[1]Jasper J
[2] Praveen B.M
[3] Berlin Shaheema

XAI Enhanced GCNN-HSA Framework for Anomaly Detection in Smart Grids

Journal of Electrical Systems

Abstract: - The integration of digital technologies enhances smart grid connectivity, dependability, and sustainability, but their growing complexity necessitates advanced, intelligent anomaly detection for secure operation. This study proposes a novel hybrid framework combining Graph Convolutional Neural Networks (GCNN) with the Harmony Search Algorithm (HSA) for robust anomaly detection in smart grids. HSA optimizes GCNN hyper parameters, significantly boosting detection accuracy and responsiveness. A key innovation is the integration of Explainable Artificial Intelligence (XAI) techniques, specifically SHAP and Grad-CAM, to render the model's decision-making transparent and interpretable. This allows stakeholders, including operators and analysts, to better understand, validate, and trust the model's predictions. Experimental evaluations on the IEC 60870-5-104 and public cyberattack datasets confirm the proposed GCNN-HSA framework's superior performance in accuracy, precision, recall, F1-score, and AUROC compared to conventional methods. The XAI components further enhance system usability and accountability. This research contributes a novel, high-performance, and inherently explainable anomaly detection framework, addressing both technical efficacy and operational transparency to foster more secure, reliable, and interpretable smart grid infrastructures.

Keywords: Smart Grid, Intrusion Detection System, Anomaly Detection, Cyber Attacks, Cyber Security, Graph Convolution Neural Network, Deep Graph Convolution Neural Network, Harmony Search Algorithm.

I. INTRODUCTION

The fusion of traditional power grid infrastructure with communication technology has led to the next-generation smart grid, enabling real-time information exchange between the grid and end-users via smart meters. These grids aim to modernize the power system by enhancing reliability, resilience, and efficiency. To fully realize these benefits, advanced energy harvesting and management strategies, including the use of smart meters and consumer applications, have been implemented [1]. Smart grids enable features such as adaptive power generation, self-healing capabilities, efficient energy use, and improved power quality. However, the incorporation of such technologies introduces significant cybersecurity risks that threaten the grid's integrity and availability [2].

The smart grid comprises four main components: production, distribution, transmission, and consumption, as visualized in Figure 1. Smart grids consist of four key components: generation, transmission, distribution, and consumption, interconnected through hierarchical communication networks—WAN, NAN, and HAN [3]. These networks facilitate real-time data exchange and operational control. The smart grid ecosystem is supported by sophisticated devices, such as PMUs, smart meters, and automated substation relays, which enable precise grid monitoring and enhance fault tolerance. Despite their advantages, the increasing integration of smart technologies creates a large attack surface, making them vulnerable to cyber threats such as data manipulation, spoofing, and denial-of-service (DoS) attacks [4]. In such a scenario, early and accurate anomaly detection becomes essential to ensuring system resilience.

Explainable Artificial Intelligence (XAI) has emerged as a crucial component in the deployment of machine learning models in critical systems like smart grids. While many AI-based Intrusion Detection Systems (IDS) offer high detection accuracy, they often function as "black boxes" with limited transparency. XAI addresses this limitation by providing insights into the decision-making process of models, making them interpretable and trustworthy. In smart grid applications, XAI can help system operators understand why a particular event is classified as anomalous, thereby improving response strategies, building trust, and facilitating compliance with safety and regulatory standards.

An IDS is a core cybersecurity mechanism that monitors systems for unusual or unauthorized activities [5]. These systems can be configured based on signatures, anomalies, or specifications, and hybrid models can combine these strategies for enhanced performance. AI-driven anomaly-based IDS models are increasingly used

[1] *Corresponding author Post-Doctoral Fellow, Institute of Engineering and Technology, Srinivas University, Mangaluru, 574146, Karnataka, India
[2] Director Research, Institute of Engineering and Technology, Srinivas University, Mangaluru, 574146, Karnataka, India
[3] National Institute of Technology, Silchar, 788010, Assam, India

due to their ability to identify unknown threats by learning behavioral patterns [6]. Yet, the success of these systems heavily depends on optimal feature selection and hyper parameter tuning, which can affect detection accuracy and computational efficiency. In response to these challenges, our work proposes a hybrid GCNN-HSA framework for anomaly detection in smart grids. The Graph Convolutional Neural Network (GCNN) excels in capturing spatial and relational patterns from the grid topology, while the Harmony Search Algorithm (HSA) is employed for efficient hyperparameter optimization. In addition, XAI techniques are integrated into the framework to generate transparent and interpretable model decisions, allowing operators to trace back the logic behind anomaly detection outcomes.

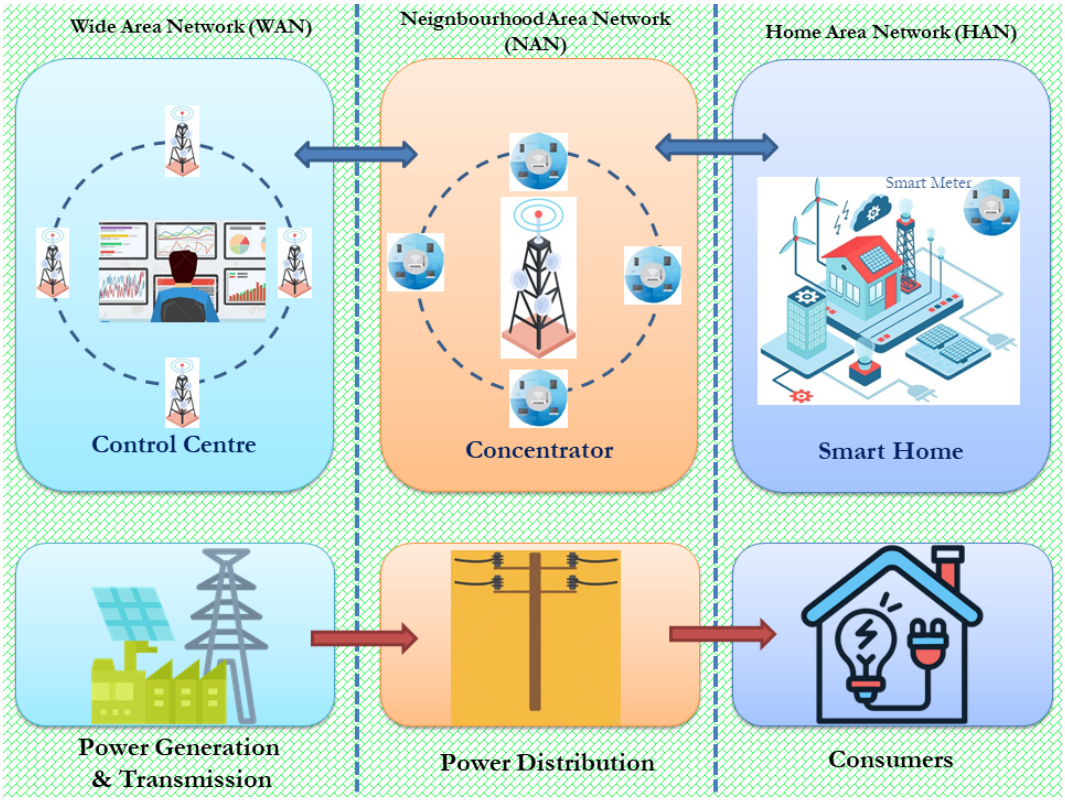


Figure 1: The Smart Grid's Architectural Design

The primary contributions of this study are summarized as follows.

- Development of a GCNN-HSA-based intrusion detection framework tailored for smart grid anomaly detection.
- Integration of XAI modules to interpret anomaly predictions, increasing the model's transparency and operational trust.
- A lightweight IDS architecture that reduces computational overhead through optimized feature selection.
- Comparative performance analysis with state-of-the-art models, demonstrating improvements in accuracy, interpretability, and reliability.

The remaining portions of the text are arranged as follows: Section 2 provides illustrations of the relevant material. Section 3 deals with the cyber-attack data description utilized in the proposed work. The proposed work is presented in Section 4. The experiments and their findings are presented in Section 5. Section 6, concludes with a discussion of a few potential areas for future investigation.

II. Related Works

Numerous research has been done to try and solve this issue, such methods have been covered in this section. The system's goal is to recognize suspicious activity and cybersecurity concerns by utilizing recurrent neural networks (RNNs) [7]. DL and block chain technology are used by Fortitude to enhance security [8]. One shortcoming is the lack of details about performance metrics and how they compare to other strategies. Machine learning (ML) [9] strategies to raise the smart grids' efficiency, security, stability, and responsiveness. The study

concentrated on tackling issues such as the lack of labeled data, changing attack patterns, and the creation of high-resolution synthetic data. Researchers [10] presented hardware hacking outcomes and hardening approaches to fend off assaults and lessen susceptibility. A deep reinforcement learning (RL) recovery technique leads to the optimal power system line following cyberattacks [11]. The suggested approach exhibits efficacy in reducing the effects of cyberattacks in a range of circumstances and can adjust to unpredictable attack scenarios.

A DL approach for identifying delay network assaults utilizing hybrid Convolutional Neural Network-Long-Short Term Memory (CNN-LSTM) models [12]. A performance rate higher than 99% was observed in the experimental data which also uses DL to detect delay network attacks. The Multicast LSTM [13] to forecast the stability of smart grids, also combined physical systems and information technology infrastructure, comparing the outcomes with existing DL techniques. National Institute of Standards and Technology (NIST) typical conceptual model analysis of smart grid domains for three key block chain characteristics: decentralization, incentive, and trust [14]. A comprehensive overview of security alternatives for smart grid systems with fog-based edge-enabled intrusion detection [15], [16]. Cyberattacks in energy systems were successfully distinguished from frequent events by the ML-based technique [17]. The body of research on AI approaches for power systems and smart grid security issues, fault detection, load forecasting, and grid stability assessment [18]. The study emphasized how AI could enhance the robustness and dependability of smart grid methods for grid stability, fault detection, and load forecasting [19].

Conventional methods use PMU to estimate the power system's state and compare the estimated readings with a threshold for detecting cyberattacks [20],[21]. The cosine similarity matching and chi-square detection methods for spotting cyberattacks on smart grids [22]. ML has been a popular tool for cyberattack detection in recent years [23], [24]. Several supervised learning methods were investigated to distinguish between cyberattacks and power system disruptions. To detect FDI assaults, [25] employed feature-level fusion and ensemble learning in conjunction with several well-known supervised algorithms. An Ad boost-based classification model [26] for power system disruptions and cyberattacks identification, utilizing individual PMU data. They used weight voting in conjunction with classification models and feature construction to generate new features from PMU for final detection.

The IEEE-designed European Low Voltage System has been the target of some models and simulations of assaults. Simulation studies indicate that these kinds of attacks could result in blackouts across the European Union. Power system equipment security is just as vital as smart meter security. A method for identifying cyberattacks based on network traffic self-similarity is developed in [27]. The IEC 61850 standard's GOOSE messages are used for power-system protection, and smart grids depend heavily on their dependable transmission. To address this, [28] develops an anomaly detection technique to identify Denial of Service (DoS) attacks against GOOSE network communication.

Attacks such as False Data Injection (FDI) are acknowledged as posing a serious risk to smart-grid functionality. A comprehensive analysis of FDI assaults, their effects on the various tiers of smart grid operation, and available mitigation strategies is given in [29]. A two-tier smart-grid architecture [30] to safeguard smart grids measurements after realizing how important it is to mitigate FDI assaults. For security, elliptic curve cryptography is used. Cybersecurity research necessitates projects involving hardware demonstrations and lab implementation in addition to theoretical study [31]. Investigations are conducted on how cyberattacks affect grid-connected storage devices and how they affect the electricity system.

It's critical to identify anomalies or attacks in smart grids. [32] Uses an LSTM architecture in conjunction with a CNN to identify electricity theft. A novel anonymous and secure metering technique was created by researchers in [33]. This is necessary to address privacy concerns related to high-resolution data that smart meters acquire. Identity-based signatures and direct anonymous attestation form the foundation of the new method. The detection methods are categorized as: anomaly-based, stateful-based protocol analysis, and signature-based. Bad patterns are identified by signature based on past data. Anomaly techniques identify irregularities by detecting deviations from network traffic. An alternative to anomaly-based detection is stateful-based protocol analysis. Smart grid infrastructure, IDS is thought to be one of the primary methods for spotting cyberattacks [34], [35]. Effective detection of unknown or zero-day threats is one of these systems' key advantages [36], [37]. IDS has been suggested in many research to identify cyberattacks. Furthermore, some efforts focused mainly on enhancing signature-based IDS. A deadly detection IDS system [38]. The suggested system combines IDS features with the Cumulative Sum (CUSUM) method.

In conclusion, the literature study addresses a variety of approaches, including the use of AI techniques to address various issues in smart grid security, as well as DL and block chain-based security solutions. According

to the assessments above, these techniques have many shortcomings, such as the inability to stop a slow attacker near the attack's origin. This research presents a GCNN-based technique to solve these shortcomings. Anomaly detection is a tool IDS and other cybersecurity technologies use to help detect abnormal or suspect user behavior or network traffic patterns. These patterns may point to possible security concerns or assaults, such as malware infections or unauthorized access.

III. Cyber Attack Data Description in Power System Framework

This study utilized two different datasets from a framework for a power system made up of network monitoring devices, supervisory control systems, and smart devices for power system monitoring, control, and other related communications to automate electric power systems as shown in Figure 2.

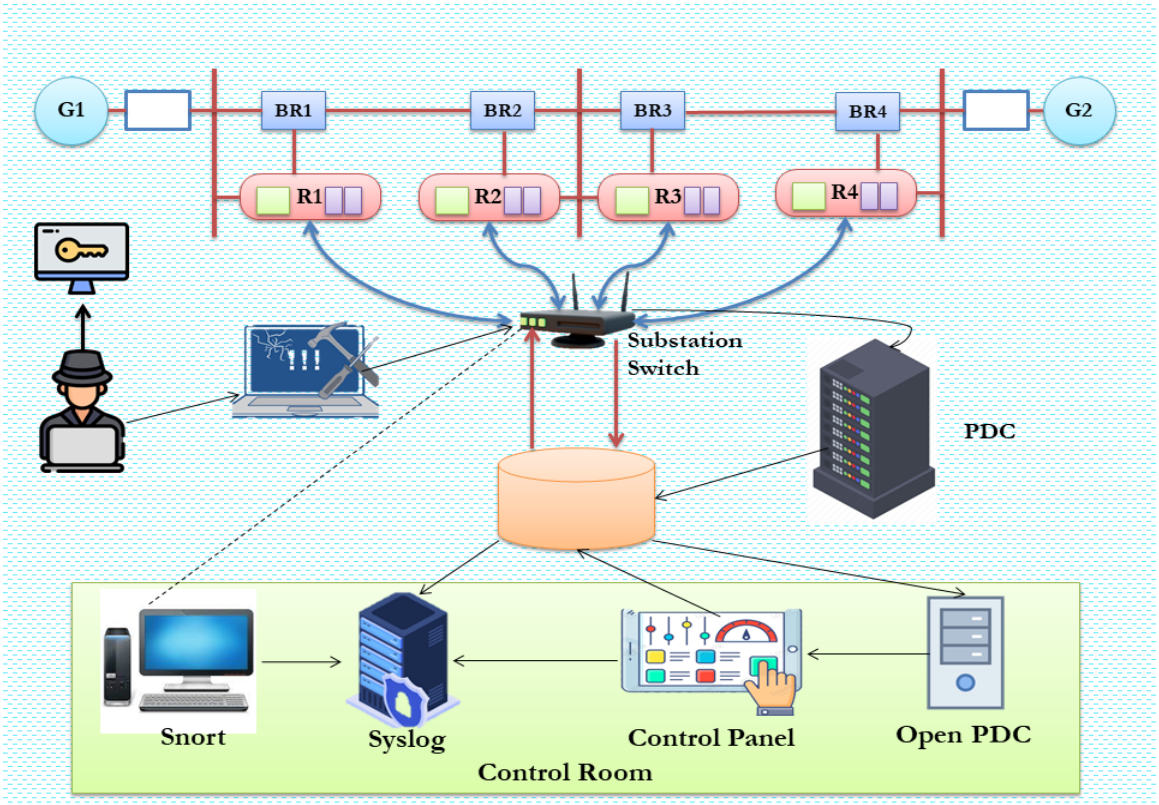


Figure 2: Power System Framework

Table 1 Power Systems operational situations and events-Test System-1

Scenario No	Event Description	Type
1–6	Short-circuit fault	Natural
13, 14	Line maintenance	Natural
7–12	Data injection	Attack
15–20	Remote tripping command injection	Attack
21–30, 35–40	Relay setting change	Attack
41	Normal Measurements.	Normal Condition

Table 2: Power Systems operational situations and events- Test System-2

Scenario No	Event Description	Type
1	Denial of Service	Attack
2	Switching Attack	Attack
3	Injection Command	Attack
4	Connection Lost	Attack

Scenario No	Event Description	Type
5	Rogue Device	Attack

The public data available at [39] was used for the experiments, the data was accessed on June 3, 2024. Subsequently, the public was able to access the IEC 60870-5-104 (IEC104) control communication protocol [40] and the datasets for the study are available [41]. Four smart electronic gadgets and two power generators (G1 and G2) (IEDs), designated R1 through R4, make up the initial test system. Four breakers (BR1 to BR4) are controlled by these IEDs, and two transmission lines (L1 and L2) link the breakers. Because of its distance protection system, the IEDs can automatically flip breakers in the event of a malfunction. Nevertheless, since the IEDs lack internal validation, these breakers can be changed regardless of whether the malfunction is the result of an attack or a natural abnormality. For maintenance reasons, operators can also manually swap the IEDs. By modeling various operating scenarios, the power system framework may also produce data about three different kinds of events: attack events, natural events, and no events.

In SCADA networks, the second dataset is frequently used for managing vital infrastructure, including power plants. In smart grid settings, the IEC 104 protocol is widely used to transfer data monitoring and control across several systems and devices. This protocol makes it possible for communication between SCADA systems, sensors, actuators, controllers, and other power grid components. The well-known IEC 60870-5-104 dataset, includes a variety of malicious attack types as Denial of Service, Switching, Injection Command, Connection Loss, and Rogue Device, to assess the efficiency of the proposed framework.

IV. Proposed GCNN-HSA-XAI Model

This section delineates the limitations observed in existing deep learning (DL)-based Intrusion Detection System (IDS) research and introduces the proposed GCNN-HSA-XAI model, which not only enhances detection capabilities but also incorporates explain ability into the decision-making process. The primary objective of this research is to identify and utilize optimal features that significantly improve the performance of IDS in terms of accuracy, precision, and interpretability.

The proposed model leverages a Graph Convolutional Neural Network (GCNN) to effectively process structured smart grid data, capturing spatial and topological patterns that are indicative of cyber anomalies. To further refine the model's performance, the Harmony Search Algorithm (HSA) is used for hyper parameter optimization, enabling efficient tuning of GCNN layers, learning rates, and other network parameters. To address the black-box nature of deep learning systems and improve transparency, the model integrates Explainable AI (XAI) components. Specifically, techniques such as graph node attribution and feature importance ranking are employed to provide human-interpretable insights into the model's predictions. These XAI methods help explain why a certain anomaly is detected, which features contributed most to the decision, and how the graph structure influences detection outcomes.

This GCNN-HSA-XAI fusion allows the system to:

- Detect attacks in smart grid systems with high precision and minimal false positives.
- Adapt effectively to diverse attack types through optimized learning.
- Provide transparent, traceable, and interpretable justifications for each anomaly detection decision, fostering trust and confidence in the IDS's outputs.

By combining the structural learning power of GCNN, the optimization efficiency of HSA, and the interpretability brought by XAI, the proposed model not only detects intrusions effectively but also explains them a critical capability for securing cyber-physical infrastructure like smart grids.

A. Dataset Description

Table 1 presents details of 37 simulated operational scenarios along with the different types of events associated with the first data set. There are six types of events, and they are explained below:

1. Normal Condition: Normal readings & Measurements.

2. Short-circuit fault: By examining the data's percentage range, it is possible to determine that there has only been one line-to-ground fault.

3. Line maintenance: Operators turn on and off one or more IEDs to carry out maintenance on specific sections of the electrical system and its components.

4. Command injection attacks on remote tripping: If an attacker manages to get access to the system, they can transmit commands that alter the status of IEDs and control the switch breakers.

5. Attacks on Data injection: Attackers alter settings, such as deactivating key functions, which prevent the IEDs from activating.

6. Data injection attack: Attackers manipulate the PMU measurements, to imitate a legitimate fault and trigger the tripping of breakers [3]

Table 2, presents the operational scenarios and the different types of events associated with the second data set. There are five types of events and they are explained below.

1. Denial of Service (DoS): Unauthorized access is gained by the attacker using a fake IP address to flood the victim with a large volume of messages, overloading and disrupt its functionality, potentially leading to grid failure.

2. Switching Attack: The attacker's goal is to manipulate the operational status of the station to disrupt its functionality, it may have detrimental effects on the stability and general operation of the grid.

3. Injection Command: The attacker wants to undermine the targeted device's functioning and integrity to possibly disrupt or harm the system as a whole. The attacker tampers with the system by altering configurations or sending fake commands to a device that is connected. This causes a variety of irregularities.

4. Connection Lost: The purpose of these assaults is to interfere with the targeted device's regular operation, possibly leading to delays, malfunctions, or data loss. The attacker tries to tamper with particular devices in an attempt to break their associated communications.

5. Rogue Device: Access to the communication network is obtained by the attacker without authorization, the attacker can force authorized devices to carry out unauthorized actions and transmit arbitrary messages, which could result in unpredictable events and potentially dangerous consequences.

B. Data Processing and Training Model

To implement an ML-based anomaly detection system, it is crucial to properly prepare the data. This research gathered data from four PMUs with integrated relays, resulting in a total of 116 features. Data was gathered from several PMU sources spread throughout the proposed power system framework, resulting in a significant volume of data exceeding several terabytes. Cleaning was done as part of the data selection process to get rid of noise, missing data, and outliers. To make sure the supplied data was reliable and consistent, data validation was done. After that, the information was included in a real-time data platform, which made it possible to receive and process data streams from PMU devices instantly. To guarantee the quality and consistency of the gathered data for anomaly detection, cleaning, and preparation phases were applied.

With the increasing complexity of anomaly detection methods, there is a growing need for interpretable models that can elucidate the rationale behind anomaly detection. Explainable AI techniques can enhance decision-making and foster greater trust. Specific hyper parameters were used to train each algorithm, and these adjustments were made iteratively as the algorithm was being trained. Each layer of the CCNN's activation function, number of hidden layers, and units were changed to correspond with the complexity of the temporal data sequences. The chosen method was refined via painstaking training with adjusted hyper parameters. The models had to be trained and validated on various combinations of subsets, and the overall performance of the model had to be evaluated by averaging the results. Furthermore, off-validation was carried out by sampling the validation set to guarantee the models' performance on data that had not been observed before. Verifying that the models could successfully detect anomalies and generalize to real-world events was crucial. The stratified dataset configuration and algorithm-specific hyper parameter optimization were taken into consideration during the meticulous execution of the training and validation phases. These crucial phases made sure the models were accurate in assessing their performance in identifying anomalies in crucial smart grid infrastructure and that they were ready for evaluation on the validation set and properly calibrated.

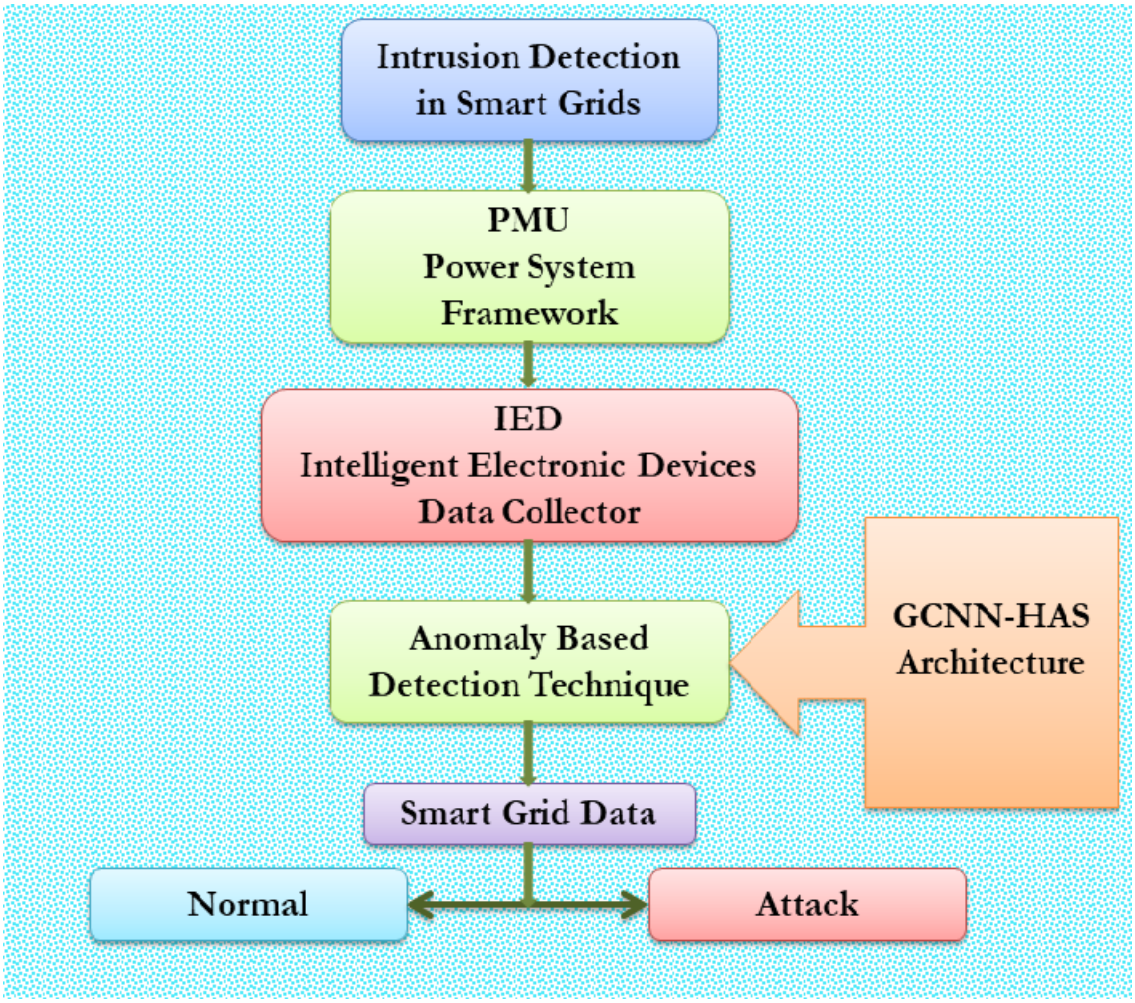


Figure 3: Flow diagram of the proposed method

C. Deep Graph Convolutional Neural Network (DGCNN) Model

DGCNNs, also known as Deep GCNN, have demonstrated effectiveness in intrusion detection, particularly in safeguarding SG. Smart Grid systems are progressively becoming more interconnected, rendering them susceptible to different online dangers, like malevolent assaults. Strong IDS are necessary to recognize and neutralize these dangers. DGCNN is utilized for anomaly detection and consists of four components:

1. The high-dimensional attributes of the input nodes and the rich structure information of the input graph are extracted by the first stage's sixteen-layer graph convolution layers.

2. To capture deeper structure information and node attributes and produce a consistent vertex ordering, the second-stage graph convolutional layers (16 layers) blend the high-dimensional node features from the first-stage graph convolutional layers with the initial low-dimensional data.

3. To standardize the number of nodes as the input for the next stage, Sort Pooling arranges the node characteristics generated by the second-stage graph convolution layers.

4. The sorted continuous node features are then used to predict graph attributes with dense and one-dimensional convolution layers.

Figure 3 shows an illustration of the flow diagram for the suggested strategy. Combining GCNN-HSA improves the anomaly detection model's accuracy, adaptability, and robustness, which in turn improves the security and dependability of intelligent power distribution systems [42]. The main goal of the model is to manage vast and complicated volumes of data while effectively identifying time series anomaly patterns by handling time series data from the power system. The addition of HSA improves the model's capacity to recognize novel or hitherto undiscovered aberrant patterns.

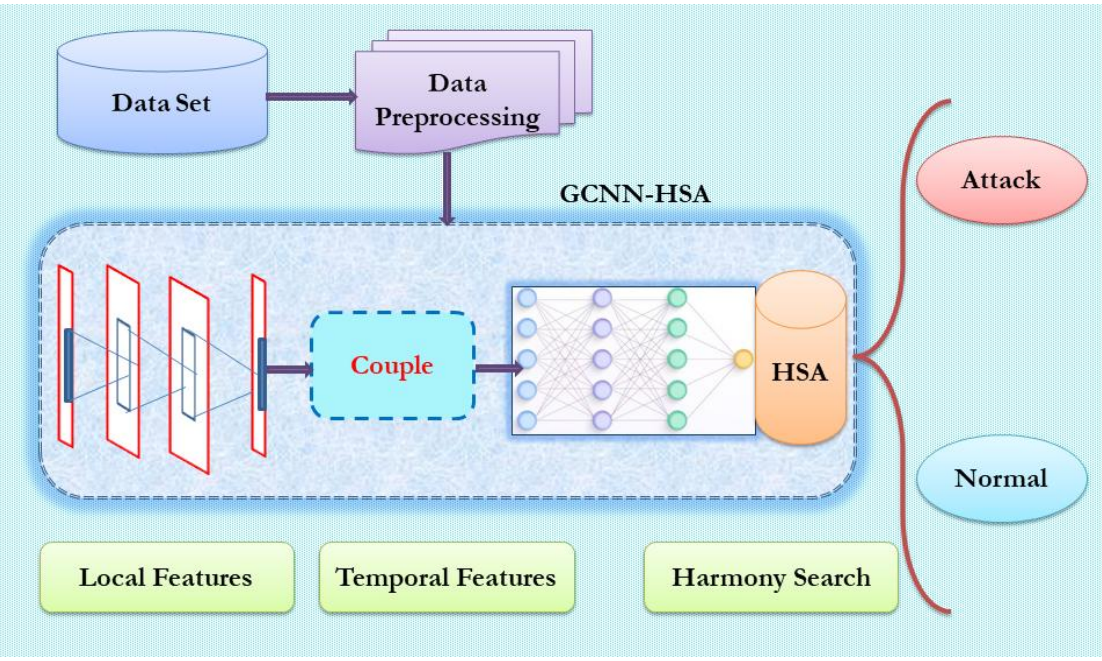


Figure 4: Overview of the GCNN-HAS

The suggested method's architecture is depicted in Figure 3. The system uses a DL method based on GCNN in two learning steps to generate an IDS. When working with datasets that have complete features, our proposed GCNN-IDS is computationally efficient. It reduces the possibility of false alarms while simultaneously providing increased precision. This research utilizes GCNN-based DL to address these challenges, which combines an HSA with a CNNDL model. The core structure of the GCNN-IDS acts as the basis for this research. A detailed illustration of the GCNN-IDS is provided in Figure 4. Two essential parts make up a CNN, as an overview of the GCNN-IDS demonstrates: a classifier and a feature extractor. The feature extractor consists of layers for pooling and convolution. The feature map that is produced as a consequence serves as the classification's input. This approach enables CNN to grasp local features effectively. However, the failure to capture temporal dependencies among critical features is a limitation. In response, we introduce recurrent layers after the CNN layers to more effectively capture both spatial and temporal features. This improves the capability of learning temporal and spatial correlations from variable-length sequences by addressing problems with disappearing and expanding gradients. In the GCNN network, the input data is first handled by the CNN and then by the recurrent layers, resulting in sequences at each time step. Spatial and temporal feature modeling is possible with this concept. To build the probability distribution over the classes, the sequence vector is passed to the next layer after being processed by a fully connected layer.

1) Non-local Message Passing Neural Network (NLMP)

Graph neural networks are based on generalizing and abstracting similar GNN network structures. These are then integrated into a single framework that offers concepts for adaptable model construction and enhancement. A deep graph neural network system that addresses the issue of excessive smoothing to obtain greater distant dependencies of nodes. Multi-layer graph convolution is stacked this method can enable the graph neural network's node information aggregation to rely on local information and information aggregation from multi-hop neighborhoods at the vertex level. To extract higher-dimensional abstract data from networks of different scales, appropriate depth and information aggregation technologies can be created to aggregate node information and structural information of the entire graph to the target node.

The following is the suggested NLMP framework in as follows:

$$h_i^{(t+1)} = \sum_{v_j \in N(v_i)} M_t\left(h_i^{(t)}, h_j^{(t)}, h_i^{(0)}, h_i^{(t-1)}, e_{ij}\right)$$

The target node i's aggregation update aggregates its first-order surrounding nodes' information along with the initial input feature $h_i^{(0)}$ of the node i and $h_i^{(t-1)}$ the node feature. At time t, there is a $\left\{h_j^{(t)} | v_j \in N(v_i)\right\}$ method for

aggregating $h_i^{(t-1)}$ and $h_i^{(0)}$ in the NLMP framework. The node information aggregation process introduces residual connections and dense connections because initial features $h_i^{(0)}$ of the target nodes are retained when the GNN is deep enough, after iterations. The preceding moment of node features $h_i^{(t-1)}$ is introduced so that the final output node features contain a part of the output results of all convolutional layers. This guarantees that node features produced by all earlier layers will always be present in the output of each graph convolution layer.

The average aggregate of neighborhood information after the import of deep graph convolution causes the features in the subgraph to tend to be since the nodes in the same subgraph are frequently densely related. As a result, the NLMP framework compensates for the over-smoothing problem brought on by the average aggregation of neighbor nodes and proposes both single- and multi-relational graph data. The following is the more detailed design of the NLMP framework:

$$h_i^{(t+1)} = M_t\left\{\left[\frac{1}{C(h)}\sum_{v_j \in N(v_i)} f\left(h_i^{(t)},h_j^{(t)}\right)\cdot g\left(h_j^{(t)}\right)\right],g\left(h_i^{(0)}\right),g\left(h_i^{(t-1)}\right)\right\}$$

The attention coefficients are represented by f in the NLMP framework. The node feature transformation function is represented by g, the message aggregation function by M, and the results are normalized by the factor $1/c(h)$.

2) Graph convolution layer

An adjacent matrix is determined by the attention coefficient. An inverse relationship between the weighted Euclidean distance, $\| v(N_i)-v(N_j) \|_2^2$, can be used to describe the similarity between pixels i and j. The data similarity, compares node-to-node similarity to quantify node feature vector similarity and computes similarity using the inner product of the linearly transformed node vectors.

$$f(h_i,h_j) = e^{\theta(h_i)^T\varphi(h_j)}$$

In this case, $\varphi(h_j) = W_\varphi h_j$ and $\theta(h_i) = W_\theta h_i$. The attention coefficient between the nodes in GAT model adopts the concatenation function based on the NLMP framework proposed is:

$$f(h_i,h_j) = e^{\text{LeakyRelu}\left(a^T[Wh_i \| Wh_j]\right)}$$

The learnable weight matrix is expressed as $W \in R^{a\times b}$, concatenate operation is represented by $(Wh_i \| Wh_j) \in R^{(2a\times 1)}$, and the feature vectors of nodes i and j are denoted by h_i and h_j.

3) Identity mapping-based feature transformation.

A straightforward linear transformation function $g(h) = Wh$ for feature transformation g in Eq [eq2]. Identity mapping in ResNet is a concept for their GCN model, which also incorporates identity mapping into GCN. The linear transformation function is denoted as, where the identity matrix I_n, and the weight matrix $W_.$:

$$g(h) = ((1-\delta_l)I_n + \delta_l W^{(l)})h$$

The aforementioned equation demonstrates how the number of layers, l, affects the weight matrix decay parameter, δ_l. The hyper-parameter, λ, that is set in GCN is $\delta_l = \log(\lambda/l+1)$. The weight matrix inactively decays as the number of layers rises due to δ_l.

The proposed DGCNN extracts the non-local structural information of nodes by stacking deep graph convolutions, given an input graph $G = (V, E)$ and its node feature matrix $X \in R^{n \times c}$. This allows for the aggregation of non-local neighborhood node information. The $(t + 1)^{th}$ graph convolutional layer has the following definition:

$$Z^{(t+1)} = \sigma\left(\left(\alpha A_{GAT}^{(t)} Z^{(t)} + \beta Z^{(t-1)} + \gamma X\right)\left((1-\delta^{(t)})I_n + \delta^{(t)} W^{(t)}\right)\right)$$

In the following equation, $Z^{(t)}$ is the result of the t^{th} graph convolution layer. The adjacency matrix of the input graph G is determined by the attention coefficient of the t^{th} graph convolution layer, where $\left(A_{GAT}^{(t)}\right)_{ij} \in \{0, r\}$, can first be made as $Z^{(0)} = Z^{(1)} = X, A_{GAT}^{(t)} \in R^{n \times n}$. The correlation between nodes' attention coefficient i and j is represented by the real number $r \in (0,1)$, if there is an edge between them, that is, $(v_i, v_j) \in E$.

The four steps below are separated into each layer of graph convolution:

1. To create the new node feature matrix $Y = A_{GAT}Z$, the node features are first transmitted through $A_{GAT}Z$ to the neighboring nodes and the nodes themselves based on various attention weights following nonlinear activation.

2. By adding Y, the output of the preceding layer's graph convolution, and the initial node feature matrix by the corresponding percentage, the new feature matrix $barY$ is produced.

3. Based on identity mapping, a linear feature transformation is applied to the node feature matrix through $barYleft((1-delta)I_n + deltaWright)$.

4. Lastly, the output of the GCN is acquired through the application of the nonlinear activation function to the preceding step's result.

4) Remaining layers

The node attributes that the convolution layer of the network has learned are combined by the graph pooling layer. The i^{th} dimension of the node embedding is denoted by $x_k^{(i)}$, while the neighbour nodes of node i are represented by N_i. The deeper the convolution layer, the more $Z^{(K)}$ that can separate a node into distinct colours or groups. The DGCNN model presented in this research, comprises 32 layers of graph convolution, whereas the DGCNN model only has 4. Based on the final channel $Z^{(K)}$ output of the graph convolution layer, the nodes are ordered in descending order. If the values of two nodes in the $Z^{(K)}$ channel are the same, then they are compared via the previous layer $Z^{(K-1)}$, and so on. A tensor of shape $n \times \sum_1^K c_t$ is received by the SortPooling layer as input from the GCN layers. The SortPooling layer expands or truncates the input tensor of $n \times \sum_1^K c_t$ to the tensor of $k \times \sum_1^K c_t$ size. This allows for easier network processing in the future. First produce a row vector of $k\left(\sum_1^K c_t\right) \times 1$ from the $k \times \sum_1^K c_t$ tensor output by the SortPooling layer.

D. Harmony Search Algorithm (HSA)

One type of swarm intelligence optimization technique is the HSA, The quest for the ideal balance is similar to the process of seeking the best solutions to engineering challenges. The HSA approach draws inspiration from the principles of harmony improvisation, encompassing four main steps. Algorithm 1 provides the pseudocode for the HSA.

Step 1: Initialization of parameters: The settings for the harmony search are controlled. These parameters include fret width (fw), harmony memory considering rate (HMCR), pitch-adjusting rate (PAR), HMS, and the last criterion. Step 2: Calculate each harmony's fitness value and initialize the Harmony Memory (HM).

$X^i = \emptyset$;

$j = 1{:}\mathbf{k}$

$a \in X^i$

$$a = [\text{rand}(0,1) \times N]$$
$$X^i \leftarrow a;$$
$$\text{Score}(i) = f(X^i);$$

Where a uniformly dispersed between 0 and 1 is represented by the $\text{rand}^{(0,1)}$. The HMS harmonies make up the harmonic memory (HM), as shown in the equation.

Step 3: A new harmony X^{new}.

$$X^{\text{new}} = \emptyset$$
$$j = 1{:}\mathbf{k}$$
$$\text{Rand}\,(0,1) < \text{HMCR}$$
$$X^{\text{new}}(j) = \text{HM}([\text{rand}(0,1) \times \text{HMS}],j)$$
$$\text{rand}(0,1) < \text{PAR}$$
$$X^{\text{new}}(j) = [X^{\text{new}}(j) \pm fw]$$
$$X^{\text{new}}(j) = [\text{rand}\,(0,1) \times N];$$

Step 4: Update the worst harmony in HM.

$$f(X^{\text{new}}) > f(X^{\text{idworst}})$$
$$X^{\text{idworst}} = X^{\text{new}};$$
$$\text{Score}\,(idworst) = f(X^{new})$$

Where the HM worst harmony index is denoted by idworst.

Step 5: The stopping requirement is attained. The outcome is ended if the halting condition is satisfied. If not, repeat steps three and four.

1) Data Preprocessing

Graph Construction: Smart grid data is represented as a graph, in which nodes represent components (e.g., sensors, meters) and edges represent connections (e.g., communication links, power lines).

Feature Extraction: Extract features, such as voltage, current, power, and status codes.

Normalization: Normalize features to ensure consistent scaling and improve model performance.

2) Graph Convolutional Network (GCN)

Smart grids can be represented as graphs where nodes represent different components (e.g., generators, substations, and meters) and edges represent their connections. GCNNs can extract features from these graphs, capturing complex relationships and patterns that traditional methods might miss.

Input Layer: The GCN takes as input a graph representation of the smart grid, where nodes represent devices (e.g., sensors, controllers) and edges represent communication links. Accepts the graph-structured input data, consisting of node features and adjacency matrix representing the graph's connectivity.

Graph Convolution Layers: Multiple layers of graph convolution are applied to capture the structural relationships and features among nodes. Each layer aggregates information from neighboring nodes, allowing the network to learn complex patterns in the graph data. This allows the model to capture spatial dependencies and structural patterns in the data.

Activation Functions: After every convolutional layer, non-linear activation functions (such ReLU) are employed to facilitate the learning.

Pooling Layers: Optional layers to reduce the dimensionality and complexity of the graph representation, aggregating information from larger neighborhoods.

Output Layer: The final output can be a classification of nodes (normal vs. anomalous) or a reconstruction of the input graph for anomaly detection.

3) Harmony Search Algorithm (HSA):

HSA can be used to optimize the hyper parameters of the GCNN. This includes tuning parameters such as learning rate, node features, and number of layers, and edge weights to enhance the GCNN's performance. HSA can also be applied to optimize the training process itself, potentially improving the convergence speed and final accuracy of the IDS. The GCNN can classify network activities as normal or anomalous based on learned patterns. The system can operate in real-time, continuously monitoring data flows and flagging suspicious activities or deviations from normal behavior.

Initialization: Generate an initial harmony memory (HM) consisting of multiple solution vectors (i.e., possible sets of GCNN parameters).

Improvisation: Create new solution vectors by combining existing vectors in HM, applying random modifications based on a harmony memory considering rate (HMCR) and pitch adjusting rate (PAR).

Parameter Optimization: HSA is utilized to optimize hyper parameters of the GCN, such as learning rates, number of layers, and regularization parameters. This optimization process helps improve the model's performance in detecting anomalies.

Fitness Evaluation: Evaluate the fitness of each solution vector using a predefined objective function, such as the accuracy of anomaly detection.

Harmony Memory: A memory structure is maintained to store the best-performing parameter sets found during the search process.

Update Harmony Memory: Update HM by replacing the worst solutions with newly improvised solutions if they offer better fitness.

Improvisation Process: New parameter sets are generated based on the harmony memory, explore the search effectively and avoid local minima.

4) Integration and Training

Parameter Optimization: Use HSA to optimize the parameters of the GCNN, including weights, biases, and hyper parameters (e.g., learning rate, number of layers).

Training Loop: Train the GCNN using labeled training data, iteratively updating the model parameters to minimize the loss.

Validation and Testing: Validate the model on a separate validation dataset to tune hyper parameters and test on a test dataset to evaluate performance.

5) Anomaly Detection Process:

Ensure the smart grid data is correctly represented as a graph structure suitable for GCNN. Properly partition data into training, validation, and testing sets to ensure model generalization. Integrate the HSA optimization process into the GCNN training pipeline to iteratively improve model performance. By combining the HSA with a GCNN for an IDSin smart grids, you leverage the strengths of both optimization and DL techniques, potentially leading to more robust and effective cybersecurity solutions for critical infrastructure like smart grids.

Training Phase: The GCN is trained on a labeled dataset including both typical and unusual data instances. The HSA optimizes the GCN parameters during this phase.

Detection Phase: After training, the GCN is used to analyze new data from the smart grid. Anomalies are detected based on deviations from learned normal patterns, with the HSA ensuring that the model remains finely tuned for accurate detection.

Improved Accuracy: The combination of GCNN and HSA enhances the model's ability to detect anomalies by leveraging graph-structured data and optimizing model parameters.

Scalability: The architecture can handle large-scale smart grid data with complex relationships and dependencies.

Robustness: The system is robust to noise and variations in the data, improving reliability in real-world scenarios.

V. Experimental Results and Discussions

The suggested model trained 64 GB of RAM and an Intel Xeon system CPU. Implemented in Tensor flow and the Keras packages in Python 3.7. First, the dataset was split into training and testing categories, and monitoring of accuracy and loss was done during the training. There were eight in the batch, 50 training epochs, and a 0.001 learning rate.

A. Training

Using the parameters and hyper parameters listed in Table 3, the training dataset is split into 80% for training and 20% for validation for each training epoch, which is how the DL network is trained. After that is done, the accuracy and loss curves for each training period can be obtained; these curves are displayed in Fig. 7 and Fig. 7, respectively. The accuracy approaches 95% in both training and validation. Conversely, the loss function in both the training and validation phases falls below 0.11.

Table 3: CNN-GRU hyper parameter configuration.

No	Hyper parameter	Recommended Values
1	Learning Rate	0.001
2	Convolutional Layers	$2-3$
3	GRU Units	32,64,128
4	Dropout Rate	$0.2-0.5$
5	Batch Size	$16-128$
6	Number of Epochs	50

B. 5.2. Metrics for performance

The following section outlines the standard metrics for assessing the quality of IDS in smart grid environments. Assessing outcomes is essential for determining how well anomaly detection algorithms work inside the vital components of the smart grid. The metrics used for detection are classified to gauge the efficiency and effectiveness of the proposed technique. Table 4 presents several evaluation metrics used in this research, such as sensitivity or recall, accuracy, precision, F1-score, false alarm rate, and detection rate to evaluate the performance of the model.

Table 4: Evaluation Metrics

Evaluation Metrics	Formula
Evaluation Metrics	Formula
Sensitivity or Recall	$Recall = \frac{T_R P}{T_R P + F_A N}$
Precision	$Precision = \frac{T_R P}{T_R P + F_A P}$
F1-Score	$F1-Score = 2 * \frac{T_R P}{2 * T_R P + F_A P + F_A N}$
Accuracy	$Accuracy = \frac{T_R P + T_R N}{T_R P + T_R N + F_A P + F_A N}$
False Alarm Rate (FAR)	$FAR = \frac{F_A P}{T_R N + F_A P}$
Detection Rate(DR)	$DR = \frac{T_R P}{T_R P + F_A N}$

The real and expected classifications are found using the confusion matrix (CM). It evaluates the results of classifying data into two groups: normal and anomalous. Within the confusion matrix, four important states need to be assessed:

1. True Positive (TR_p): This suggests that the model accurately detects typical occurrences and forecasts favorable results.

2. False Negative (FA_n): This happens when the model misclassifies anomalies as normal and predicts negative outcomes for those occurrences.

3. False Positive (FA_p): When the observed cases are normal, the model in this instance predicts a positive outcome wrongly.

4. True Negative (TR_n): This indicates occurrences that are accurately classified as anomalies and forecasts unfavorable results.

C. *Performance Evaluation*

Test System-1: This study employs multiple assessment metrics to evaluate the system's performance, including precision, recall, accuracy, area under the curve, and F1-score. Identifying the number of anomalies is reliant on understanding and analyzing the critical infrastructure system being studied. The main purpose of this metric is to analyze how well binary classification problems perform. Figure 5 shows a confusion matrix, which is a method used to calculate the model's accuracy. The effect of expanding the dataset's instance count is shown in the confusion matrices. The model performed a very good job of distinguishing between the anomaly and normal scenario. The Precision-Recall curves for every anomaly detection model are shown in Figure 6a as the decision threshold is changed, each curves shows how accuracy changes. The model's capacity to recognize every positive instance in the dataset is measured by completeness, whereas precision indicates the percentage of accurate positive detection's. These curves have been analyzed to gain a better knowledge of how well each model performs in various settings. For instance, when limiting false positives is important, a model with a high degree of precision but relatively low completeness may be preferred. Conversely, a model whose angle strikes a compromise between fullness and precision can be suitable in situations where both metrics are equally significant. Making informed judgments about which model is most appropriate for a certain anomaly detection application requires the use of visualizations. Figure 6b showcases the average AUC-ROC curve of these algorithms calculated from 15 runs for the dataset. These findings suggest that the method presented is more capable of identifying anomalies.

D. *Comparision with Existing Approaches*

The graphs from a single trial of the study demonstrate how four algorithms performed. Figure 7 provides a visual representation of the average accuracy for each target achieved by the different methods, while Table 5 displays the overall accuracy across all targets for each method. Notably, One-class SVM (OCSVM), K-nearest-neighbor outlier detection (KNNOD), Angle-based outlier detector (ABOD), and Clustering-based local outlier factor (CBLOF) [3] are identified as the top-performing algorithms in detecting cyber-attacks in smart grids based on their average AUC, with the proposed algorithm exhibiting superior performance compared to the others. The detection threshold of each algorithm, as determined by the distance to the corner'd', influences the recall, precision, and F1 score.

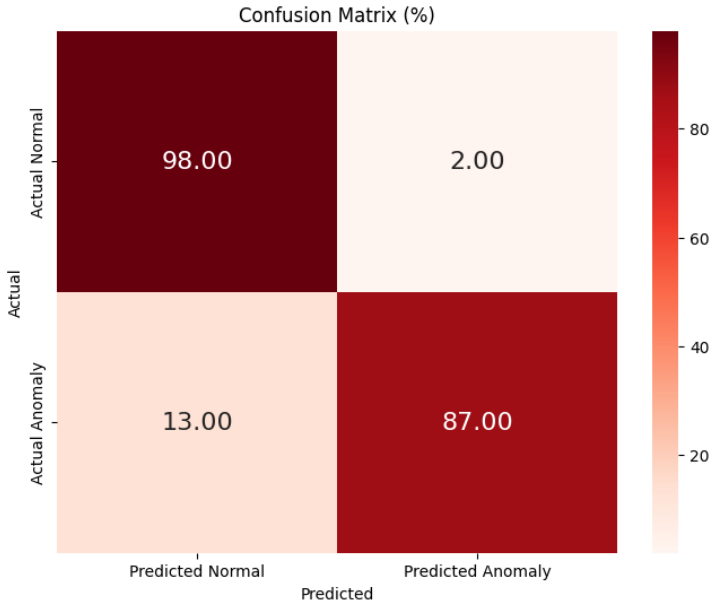
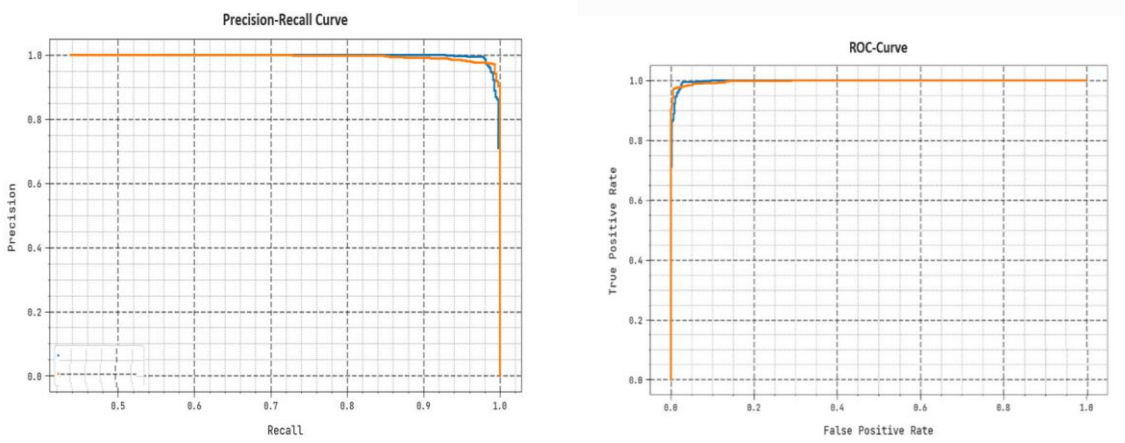


Figure 5: Confusion Matrix of the Proposed Method



a) Precision-Recall Curve *b)ROC Curve*

Figure 6: Performance Comparison of Precision-Recall Curve and ROC-Curve

Based on Figure 8a, it is evident that other algorithms exhibit slightly lower precision compared to the algorithm being presented. The ROC curves from one run of the experiment illustrate the performance of four algorithms, as depicted in Figure 8b. The method performs better when the area under the curve has a bigger value. However, the compared algorithms demonstrate significantly lower recall than the presented algorithm, as indicated in Figure 9a. This suggests that the presented algorithm is better equipped to identify attack events compared to other algorithms. Additionally, Figure 9b illustrates that the overall performance effectiveness of the presented algorithm, in terms of F1-scores, is significantly superior to that of compared algorithms, indicating that GCNN-HSA outperforms OCSVM, KNNOD, CBLOF, and ABOD, while Table 5 presents the metrics for each method across all targets.

Various ML algorithms have been examined for anomaly detection. The research includes a comparison of several techniques used for anomaly detection. The comparison is outlined in Table 5, which provides an overview of the results obtained using different methods. The proposed GCNN-HSA demonstrates superior performance in terms of accuracy and FAR with the compared techniques. It's worth noting that the similarities are provided for reference only, as different researchers have utilized diverse data distributions, pre-processing techniques, and sampling methods. Therefore, a simple comparison of metrics such as testing and training time may not be adequate. While the proposed GCNN-HSA showed better performance in the evaluated metrics, it is challenging to assert that it completely outperforms other approaches. The proposed solution, however, has the potential to significantly enhance smart grid protection by effectively identifying attacks. The results of the evaluation demonstrate that the provided models for anomaly detection perform well in identifying anomalies. The findings validate the efficiency of the approach introduced for detecting anomalies.

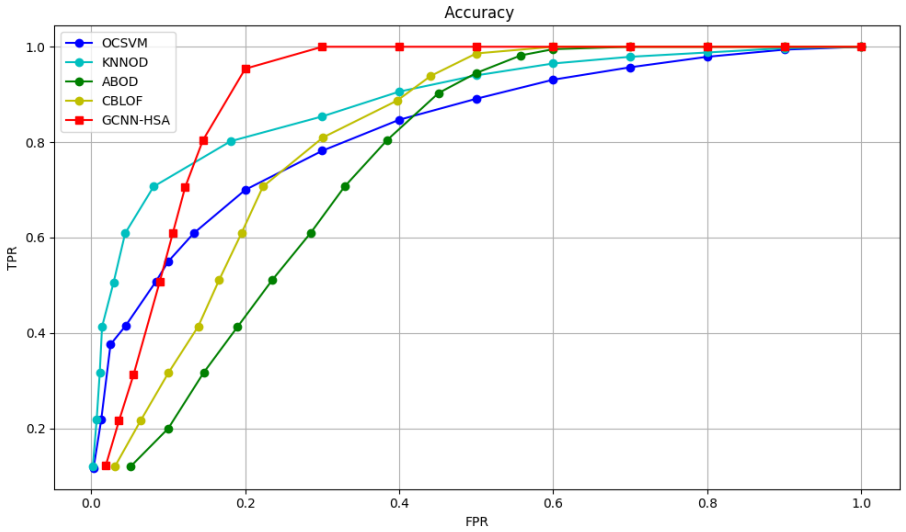
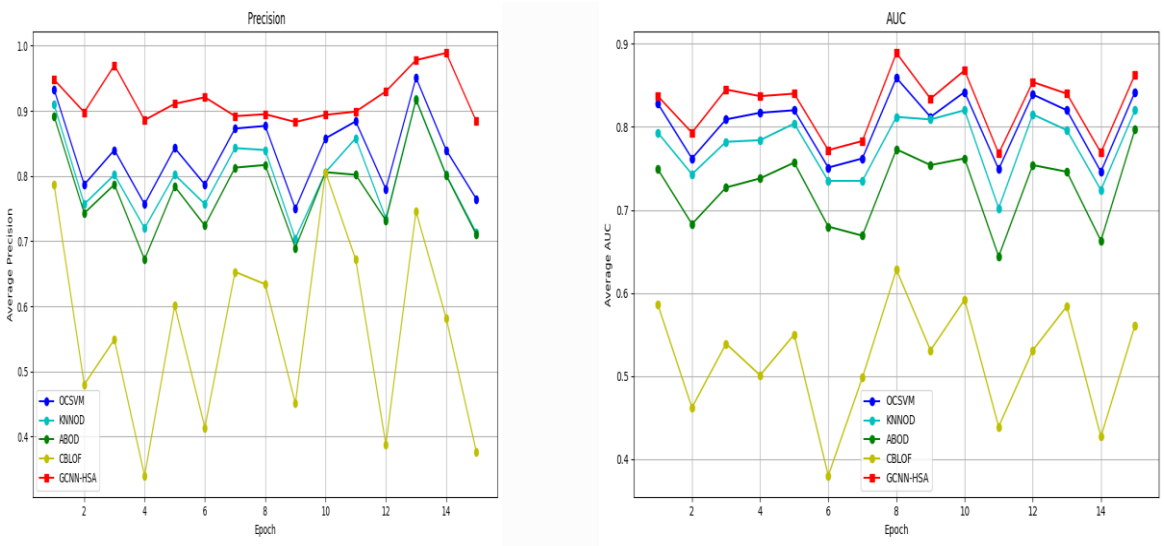


Figure 7: Performance Comparison in terms of Accuracy]



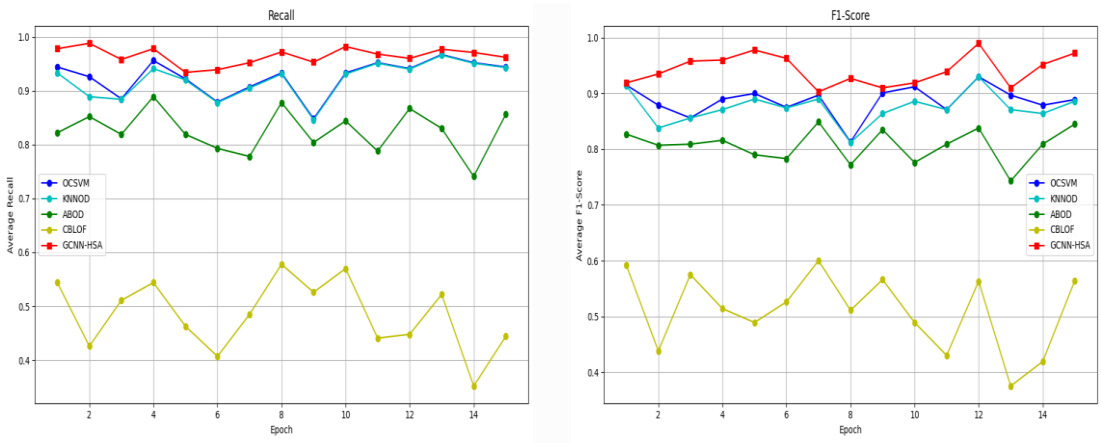
a) Precision b) AUC

Figure 8: Performance Comparison of precision and AUC

Table 5: Performance comparison for compared methods

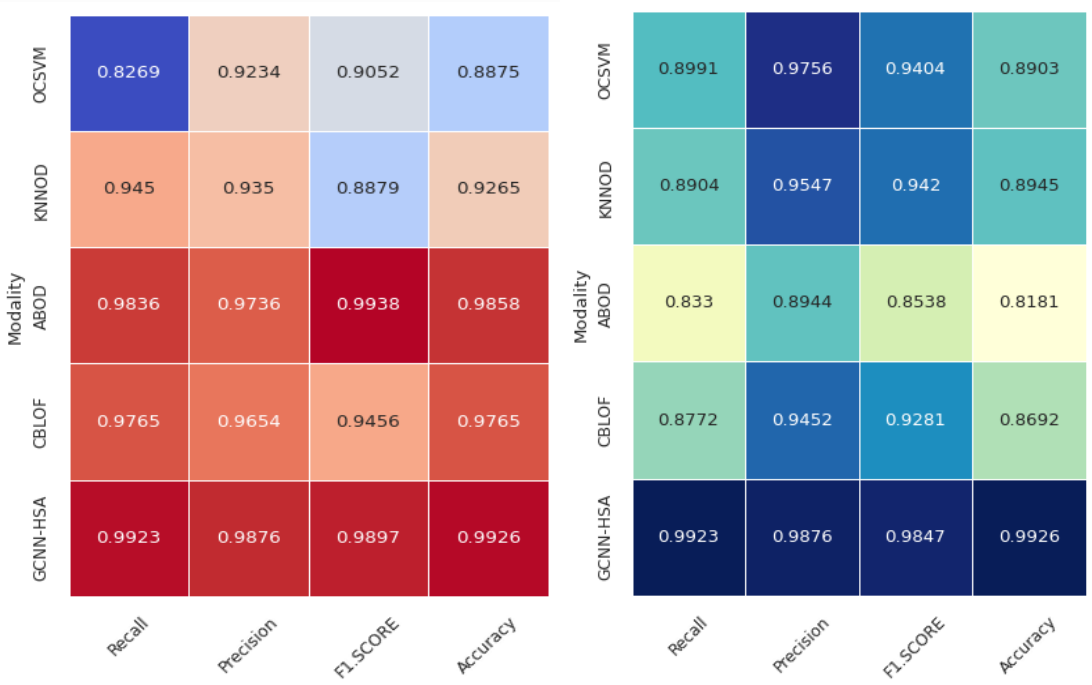
Classifier	Precision	Recall	F1-Score	DR	FAR	Accuracy
OCSVM	0.781	0.801	0.791	0.80	11.50	86.95
KNNOD	0.845	0.834	0.839	0.83	9.13	97.50
ABOD	0.8733	0.885	0.879	0.88	7.8	97.00
CBLOF	0.9633	0.9712	0.976	0.97	2.5	98.61
GCNN-HSA	0.9633	0.9712	0.976	0.97	2.5	99.21

It's crucial to evaluate the performance effectiveness of an anomaly detection model after training by using the appropriate metrics. Some common metrics for evaluation include Precision, which measures the model's accuracy in detecting anomalies by calculating the proportion of true positive forecasts to all positive predictions, and Recall evaluates the model's capacity to identify anomalies by calculating the proportion of real anomalies to true positives.



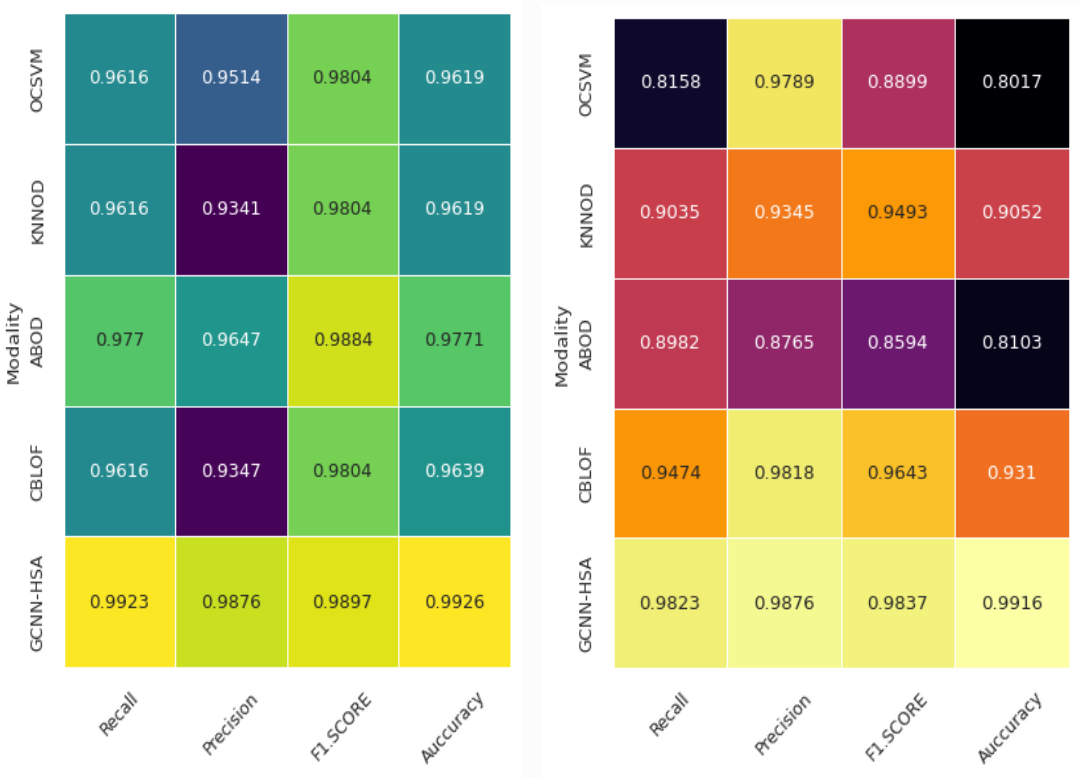
a) Recall b) F1-Score

Figure 9: Performance Comparison of Recall and F1-Score



a) Denial of Service b) Switching Areas

Figure 10: Performance Metrics for Scenarios 1 and 2



a) Injection Command b) Connection Lost

Figure 11: Performance Metrics for Scenarios 3 and 4

Additionally, in the analysis of an IDS, DR and FAR are two important parameters. FAR represents the sum of misclassified regular instances, while DR represents the number of intrusion events identified.

When working with three-class data, which consists of natural events, attacks, and normal conditions, it is imperative to differentiate between attacks and natural occurrences that occurred in the power grids. Three-class classification is consequently more important and challenging than binary-class classification. The F1-score, recall, accuracy, and precision are displayed in Table 5. The best detection results using our method for three-

class classification above 99% accuracy rate are shown in Table 6. Furthermore, the F1-score, recall, and precision findings, among other metrics, achieve acceptable success rates. Our approach outperforms other approaches in terms of performance over a wide range of classes.

To distinguish between assaults and anomalies in SG system data, this paper presents an ECS model and a Dense-Gated U-Net model based on DL approaches. One of the biggest challenges is classifying attacks and disruptions in the SGs' control units. Though they need the settings of a DNN to be adjusted, DL approaches have demonstrated encouraging results in identifying these types of attacks. The hybrid Dense-Gated U-Net with ECS is a good model for comparing different approaches. The outcome shows how important optimizing the hyper parameters of the DL models is. Results from analyzing performance using various techniques for different attack types are inaccurate. These approaches perform better than other supervised machine-learning strategies.

Table 6: Metric Classifiers for compared methods

Metric Classifiers	GCNN-HSA	OCSVM	KNNOD	ABOD
Overall Accuracy	98.76	96.92	93.62	86.55
Average Accuracy	99.23	98.78	97.46	94.63
Overall Error Rate	1.23	2.44	5.27	12.01
Average Error Rate	1.09	1.22	2.54	5.37
Macro-Averaged Precision	99.13	98.11	95.87	91.41
Macro-Averaged Recall	95.47	92.30	86.21	69.84
Macro-Averaged F1-Score	96.52	95.12	90.78	79.18
Micro-Averaged Precision	98.67	96.92	93.62	86.55
Micro-Averaged Recall	98.56	96.92	93.62	86.55
Micro-Averaged F1-Score	98.94	96.92	93.62	86.55

Test System-2: This study employs multiple evaluation metrics to assess the system's performance through heat maps. Identifying the number of anomalies is reliant on understanding and analyzing the critical infrastructure system being studied. The main purpose of this metric is to analyze how well binary classification problems perform. In the context of a denial of service attack scenario, Figure 10 (a) displays a heat map representing the detection performance metrics of the GCNN-HSA anomaly detection method. The graphic makes it clear that the performance levels of the GCNN-HSA approach varied when compared to other models. It demonstrated strong precision with an accuracy of 0.9926 and F1-score values of 0.9876 and 0.9897, respectively. This suggests that the GCNN-HSA approach balanced precision and recall while successfully identifying anomalies with few false positives.

On the other hand, the OCSVM approach had trouble accurately classifying cases, resulting in a lower accuracy of 0.8875. Its recall, precision, and F1-score were significantly lower, at 0.8269, 0.9234, and 0.9052, respectively. However, the KNNOD and ABOD approaches performed quite well, outperforming 0.92 in accuracy and attaining flawless precision, recall, and F1 scores, demonstrating their resilience in identifying denial-of-service assaults. With an accuracy of 0.9765, the CBLOF approach performed somewhat worse than the GCNN-HAS method. While the precision and F1-score values were acceptable at 0.9876 and 0.9897, respectively, the recall was significantly lower at 0.9923, suggesting a larger false negative rate.

Similarly, the CBLOF and ABOD approaches demonstrated efficacy in identifying denial-of-service attacks with an accuracy of 0.9765 with flawless performance scores. The KNNOD approach, in contrast to the other approaches, had a lower accuracy of 0.9265, efficiency measure values of 0.9350, 0.9450, and 0.8879, respectively. This indicates that the method has a higher recall but a lower precision.

The heat map presented in Figure 10(b) compares the detection performance metrics of various anomaly detection methods, including GCNN-HSA, CBLOF, ABOD, KNNOD, and OCSVM, during a switching attack scenario. In this scenario, the GCNN-HSA, CBLOF, and ABOD methods exhibited good accuracy and precision. CBLOF achieved an accuracy of 0.8692 and a precision of 0.9452, while ABOD had an accuracy of 0.8181 and a precision of 0.8944. Conversely, the KNNOD method showed poor performance with an accuracy of 0.8945 and a recall of 0.8904, while the OCSVM model had moderate accuracy (0.8903) and precision (0.9756). The GCNN-HSA method outperformed the others, displaying an accuracy of 0.9952 and a perfect precision score of 0.9876. Overall, the consistent strong performance of GCNN-HSA, CBLOF, and ABOD indicates their capacity to identify irregularities in the context of the switching attack.

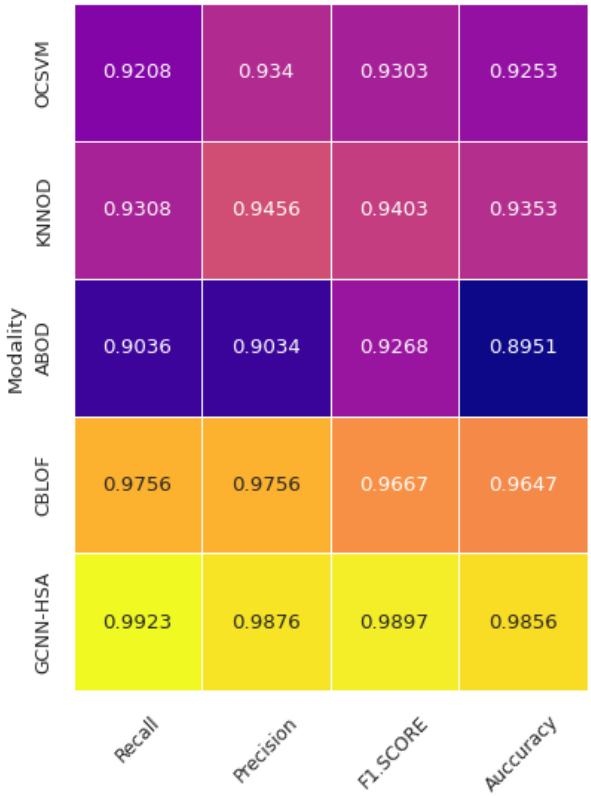


Figure 12: Performance Metrics for Scenario 5

In the heat map presented in Figure 11(a), the performance metrics of the GCNN-HAS anomaly detection methods during the injection command attack scenario. Both the GCNN-HAS and ABOD anomaly detection techniques showed excellent accuracy, precision, and recall in this particular situation. The accuracy values of the other methods ranged from 0.9639 to 0.9619, with the GCNN-HAS method achieving an accuracy score of 0.9926. Precision scores of 0.9876, 0.9347, 0.9647, 0.9341, and 0.9514 were shown for all approaches, showing almost flawless identification of attack events. The recall values for the other methods varied from 0.9616 to 0.977, with the GCNN-HAS approach displaying a value of 0.9923. The GCNN-HAS approach continuously beat the other approaches, proving its better injection attack detection powers. Furthermore, both the GCNN-HAS and CBLOF anomaly detection techniques obtained high accuracy, precision, and recall in the injection assault scenario, supporting the GCNN-HAS method's superior detection capabilities.

Using test data from a connection loss attack scenario, the heat map in Figure 11(b) compares the assessment metrics of the GCNN-HSA with other anomaly detection techniques. With F1-score values of 0.9837 and 0.9642, respectively, the CBLOF and ABOD approaches of the GCNN-HAS method demonstrated much better detection performance than the other methods. This suggests an accuracy and recall performance that is well-balanced, which is essential for precise anomaly identification. However, the F1 scores of other approaches, such as the OCSVM and KNNOD methods, were lower; the OCSVM method scored 0.8899, while the KNNOD method scored 0.9493. This shows that although these methods might be superior to the GCNN-HSA method in some areas, such as recall, they do not perform as well overall in terms of precision and recall balance. These variations in the F1-score highlight how well the GCNN-HAS technique balances precision and recall, providing a more sensible means of anomaly detection in the event of connection loss assaults. To maximize the cyber security of smart grid networks, this balance must be maintained between limiting false positives and guaranteeing the correct identification of anomalies.

In situations involving a rogue device attack, the GCNN-HSA anomaly detection methods displayed superior performance compared to other methods, as illustrated in Figure 12. Specifically, the GCNN-HSA method, as well as the CBLOF and ABOD methods, obtained outstanding F1-scores of 0.9897, demonstrating very accurate detection with low false negatives and positives. In contrast, the KNNOD and OCSVM methods exhibited more varied performances. The ABOD approach displayed a lower F1-score of 0.9268, indicating a lower detection performance, compared to the GCNN-HSA method, which attained a perfect F1-score. Furthermore, these findings highlight how well the GCNN-HSA anomaly detection techniques perform in precisely locating rogue

devices in smart grid networks. These techniques, which make use of DL modeling, offer a reliable way to improve cyber security by efficiently identifying unusual behavior. In addition, the F1-scores of 0.9667 obtained by the CBLOF and KKNOD approaches demonstrate remarkable accuracy in detection with few false positives and negatives. In contrast to the GCNN-HSA approach, other techniques showed more inconsistent performance and lacked balanced precision and recall.

These definitive findings highlight how well the GCNN-HSA anomaly detection technique performs in precisely detecting rogue devices in smart grid networks. The GCNN-HSA anomaly detection method consistently outperformed CBLOF, ABOD, KNNOD, and OCSVM in detecting different forms of assaults in smart grid networks, despite no single strategy showing higher performance across all considered attacks. There are various reasons for this supremacy. First off, by using DL architectures auto encoders and GRUs in particular—to accurately simulate the intricate temporal correlations present in time-series data, like network traffic, the GCNN-HSA anomaly detection technique takes advantage of these capabilities. This makes it possible for the methods to pick up on minute patterns and abnormalities that other methods would miss. Furthermore, the GCNN-HSA model may be used to train data features through ML, which enables it to provide feature representations that are more discriminative and informative than those produced by other techniques.

E. Discussions

The findings highlight the effectiveness of utilizing GCNN-HSA for fine-tuning hyper parameters in a DL model aimed at identifying fraudulent activities. The notable improvements in recall and AUROC, demonstrate a strong capability to detect anomalies, which is crucial in real-world scenarios. These modifications underscore the importance of fine-tuning the model to deal with the complexities as well as disparities found in the SG dataset. The customized model better caters to the specific requirements of the anomaly detection task. Although the results are promising, it's important to consider the possible variability brought on by GA's stochastic character. Further investigation could explore the findings' coherence between different datasets and iterations. Further insights into the model's applicability and efficacy may be obtained by analyzing its performance on data from the SGCC dataset.

The results of this study demonstrate a good synergy with earlier research on hyper parameter optimization approaches and their use in anomaly identification. Numerous scholarly investigations have underscored the importance of sophisticated DL algorithms in augmenting the accuracy and efficacy of fraud detection systems. To increase detection performance metrics, our research goes one step further and uses HSA to fine-tune GCNN model hyper parameters. The significant improvement in model accuracy, along with improvements in precision, recall, and AUROC, is consistent with the favorable outcomes of hybrid GCNN model usage. This analysis highlights the significance of our study in the realm of smart grid anomaly detection and supports the validity of our methodology.

VI. Conclusion and Future Work

The integration of digital technologies into power grids has significantly improved sustainability, reliability, and interconnectivity. In this study, we proposed a novel hybrid anomaly detection approach that combines Graph Convolutional Neural Networks (GCNN) with the Harmony Search Algorithm (HSA) for effective intrusion detection in smart grid environments. The HSA component optimizes GCNN hyper parameters to enhance the model's ability to identify anomalies with high precision and speed. Experimental results demonstrate that our model outperforms existing techniques, achieving superior performance across key evaluation metrics such as AUROC, accuracy, precision, recall, and F1-score.

Notably, our method attained an impressive accuracy of 98.76%, confirming the efficacy of HSA in navigating the complex hyper parameter space of GCNNs compared to conventional methods. This integration not only boosts the detection capability but also addresses the critical need for robustness and scalability in anomaly detection frameworks within cyber-physical systems like smart grids.

Furthermore, this study advances the interpretability of AI-driven anomaly detection systems by integrating Explainable AI (XAI) components. These components help demystify the model's decision-making process by identifying the most influential features and graph regions contributing to each detection. By providing transparency, XAI builds operator trust, supports better incident response, and aids in compliance with safety and regulatory standards.

In summary, the proposed GCNN-HSA model, enhanced with XAI capabilities, presents a promising and practical solution for safeguarding smart grids. It not only strengthens cyberattack detection but also ensures the protection of sensitive user data and system integrity in real-world deployments. Future Work: Subsequent

research will focus on further enhancing the interpretability and real-time applicability of the model. Key directions include: Adapting the proposed model for low-latency anomaly detection in real-time smart grid operations. Validating the model across diverse smart grid datasets and cyberattack scenarios to test its generalizability and resilience.

VII. Declarations Author Contributions:

Conceptualization, investigation, writing—original draft preparation, writing—review and editing, J.Jasper; software, visualization, data curation, methodology, J.Jasper, Dr.Praveen B M, S.Berlin Shaheema; conceptualization, Anish Kumar J. All authors have read and agreed to the published version of the manuscript.

Funding: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement:

The power system cyber-attack datasets are publicly available at https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-sets (accessed on 03 June 2024).

The test system used in this study are available at https://ieee-dataport.org/documents/ics-dataset-smart-grid-anomaly-detection (accessed on 25 June 2024)

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] Palensky, Peter, and Friederich Kupzog. "Smart grids." Annual Review of Environment and Resources 38 (2013): 201-226.

[2] Ding, Jianguo, et al. "Cyber threats to smart grids: Review, taxonomy, potential solutions, and future directions." Energies 15.18 (2022): 6799.

[3] Qi, Ruobin, et al. "Detecting cyber attacks in smart grids using semi-supervised anomaly detection and deep representation learning." Information 12.8 (2021): 328.

[4] Kurt, Mehmet Necip, et al. "Online cyber-attack detection in smart grid: A reinforcement learning approach." IEEE Transactions on Smart Grid 10.5 (2018): 5174-5185.

[5] Zhang, Yichi, et al. "Distributed intrusion detection system in a multi-layer network architecture of smart grids." IEEE Transactions on Smart Grid 2.4 (2011): 796-808.

[6] Radoglou-Grammatikis, Panagiotis I., and Panagiotis G. Sarigiannidis. "Securing the smart grid: A comprehensive compilation of intrusion detection and prevention systems." Ieee Access 7 (2019): 46595-46620.

[7] Vajpayee, Prashant, and Gahangir Hossain. "Reduction of Cyber Value at Risk (CVaR) Through AI Enabled Anomaly Detection." SoutheastCon 2024. IEEE, 2024.

[8] Bhattacharya, Sweta, et al. "Incentive mechanisms for smart grid: State of the art, challenges, open issues, future directions." Big Data and Cognitive Computing 6.2 (2022): 47.

[9] Azad, Salahuddin, Fariza Sabrina, and Saleh Wasimi. "Transformation of smart grid using machine learning." 2019 29th Australasian Universities Power Engineering Conference (AUPEC). IEEE, 2019.

[10] Konstantinou, Charalambos, and Saraju P. Mohanty. "Cybersecurity for the smart grid." Computer 53.5 (2020): 10-12.

[11] Mohammadpourfard, Mostafa, et al. "Cyber-resilient smart cities: Detection of malicious attacks in smart grids." Sustainable Cities and Society 75 (2021): 103116.

[12] Alhanaf, Ahmed Sami, Murtaza Farsadi, and Hasan Huseyin Balik. "Fault Detection and Classification in Ring Power System with DG Penetration Using Hybrid CNN-LSTM." IEEE Access (2024).

[13] Wang, Feng, et al. "A LSTM-Based Method for Prediction of Network Security Situation in Smart Electric Power Grid." 2022 IEEE 5th International Conference on Automation, Electronics and Electrical Engineering (AUTEEE). IEEE, 2022.

[14] Aderibole, Adedayo, et al. "Blockchain technology for smart grids: Decentralized NIST conceptual model." IEEE Access 8 (2020): 43177-43190.

[15] Ferrag, Mohamed Amine, Messaoud Babaghayou, and Mehmet Akif Yazici. "Cyber security for fog-based smart grid SCADA systems: Solutions and challenges." Journal of Information Security and Applications 52 (2020): 102500.

[16] Tariq, Noshina, et al. "A fog-edge-enabled intrusion detection system for smart grids." Journal of Cloud Computing 13.1 (2024): 43.

[17] Wen, Mi, et al. "Security and efficiency enhanced revocable access control for fog-based smart grid system." IEEE Access 7 (2019): 137968-137981.

[18] Hussain, Shahbaz, et al. "A novel hybrid cybersecurity scheme against false data injection attacks in automated power systems." Protection and Control of Modern Power Systems 8.3 (2023): 1-15.

[19] Mohammed, Saad H., et al. "Evaluation feature selection with using machine learning for cyber-attack detection in smart grid." IEEE Access (2024).

[20] Kim, Tung T., and H. Vincent Poor. "Strategic protection against data injection attacks on power grids." IEEE Transactions on Smart Grid 2.2 (2011): 326-333.

[21] Chen, Po-Yu, et al. "Detection of false data injection attacks in smart-grid systems." IEEE Communications Magazine 53.2 (2015): 206-213.
[22] Rawat, Danda B., and Chandra Bajracharya. "Detection of false data injection attacks in smart grid communication systems." IEEE Signal Processing Letters 22.10 (2015): 1652-1656.
[23] Mololoth, Vidya Krishnan, Saguna Saguna, and Christer Åhlund. "Blockchain and machine learning for future smart grids: A review." Energies 16.1 (2023): 528.
[24] Kim, Yoonjib, Saqib Hakak, and Ali Ghorbani. "Smart grid security: Attacks and defence techniques." IET Smart Grid 6.2 (2023): 103-123.
[25] Ozay, M.; Esnaola, I.; Vural, F.T.Y.; Kulkarni, S.R.; Poor, H.V. Machine learning methods for attack detection in the smart grid. IEEE Trans. Neural Netw. Learn. Syst. 2015, 27, 1773–1786.
[26] Wang, D.; Wang, X.; Zhang, Y.; Jin, L. Detection of power grid disturbances and cyber-attacks based on machine learning. J. Inf. Secur. Appl. 2019, 46, 42-52.
[27] Kotenko, Igor, et al. "An approach to detecting cyber attacks against smart power grids based on the analysis of network traffic self-similarity." Energies 13.19 (2020): 5031.
[28] Elbez, Ghada, et al. "Detection of DoS attacks using ARFIMA modeling of GOOSE communication in IEC 61850 substations." Energies 13.19 (2020): 5176.
[29] Unsal, Derya Betul, et al. "Enhancing cybersecurity in smart grids: false data injection and its mitigation." Energies 14.9 (2021): 2657.
[30] Aziz, Israa T., et al. "T2S2G: a novel two-tier secure smart grid architecture to protect network measurements." Energies 12.13 (2019): 2555.
[31] Culler, Megan, and Hannah Burroughs. "Cybersecurity considerations for grid-connected batteries with hardware demonstrations." Energies 14.11 (2021): 3067.
[32] Hasan, Md Nazmul, et al. "Electricity theft detection in smart grid systems: A CNN-LSTM based approach." Energies 12.17 (2019): 3310.
[33] Xie, Shaohao, et al. "A New Secure and anonymous metering scheme for smart grid communications." Energies 12.24 (2019): 4751.
[34] Jithish, J., et al. "Distributed anomaly detection in smart grids: a federated learning-based approach." IEEE Access 11 (2023): 7157-7179.
[35] Abdel-Basset, Mohamed, Nour Moustafa, and Hossam Hawash. "Privacy-preserved generative network for trustworthy anomaly detection in smart grids: A federated semisupervised approach." IEEE transactions on industrial informatics 19.1 (2022): 995-1005.
[36] Khan, Muhammad Ashfaq. "HCRNNIDS: Hybrid convolutional recurrent neural network-based network intrusion detection system." Processes 9.5 (2021): 834.
[37] Duan, Jing. "Deep learning anomaly detection in AI-powered intelligent power distribution systems." Frontiers in Energy Research 12 (2024): 1364456.
[38] Radoglou Grammatikis, P., Sarigiannidis, P., Efstathopoulos, G. and Panaousis, E. (2020) ARIES: A Novel Multivariate Intrusion Detection System for Smart Grid. Sensors, 20, Article 5305.
[39] https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-sets
[40] https://www.fit.vut.cz/research/project/1303/.
[41] https://ieee-dataport.org/documents/ics-dataset-smart-grid-anomaly-detection.
[42] Shaheema, S. Berlin, and Naresh Babu Muppalaneni. "Explainability based Panoptic brain tumor segmentation using a hybrid PA-NET with GCNN-ResNet50." Biomedical Signal Processing and Control 94 (2024): 106334.
[43] Zhou, Yuchen, et al. "A deep graph convolutional neural network architecture for graph classification." Plos one 18.3 (2023): e0279604.
[44] Das, Swagatam, et al. "Exploratory power of the harmony search algorithm: analysis and improvements for global numerical optimization." IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 41.1 (2010): 89-106.
[45] Ambia, Mir Nahidul, et al. "Harmony search algorithm-based controller parameters optimization for a distributed-generation system." IEEE Transactions on Power Delivery 30.1 (2014): 246-255.