Enhancing IoT Network Efficiency Using Machine Learning Approaches

Varun Arvind[1*], *Kushi* Budale[1], *Lavanya* Balasubramanyam[1], *Annapareddy* Haarika[1], *Geeta* Amol Patil[1], and *Surekha.* K B[1]

[1]Department of Information Science and Engineering, BMS Institute of Technology and Management, Bengaluru, India

Abstract. Technology has seen an exponential growth in the recent times. This in turn, has advanced the extent of device connectivity at an unrivaled pace by enabling large number of devices to communicate with each other at the same time and smooth data transfer. However, the management of such interconnected systems under the IoT framework stands to be a formidable task. Its deployment gets impeded by several challenges, including resource wastage, scalability, heterogeneity, and security attacks. This paper aims to study the unusual activities that can threaten the security of IoT connections, and to offer a feasible solution using deep learning approaches. In particular, it explores Graph Neural Networks (GNNs) and specifically examines Graph Convolutional Networks (GCNs). The 2-layer GCN model outperforms a similar MLP with an F1-score of 0.9577 and a test accuracy of 95.89%. The outcomes show the model's robustness and strong capacity for generalization, with steady convergence throughout training. These findings outline the scope for this type of neural network model to be vastly employed in anomaly detection mechanisms, leading to enhanced IoT network management, deployment and performance capabilities.

1 Introduction

In the continuously evolving technology landscape, a large number of IoT devices are "resource-constrained" with limited processing power, memory, or bandwidth. Additionally, IoT devices, often connected through suboptimal radio channels and deployed in harsh environments, face performance complications such as slow data transmission and data loss [1]. To resolve this problem, a deep learning-based approach has been employed. Numerous deep learning algorithms are in existence for graph-structured data, namely Graph Autoencoders (GAEs), Graph Recurrent Neural Networks (GRNNs) [2], Graph Neural Networks (GNNs). In this paper, emphasis is placed on Graph Convolutional Networks (GCNs) due to their special capability of being able to efficiently register highly complex relationships and dependencies within graph structures, making them particularly well-suited for our issue. In order to evaluate the efficiency of the proposed method, an environment tracking situation is considered, where temperature, humidity, light, and CO2 levels are detected.

2 Literature Review

In a world where technology is evolving at an exponential rate, a number of approaches have been proposed to improve IoT network communications. The following is a selection of key research papers that implement these methods.

This paper provides some of the details of HP-GCN-GAD that uses concatenation as an aggregation method. In combination with high-pass filtering, structured training, and ChebConv as feature extractors, the model significantly enhances anomaly detection [3]. Problems with hyperparameter tuning and scalability affecting the larger graphs, and semi-supervised learning do not generalize well to real-world settings.

Fixed convolutional layers within GCNs solve this problem by reducing the need for tuning, sampling-based techniques such as GraphSAGE help in allowing scalability, and self-supervised learning aids in improving generalization. Therefore, HP-GCN works well for anomaly detection, while GCNs are more flexible to various graph structures and efficient.

Fulfilling all the requirements of security, the work of Onyia et al. (2021) delves into the implementation of machine learning to identify anomalies within the IoT network [4]. Their findings suggest that a machine learning-based solution could very effectively identify unauthorized and intrusive access to the network, thus further bolstering the security status of the IoT deployments.

This paper uses a two-layer Graph Convolutional Network (GCN) which highly supports the learning of useful inputs for efficient anomaly detection with scalability, improved accuracy, real-time adaptability, and optimal network traffic. The challenges consist of computational complexity, data dependency, limited interpretability, and sensitivity towards the network structure [5]. GCNs mitigate these limitations through graph sampling techniques, various libraries for sparse

* Corresponding author: 1by22is188@bmsit.in

© The Authors, published by EDP Sciences. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

matrix operations, and hardware acceleration to enhance the efficiency of the GCN. In yet another interesting research work, ElMossallamy et al. (2020) have considered the role of reconfigurable intelligent surfaces (RISs) [6] in the optimization of communication links in IoT networks. Their work shows that through machine learning, such surfaces can be dynamically managed to improve quality of service and resource allocation.

The conversion technique proposed in this paper provides data representation in graph form for Graph Neural Networks (GNNs), which gives guarantees on performance, scalability, robustness, and automation but suffers from various issues like computational complexity, data sensitivity, low interpretability, and generalization. Interpretability has been found to be enhanced using GNNExplainer and Graph Attention Networks (GAT) [7]. This gives GCN greater efficiency through self-supervised learning, domain adaptation, and dropout techniques, which involve setting the values of some features to zero during the model training process, with the objective of boosting generalization by eliminating the dependence of the model on particular nodes or edges.

A review by Aboubakar et al. (2021) indicates that the challenges faced in IoT network management include heterogeneity and scalability. They argue that machine learning is the answer to adaptive management frameworks that will address these challenges. These studies, in brief, show how machine learning will transform the very fabric and the functioning of IoT networks into possibly better, safer, and scalable platforms, wherein such networks will then be ready to be the backbone for more robust and responsive IoT applications [8].

A GAN-based strategy for detecting image forgeries was presented by Sharma et al. [9]. Their research emphasized on establishing a generalized framework that uses generative adversarial networks' (GANs') learning capabilities to detect forged areas in photos. In comparison to conventional classifiers, the study showed detection accuracy of a higher order and focused on the approach's resilience to various image manipulation techniques. A GAN based CNN model for deepfake detection was presented [2]. This study combined the generative ability of GANs with the discriminative strength of convolutional neural networks (CNNs) to synthesize a hybrid detection framework mechanism. The model effectively generalizes across several deepfake datasets, thus resolving the limitations of strategies that rely on a particular type of datasets.

Moreover, the benefits of attention mechanisms in GAN-based models for quality detection were studied [10]. This work represented how extra attention layers in GANs can enhance the network's focus on important forged areas within an image. This mechanism upgraded detection accuracy as well as understanding through observing highly subtle entities introduced during the forgery process.

Overall, these studies represent the advanced developments made in the synthesis of highly effective and efficient models for deepfake and forgery detection. Hybrid CNN and GAN frameworks, and attention mechanisms are key directions in this area of study.

The method proposed in this paper integrates natural links and the dynamic real-world interactions to bridge the gap of contextual awareness. Anomalies can thus be easily detected with minimal computational load compared to other existing techniques.

3 Challenges Faced by IoT Networks and Solutions

This paper addresses the issues of security and privacy in the IoT networks, through the use of artificial intelligence and machine learning, as tools to increase the effectiveness of fault detection, automation, and decision-making in increasingly complex environments. The important challenges in IoT networks are further described as follows:

Heterogeneity: The rapid expansion of wireless networking systems has led to a wide range of communication pathways. For the management of these networks, architectures like Software Defined Networking (SDN), semantic-based approaches, or cloud-based solutions are incorporated with IoT gateways. These gateways establish effective interaction between IoT devices (IP based and non-IP based). The drawback of such networks is their inconsistency in satisfying the Quality of Service (QoS) standards [11]. Strategies to ensure the fulfillment of these standards are necessary for seamless interaction between IoT devices.

Scalability: With the expansion of IoT networks, and due to the emergence of technologies such as 5G, Web 2.0 there is an increase in the number of devices that are interconnected. This leads to complexity in managing the communication between these IoT devices. To tackle this problem, network management solutions are necessary. One of the common management solutions is fog computing, which deals with extending the cloud resources closer to edge devices [12]. It provides increased scalability by migrating cloud features, like computing, storage, and networking, to be closer to the devices. This technology facilitates acceleration on the reconfiguration of IoT solutions by leveraging both edge and cloud resources.

Resource Efficiency: Optimum performance in resource-constrained IoT networks is necessary in today's world. There is also a need to maintain a high level of mobility and reliability with self-configuration [13]. Many machine learning methodologies can be put to use for better resource efficiency; however, these should have the capacity of storage and processing to decrease unnecessary load or traffic [14] on the network. Managing IoT networks is challenging because of resource limitations, but this can be improved with proper architectural design.

Security and Privacy: Safeguarding of confidential data is pivotal. Due to the limited resources and the low powered nature of IoT devices, the generation of mechanisms that guarantee security is complex. All popular approaches are vulnerable to known and unknown attacks, this results in several classes of security challenges in resource constrained networks

[15]. One notable example is the advancement in IoT security by the establishment of a key management scheme for light-weight block ciphers. An encryption system that provides for key generation, storage, transmission, and utilization is needed. This paper proposes another alternative by using AI and ML techniques to detect breaches [16]. Anomalies are detected instantaneously to thwart security threats, which facilitates their neutralization.

4 Discussion

For assessing the viability of this method, it was applied on an environment monitoring dataset, which consisted of four sensors readings namely humidity, temperature, light, and CO2. Figure 1 portrays the distribution of sensor data. This shows that there is high diversity in values of the training dataset.

Figure 1 shows the distribution of sensor data, namely temperature, humidity, CO2 and light. The temperature, humidity, and CO2 sensors follow a bimodal distribution. While temperature shows a clear bimodal curve with two distinct peaks, CO2 distribution displays a uniform baseline, and the humidity sensor demonstrates an asymmetric pattern. The light sensor exhibits a periodic trend that mirrors a sinusoidal shape [17].

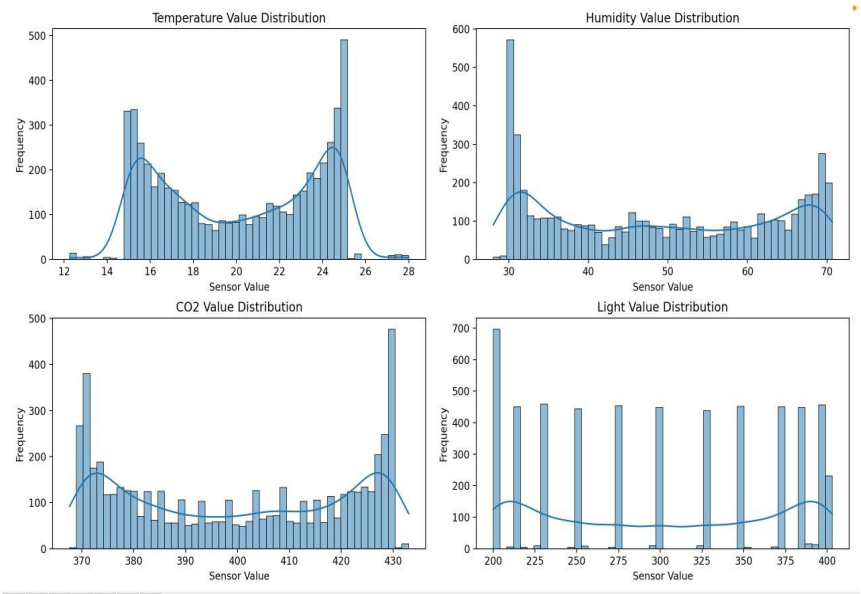

Fig. 1. Distribution of sensor data.

The dataset has 728 rows, which are equally distributed for normal and abnormal values. Several system and software requirements have to be fulfilled in order to ensure seamless execution of the proposed system. A recent release of a globally standard operating system is preferable. VS Code and Python can be installed from their official websites. Relevant Python libraries, such as pandas, numpy, gcn and networkx, should also be installed. Various sensor types are used for recording data values from the environment. For reading the processed data, NumPy files are used. An optimal system configuration comprises of 8 GB RAM, 2 GHz dual-core processor, at least 256 GB SSD storage and GPU processor. This facilitates coding, development and data analysis. It is of prime importance that storage space is sufficient and supplementary needs, which are mainly dependent on dataset size, are met. By incorporating this configuration, it is possible to ensure efficient workflow and boost productivity.

5 Methodology

To address the security vulnerability, present in IoT networks, the paper suggests the use of a two-layer Graph Convolutional Network (GCN). The AI based models such as GCN would serve for the noticing of such abnormalities by considering the interactions among sensors for the real-time classification of behavior anomalies. The abnormal behavior in sensor networks that refers to IoT would be the problem deviation of the sensor readings or even operation patterns, which is always a clear indicator of the possible fault or anomaly.

The GCN works on the principle of message passing between the neighboring nodes (sensors). It considers every device in the network as a node of a graph. The system determines the interaction and dependency from the structure of the graphical representation to make efficient possible extraction. It learns from the past data patterns and structures of the network to infer its relations, which enables its capacity to probably recognize unusual behaviors- usually anticipated correlations.

A two-layer GCN can be distinguished from a single-layer GCN by considering its capability to extract information from a much wider neighborhood in the graph. Moreover, a two-layer GCN transforms the first layer's output by supplementing with another convolution, which provides it with the ability to embed information from 2-hop neighbors. This is in contrast to a single-layer GCN which relies only on 1-hop neighbors for the feature aggregation process. Hence, highly complex and less intuitive relationships and

patterns, which are based on connections of indirect nature, can be effectively learnt by the model, albeit the fact that addition of extra layers could potentially cause over-smoothing. In over-smoothing, the similarity of node features takes a very high value, which hinders node differentiation by the model.

The above Figure 2 depicts the general working of the GCN model in the case of environmental monitoring situations. It represents the sensors (light, temperature, humidity, and CO2) as nodes of a graph. This GCN model reflects relations by using correlation to decide the edges. The model is thus able to understand the real-world relationships between the sensors.

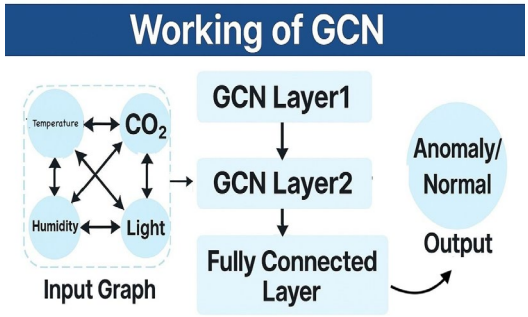

Fig. 2. Working of GCN for the model.

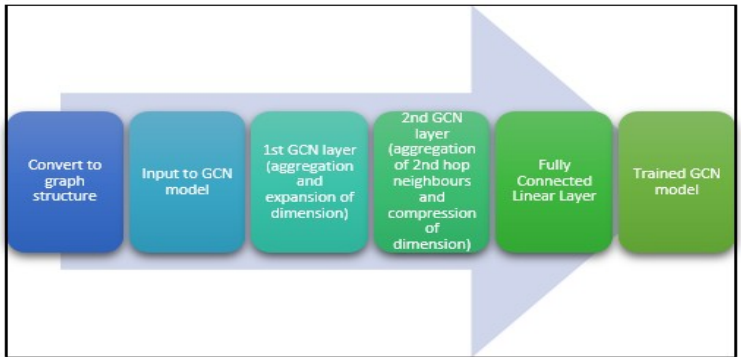

Fig. 3. In-depth training of the proposed system.

The training of the proposed system is depicted in Figure 3. It first converts the network into a graph and inputs it into the 2 layered GCN model along with a fully connected linear layer. After processing of the entire network, the trained GCN is able to recognize and detect anomalies.

The derived graph is passed through 2 GCN layers. Each layer generates a new feature representation that combines information from neighboring nodes. If a node's representation doesn't match the pattern extracted during training, the GCN labels it as abnormal.

For example, if the model is tested using four sensors, its initial dimension is four. The first layer expands the dimension to 16 (to optimize tensor operations), by utilizing learnable weight transformation. This enables the model to look for complex patterns between adjacent nodes in a larger feature space. The features at a node like Temperature will be influenced by its neighbor nodes.

$$h^{(0)} \in R^{4} \tag{1}$$

The initial feature vector at each node is represented by Equation (1), where $h^{(0)}$ denotes the raw input before any GCN transformation with dimension four.

$$h^{(1)} = \sigma\left(W^{(1)} \cdot h_{agg}^{(0)}\right),\ W^{(1)} \in R^{16\times 4} \tag{2}$$

After the first GCN transformation, we arrive at Equation (2) where, $h^{(1)}$ represents the first GCN layer where the dimension is expanded to 16. $W^{(1)}$ is the weight matrix of the first GCN layer. $h_{\text{agg}}^{(0)}$ denotes the aggregated features from 1-hop neighbors. It has 4 dimensions, showing the original 4 sensor features. σ(.) represents the ReLU (Rectified Linear Unit) activation function, which introduces non-linearity.

The second GCN layer aggregates information from 2-hop neighborhood. It then compresses the feature dimension to eight, removing the irrelevant information and thus, reducing overfitting of the model.

$$h^{(2)} = \sigma\left(W^{(2)} \cdot h_{agg}^{(1)}\right),\quad W^{(2)} \in R^{8\times 16} \tag{3}$$

Equation (3) represents the transformation which is visible after the second GCN transformation. Here, $h^{(2)}$ represents the second GCN layer where the dimension is reduced to 8. $W^{(2)}$ *is* the weight matrix of the second GCN layer. $h_{\text{agg}}^{(1)}$denotes the aggregated features from the first layer. It has 4 dimensions, showing the original 4 sensor features.

The output of the second layer is passed through a Fully Connected final linear layer. This layer reduces the dimension to 1, for the final anomaly prediction. The Equation (4) shows this transformation.

$$output = W_{fc} \cdot h^{(2)} + b_{fc},$$
$$W_{fc} \in R^{1\times 8}, \tag{4}$$
$$b_{fc} \in R,$$
$$\sigma(x) = max(0, x)$$

Here, W_{fc} is weight matrix of the final layer. $h^{(2)}$ represents node's feature vector from the second GCN layer (8-dimensional). b_{fc} is the Bias term that shifts the output to improve learning. This model can efficiently predict the labels by using adaptive thresholding

(dynamic thresholds) and can trigger an alarm if sensors values exceed or drop below these thresholds.

Thus, GCN assists in the detection of abnormal sensor behavior, sensor failure detection, and ensures smooth IoT system operation. Equation (5) shown below.

$$A = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} \quad (5)$$

During the representation of the four sensors, a cyclic graph is constructed based on the relationships between the four readings. The adjacency matrix that is generated, indicates the interdependence of all four sensors. This graph is fed into the GCN which analyzes the patterns accordingly over multiple epochs.

6 Results

Since the system learns iteratively, it refines the accuracy by analyzing each new incoming set of data submitted for processing. This enables the model to learn incrementally thus becoming increasingly proficient in pattern recognition [18]. This way, the system is able to detect compromised devices that exhibit anomalous behavior and identify malicious attacks [9] on the IoT network, thus improving security.

Table 1 presents the GCN's performance metrics, namely loss, accuracy, recall, and F1-score across training 50 epochs. The loss shows a gradual decrease over time suggesting effective backpropagation, while the accuracy, recall, and F1-score progressively increase during the epochs and stabilize nearing the end, indicating that the model learns significant patterns and generalizes well [10] to the data.

Table 1. Training of GCN in 50 epochs.

Epoch	Loss	Test Acc	Recall	F1
01	0.6858	0.6849	0.6757	0.6849
02	0.6727	0.7123	0.7027	0.7123
03	0.6619	0.7260	0.7027	0.7222
04	0.6417	0.8014	0.8649	0.8153
05	0.6110	0.8699	0.9054	0.8758
06	0.5716	0.9315	0.8649	0.9275
07	0.5223	0.9384	0.8784	0.9353
08	0.4655	0.9315	0.8649	0.9275
09	0.4150	0.9315	0.8649	0.9275
10	0.3670	0.9452	0.8919	0.9429
...	...	...	...	...
41	0.2028	0.9589	0.9189	0.9577
42	0.1951	0.9589	0.9189	0.9577
43	0.1983	0.9521	0.9189	0.9510
44	0.2133	0.9589	0.9189	0.9577
45	0.2068	0.9589	0.9189	0.9577
46	0.2030	0.9589	0.9189	0.9577
47	0.1963	0.9589	0.9189	0.9577
48	0.2041	0.9589	0.9189	0.9577
49	0.2074	0.9589	0.9189	0.9577
50	0.1908	0.9589	0.9189	0.9577

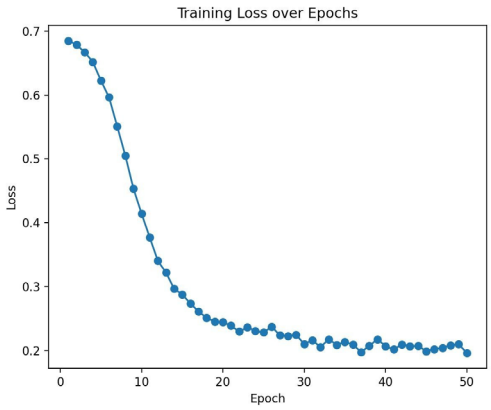

Fig. 4. Training loss over epochs.

The training loss over epochs is shown in Figure 4. The loss value initially starts around 0.7 and gradually decreases till it reaches an almost constant value of 0.2. This trend highlights the effective minimization of errors by using backpropagation. The low value of 0.2 demonstrates high learning capability of the model.

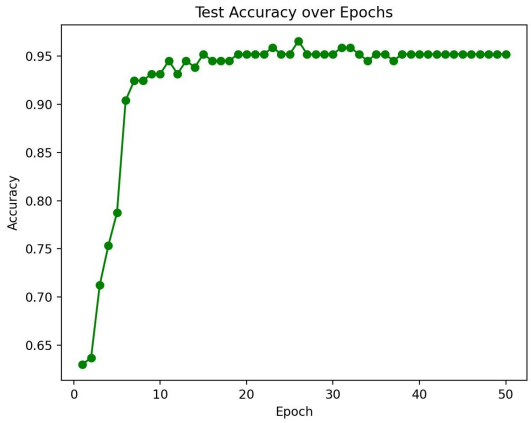

Fig. 5. Accuracy over epochs.

The accuracy over 50 epochs is depicted in Figure 5. The accuracy begins at a low value consistently improving till it reaches the peak value of 0.95 signaling that the model has achieved the maximum accuracy attainable. This trend illustrates the model's ability to maintain high accuracy over time.

After the training over 50 epochs, the GCN model achieved an overall accuracy of 95.89% with a high F1-score and recall. The final performance metrics is shown in Table 2. This demonstrates the superior detection ability to recognize anomalies in sensor readings.

Table 2. Model evaluation metrics.

Metric	Value
Accuracy	0.9589
Recall	0.9189
F1-Score	0.9577

A Multi-Layer Perceptron (MLP) [19] was trained on the same dataset, and was compared to the GCN model developed. Figure 6 demonstrates the comparative evaluation between the proposed model and MLP. The figure shows that the selected GCN model outperformed the MLP in all the measures. This verifies that the proposed GCN-based solution is both

suitable and viable for effective anomaly detection in IoT networks.

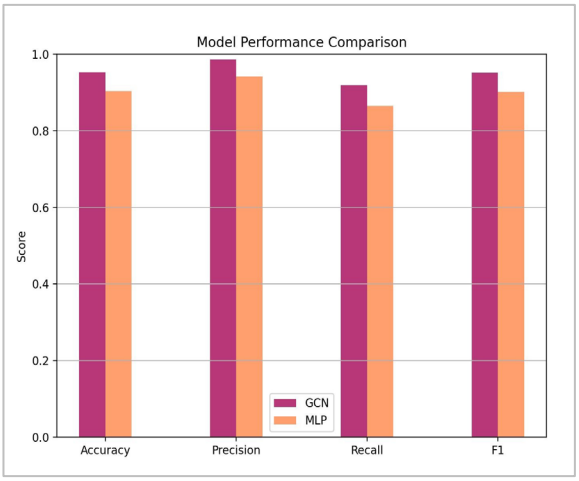

Fig. 6. Performance comparison of GCN and MLP.

7 Conclusion

In conclusion, the safeguarding of data in IoT networks is vital. This system provides an efficient solution to the challenges faced by low powered networks. The method employs correlation to dynamically represent relationships between nodes. The use of adaptive thresholding allows this GCN based architecture to achieve a higher degree of precision. This approach achieves a high level of performance and accuracy, while maintaining its light weight nature. The proposed method learns from previous behaviors of the connected devices to detect abnormal changes in their usual patterns. This allows it to identify malicious threats. However, while the proposed GCN based IoT security system is reliable and efficient, it may be obstructed in terms of computational load, especially in the case of low-powered devices. Its dependence on stored historical patterns hinders the threat detection mechanisms. Also, the interpretation of decisions taken by the model is a tough process. Future work can revolve around allowing only pre-approved behaviors and devices in IoT security system, ensuring the compliance with the CIA Triad. Additionally, upgrading scalability, explainability and robustness must be the larger objective of future modifications. By emphasizing the inter-sensor connections and features of sensors which are vital in the process of detection of anomalies, GNNExplainer or attention-based techniques possess the ability to drastically transform the values of model explainability. The innovative segment of this work lies in the combination of adaptive rolling-threshold labeling, correlation-weighted sensor graph creation and a lightweight two-layer GCN that aids in anomaly detection in the case of IoT data pertaining to multiple sensors. This is reflected in the effectiveness of the GCN's performance metrics, specifically accuracy, loss, recall, and F1-score, which in turn showcase the model's ability to learn patterns in an efficient manner leading to enhanced generalization capabilities. It can, therefore, be concluded that this strategy is highly fruitful and reproducible, designed to work even in situations where streams are continuous, unbalanced and noisy.

References

1. G. Kornaros, Hardware-assisted machine learning in resource-constrained IoT environments for security: Review and future prospective. IEEE Access **10**, 58603-58622 (2022). https://doi.org/10.1109/ACCESS.2022.3179047
2. P. Sharma, M. Kumar, H.K. Sharma, Robust GAN-based CNN model as generative AI application for deepfake detection. EAI Endorsed Trans. Internet Things **10**, (2024). https://doi.org/10.4108/eetiot.5637
3. M. Mesgaran, A.B. Hamza, A graph encoder–decoder network for unsupervised anomaly detection. Neural Comput & Applic. **35**, 23521-23535 (2023). https://doi.org/10.1007/s00521-023-08964-5
4. A.U. Ezeagwu, P.C. Onyia, O.S. Alagba, J.A. Nnamani, E.E. Uzywekwu, A comparative study of OSI and TCP/IP models in network communication, J. Softw. Eng. Simul. **7**, 08-14 (2021). https://doi.org/10.35629/9795-07060814
5. S. Li, Y.C. Tan, T. Vincent, High-pass graph convolutional network for enhanced anomaly detection: A novel approach. arXiv preprint arXiv:2411.01817, (2023)
6. M.A. ElMossallamy, H. Zhang, L. Song, K.G. Seddik, Z. Han, G.Y. Li, Reconfigurable intelligent surfaces for wireless communications: Principles, challenges, and opportunities. IEEE Trans. Cogn. Commun. Netw. **6,** 990-1002 (2020). https://doi.org/10.1109/TCCN.2020.2992604
7. Y. Pei, T. Huang, W. van Ipenburg, M. Pechenizkiy, RESGCN: Attention-based deep residual modeling for anomaly detection on attributed networks. Mach Learn. **111**, 519-541 (2022). https://doi.org/10.1007/s10994-021-06044-0
8. M. Aboubakar, M. Kellil, P. Roux, A review of IoT network management: Current status and perspectives. J. King Saud Univ. Comput. Inf. Sci. **34**, 4163-4176 (2022). https://doi.org/10.1016/j.jksuci.2021.03.006
9. P. Sharma, M. Kumar, H.K. Sharma, A generalized novel image forgery detection method using generative adversarial network. Multimed. Tools Appl. **83**, 53549-53580 (2024). https://doi.org/10.1007/s11042-023-17588-9
10. P. Sharma, S. Bagui, A. Kumar, S. Tripathi, Implementation of attention mechanism in GAN for the purpose of quality detection, in Proceedings of the IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT), IEEE, Greater Noida, India, April 08 (2024), 1081-1087
11. I. Ahmad, S. Shahabuddin, H. Malik, E. Harjula, T. Leppänen, L. Lovén, A. Anttonen, A.H. Sodhro,

M.M. Alam, M. Juntti, A. Ylä-Jääski, T. Sauter, A. Gurtov, M. Ylianttila, J. Riekki, Machine learning meets communication networks: Current trends and future challenges. IEEE Access **8**, 223418-223460 (2020). https://doi.org/10.1109/ACCESS.2020.3041765

12. M.A. Al-Shareeda, A.A. Alsadhan, H.H. Qasim, S. Manickam, The fog computing for internet of things: Review, characteristics and challenges, and open issues. Bull. Electr. Eng. Inform. **13**, 1080-1089 (2024). https://doi.org/10.11591/eei.v13i2.5555
13. F. Alwahedi, A. Aldhaheri, M.A. Ferrag, A. Battah, N. Tihanyi, Machine learning techniques for IoT security: Current research and future vision with generative AI and large language models. Internet Things Cyber-Phys. Syst. **4**, 167-185 (2024). https://doi.org/10.1016/j.iotcps.2023.12.003
14. S. Elkateb, A. Métwalli, A. Shendy, A.E.B. Abu-Elanien, Machine learning and IoT-based predictive maintenance approach for industrial applications. Alexandria Eng. J. **88**, 298-309 (2024). https://doi.org/10.1016/j.aej.2023.12.065
15. J. Bhayo, S.A. Shah, S. Hameed, A. Ahmed, J. Nasir, D. Draheim, Towards a machine learning-based framework for DDoS attack detection in software-defined IoT (SD-IoT) networks. Eng. Appl. Artif. Intell. **123**, 106432 (2023). https://doi.org/10.1016/j.engappai.2023.106432
16. A. Nazir, J. He, N. Zhu, A. Wajahat, X. Ma, F. Ullah, S. Qureshi, M.S. Pathan, Advancing IoT security: A systematic review of machine learning approaches for the detection of IoT botnets. J. King Saud Univ. – Comput. Inf. Sci. **35**, 101820 (2023). https://doi.org/10.1016/j.jksuci.2023.101820
17. S.S. Samaan, A.E. Korial, R.R. Sarra, A.J. Humaidi, Multilingual web traffic forecasting for network management using artificial intelligence techniques. Results Eng. **26**, 105262 (2025). https://doi.org/10.1016/j.rineng.2025.105262
18. J. Cámara, H. Muccini, K. Vaidhyanathan, Quantitative verification-aided machine learning: A tandem approach for architecting self-adaptive IoT systems. 2020 IEEE International Conference on Software Architecture (ICSA), IEEE, Salvador, Brazil, May 27 (2020), 11-22
19. S. Cherfi, A. Lemouari, A. Boulaiche, MLP-based intrusion detection for securing IoT networks. J. Ambient Intell. Hum. Comput. **33**, 20 (2025). https://doi.org/10.1007/s10922-024-09889-7