73.7	65.4
LSTM [18]	84.2	0.325	82.9	85.3	83.6	82.3	75.7	67.5
Proposed BFRGD-NeT	97.8	0.071	97.5	98.3	97.6	97.4	94.2	88.7

predicts more accurately than other techniques. Other models like GNN and CNN perform well but fall short of the proposed strategy, notably in BPI and FDVC, which imply balanced prediction accuracy and fault detection consistency. Traditional Decision Trees and KNN have poorer accuracy, F1-Score, and AUC values, demonstrating they cannot handle complicated cybersecurity threat data. The table shows that the BFRGD-NeT model is best for real-time threat detection due to its excellent classification performance.

TABLE IV. STATISTICAL ANALYSIS OF CLASSIFICATION METHODS (F-STATISTIC & P-VALUE)

Statistical Method	ANOVA	Student's T-test	Pearson Correlation (r)	Kendall's Tau (τ)	Chi-Square (χ^2)
CNN [14]	6.98	0.020	0.88	0.75	7.92
Decision Trees [16]	4.89	0.043	0.61	0.57	6.18
GNN [14]	7.56	0.014	0.84	0.72	8.63
DBN [21]	6.45	0.017	0.78	0.71	7.45
SVM [10]	5.67	0.029	0.70	0.64	6.88
BFRS [20]	7.12	0.021	0.81	0.69	7.58
KNN [11]	5.12	0.036	0.62	0.58	6.33
LSTM [18]	5.22	0.031	0.65	0.59	6.54
Proposed BFRGD-NeT	**8.93**	**0.007**	**0.94**	**0.81**	**9.92**

Table IV compares cybersecurity threat detection categorization approaches using statistical tests including ANOVA, Student's T-test, Pearson Correlation (r), Kendall's Tau (τ), and Chi-Square (χ^2). Decision Trees, KNN, SVM, CNN, GNN, and DBN perform differently, as seen in the table. The BFRGD-NeT technique classifies cybersecurity risks with the greatest statistical values across all parameters, demonstrating its consistency, correlation, and robustness. ANOVA and Chi-Square scores for BFRGD-NeT are greater than other approaches, indicating a more statistically significant difference between predictions. BFRGD-NeT's Pearson Correlation (0.94) and Kendall's Tau (0.81) reflect greater correlations and rank correlation with outcomes, improving prediction. Decision Trees and KNN have poorer statistical results, indicating their inability to handle complicated cybersecurity data. Table IV shows that BFRGD-NeT produces more accurate threat categorization than standard and deep learning approaches.

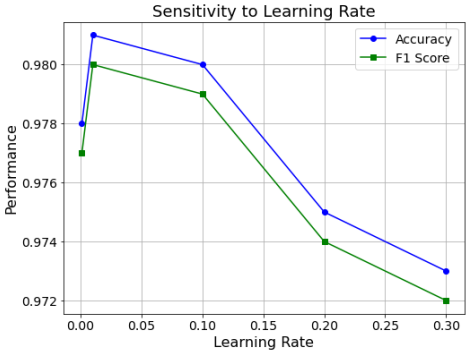

Fig. 12. Sensitivity to learning rate.

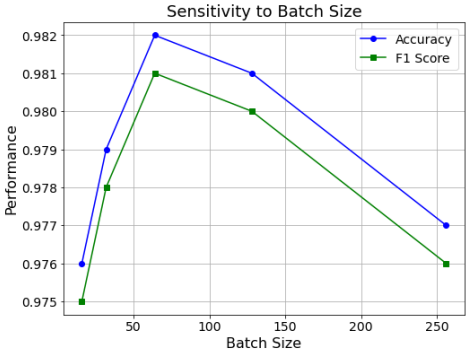

Fig. 13. Sensitivity to batch size.

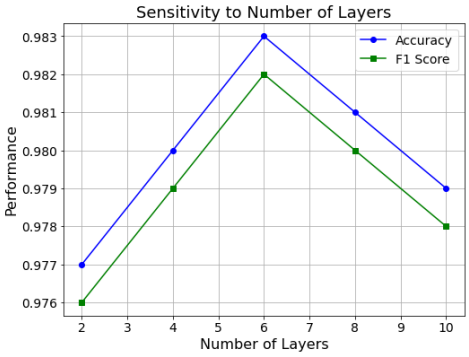

Fig. 14. Sensitivity to number of layers.

This model's sensitivity analysis shows how learning rate, batch size, and layer count impact its performance in Figure 12, 13 and 14. The figure shows the sensitivity to learning rate, batch size, and number of layers, with accuracy and F1-score displayed in each subplot. In the plots, the model regu-

larly achieves excellent accuracy and F1-scores close to 98%, however hyperparameters vary somewhat. These variations show that appropriate learning rates or batch sizes improve outcomes, helping to optimize the model.

V. Conclusion

In cybersecurity threat detection, the BFRGD-Net framework outperforms standard models in accuracy, F1-score, and AUC. It captures local interdependence and global traffic patterns to solve difficult cybersecurity data problems using BFRS, GNN, and DenseNet. A comprehensive hybrid feature selection approach and sophisticated preprocessing methods like HTS and CBSA have improved data quality and feature relevance, improving classification performance. Novel metrics TDS, AIS, and BRDE show how the system can effectively detect uncommon, high-severity threats in real time while retaining operational efficiency. Statistical research shows the model's consistency and resilience, making it suitable for cybersecurity situations that need accurate and quick threat detection. The findings are encouraging, but further study is required to improve the framework. Data sources, adjust the model for ICS and IoT networks and optimize hyperparameters to enhance performance. This work enhances threat detection and develops scalable, adaptive algorithms to react to the quickly changing cyber threat environment, enabling more proactive and robust security systems.

For cybersecurity dataset uncertainty management, Bipolar Fuzzy Rough Sets (BFRS) in the BFRGD-Net architecture are effective. Future work will include dynamic thresholds for positive and negative areas and investigate context-aware modifications to BFRS. These improvements improve the model's capacity to distinguish complex, dynamic threat patterns, making it more applicable in real-world cybersecurity settings.

VI. Practical Implications of Theoretical Results

Real-world cybersecurity applications benefit from BFRGD-Net framework theoretical findings. The model is ideal for dynamic and high-risk contexts because it can manage skewed datasets, identify infrequent but crucial threats, and adapt to changing assault patterns.

- **Critical Infrastructure Protection:** The framework protects electricity grids, healthcare systems, and transportation networks against undiscovered cybersecurity attacks, which might have serious effects.
- **Proactive Threat Mitigation:** This research offered unique metrics including Threat Detection Sensitivity (TDS) and Anomaly Impact Score (AIS) to prioritize and mitigate cybersecurity threats. These measurements help organisations prioritise the biggest dangers and allocate resources more strategically.
- **Scalable, Real-Time Performance:** BFRS, GNNs, and DenseNet are used in BFRGD-Net's architecture to ensure computational efficiency and scalability, making it suitable for real-time systems like industrial control systems (ICS) and IoT networks.

The suggested framework may improve cybersecurity measures in many applications by combining theoretical and practical contributions. These practical advantages demonstrate the theoretical conclusions' relevance and application to cybersecurity issues.

References

[1] M. A. I. Mallick and R. Nath, *Navigating the cybersecurity landscape: A comprehensive review of cyber-attacks, emerging trends, and recent developments*, World Scientific News, vol. 190, no. 1, pp. 1-69, 2024.

[2] S. Razaulla, C. Fachkha, C. Markarian, A. Gawanmeh, W. Mansoor, B. C. Fung, and C. Assi, *The age of ransomware: A survey on the evolution, taxonomy, and research directions*, IEEE Access, vol. 11, pp. 40698-40723, 2023.

[3] S. Abdelkader, J. Amissah, S. Kinga, G. Mugerwa, E. Emmanuel, D. E. A. Mansour, *Securing modern power systems: Implementing comprehensive strategies to enhance resilience and reliability against cyber-attacks*, Results in Engineering, p. 102647, 2024.

[4] F. N. U. Jimmy, *Cybersecurity vulnerabilities and remediation through cloud security tools*, Journal of Artificial Intelligence General Science (JAIGS), vol. 2, no. 1, pp. 129-171, 2024.

[5] H. J. Ofte and S. Katsikas, *Understanding situation awareness in SOCs, a systematic literature review*, Computers & Security, vol. 126, p. 103069, 2023.

[6] A. A. Almazroi, F. S. Alsubaei, N. Ayub, and N. Z. Jhanjhi, *Inclusive smart cities: IoT-cloud solutions for enhanced energy analytics and safety*, International Journal of Advanced Computer Science & Applications, vol. 15, no. 5, 2024.

[7] S. Chaudhary, V. Gkioulos, and S. Katsikas, *Developing metrics to assess the effectiveness of cybersecurity awareness program*, Journal of Cybersecurity, vol. 8, no. 1, p. tyac006, 2022.

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

[9] M. Thakur, *Cybersecurity threats and countermeasures in the digital age*, Journal of Applied Science and Education (JASE), vol. 4, no. 1, pp. 1-20, 2024.

[10] M. Aslam, D. Ye, A. Tariq, M. Asad, M. Hanif, D. Ndzi, and S. F. Jilani, *Adaptive machine learning-based distributed denial-of-service attacks detection and mitigation system for SDN-enabled IoT*, Sensors, vol. 22, no. 7, p. 2697, 2022.

[11] J. Bhayo, S. A. Shah, S. Hameed, A. Ahmed, J. Nasir, and D. Draheim, *Towards a machine learning-based framework for DDoS attack detection in software-defined IoT (SD-IoT) networks*, Engineering Applications of Artificial Intelligence, vol. 123, p. 106432, 2023.

[12] M. Memarzadeh, B. Matthews, and T. Templin, *Multiclass anomaly detection in flight data using semi-supervised explainable deep learning model*, Journal of Aerospace Information Systems, vol. 19, no. 2, pp. 83-97, 2022.

[13] A. Sahu, Y. A. B. El-Ebiary, K. A. Saravanan, K. Thilagam, G. R. Devi, A. Gopi, and A. I. Taloba, *Federated LSTM model for enhanced anomaly detection in cybersecurity: A novel approach for distributed threat*, International Journal of Advanced Computer Science & Applications, vol. 15, no. 6, 2024.

[14] M. Ozkan-Ozay, E. Akin, Ö. Aslan, S. Kosunalp, T. Iliev, I. Stoyanov, and I. Beloev, *A comprehensive survey: Evaluating the efficiency of artificial intelligence and machine learning techniques on cybersecurity solutions*, IEEE Access, 2024.

[15] M. Basnet and M. H. Ali, *A deep learning perspective on connected automated vehicle (CAV) cybersecurity and threat intelligence*, in Deep Learning and Its Applications for Vehicle Networks, CRC Press, pp. 39-56, 2023.

[16] J. Axali, L. Devereaux, A. Spencer, and F. Vasilev, *A multicriteria decision-making approach for ransomware detection using MITRE ATT&CK mitigation strategy*, Authorea Preprints, 2024.

[17] S. Lin, C. Feng, T. Jiang, and H. Jing, *Evaluation of network security grade protection combined with deep learning for intrusion detection*, IEEE Access, vol. 11, pp. 130990-131000, 2023.

[18] D. Javaheri, S. Gorgin, J. A. Lee, and M. Masdari, *Fuzzy logic-based DDoS attacks and network traffic anomaly detection methods: Classification, overview, and future perspectives*, Information Sciences, vol. 626, pp. 315-338, 2023.

[19] T. A. S. Srinivas, G. Mahalaxmi, A. D. Donald, and R. Varaprasad, *Traffic prediction using a wide range of techniques: A review*, IUP Journal of Information Technology, vol. 19, no. 1, pp. 19-47, 2023.

[20] Z. Zhao, A. Hussain, N. Zhang, K. Ullah, S. Yin, A. Awsar, and S. M. El-Bahy, *Decision support system based on bipolar complex fuzzy Hamy mean operators*, Heliyon, vol. 10, no. 17, 2024.

[21] T. Mahmood, U. U. Rehman, S. Shahab, Z. Ali, and M. Anjum, *Decision-making by using TOPSIS techniques in the framework of bipolar complex intuitionistic fuzzy N-soft sets*, IEEE Access, vol. 11, pp. 105677-105697, 2023.

[22] D. B. Chakraborty and J. Yao, *Event prediction with rough-fuzzy sets*, Pattern Analysis and Applications, vol. 26, no. 2, pp. 691-701, 2023.

[23] Z. Tan, *Cybersecurity threat and awareness program dataset [Data set]*, Kaggle, https://doi.org/10.34740/KAGGLE/DSV/9665651, 2024.

[24] V. Werner de Vargas, J. A. Schneider Aranda, R. dos Santos Costa, P. R. da Silva Pereira, and J. L. Victória Barbosa, *Imbalanced data preprocessing techniques for machine learning: A systematic mapping study*, Knowledge and Information Systems, vol. 65, no. 1, pp. 31-57, 2023.

[25] A. A. Almazroi and N. Ayub, *Enhancing smart IoT malware detection: A GhostNet-based hybrid approach*, Systems, vol. 11, no. 11, p. 547, 2023.

[26] T. Verdonck, B. Baesens, M. Óskarsdóttir, and S. vanden Broucke, *Special issue on feature engineering editorial*, Machine Learning, vol. 113, no. 7, pp. 3917-3928, 2024.

[27] N. Malik, M. Shabir, T. M. Al-shami, R. Gul, and M. Arar, *A novel decision-making technique based on T-rough bipolar fuzzy sets*, Journal of Mathematics and Computer Science, vol. 33, pp. 275-289, 2024.

[28] M. Z. Naser and A. H. Alavi, *Error metrics and performance fitness indicators for artificial intelligence and machine learning in engineering and sciences*, Architecture, Structures and Construction, vol. 3, no. 4, pp. 499-517, 2023.