Journal of Multidisciplinary Engineering Science and Technology (JMEST)
ISSN: 2458-9403
Vol. 12 Issue 12, December - 2025

Hybrid CNN-GRU Model For Multi-Class Malicious Traffic Detection On MQTT-IOT-IDS2020-Based Networks Applicable To Weather Monitoring

Edemeka, Victor Usiere[1]
Department of Electrical/ Electronic Engineering
Akwa Ibom State Polytechnic, Ikot Osurua
Akwa Ibom State

Ogbu Ifeyinwa[2]
Advanced Space Technology Applications Laboratory Uyo,
National Space Research and Development Agency,
Federal Capital Territory, Abuja, Nigeria

Ohaga Blessing Chika[3]
Advanced Space Technology Applications Laboratory Uyo,
National Space Research and Development Agency,
Federal Capital Territory, Abuja, Nigeria

***Abstract*—The rapid proliferation of Internet of Things (IoT) devices has introduced significant security vulnerabilities, particularly in critical infrastructure like weather monitoring systems. This study proposes a hybrid multi-class deep learning model, combining Convolutional Neural Networks (CNN) and Gated Recurrent Units (GRU), to detect malicious traffic within MQTT-based IoT networks using the MQTT-IoT-IDS2020 dataset.The methodology leverages a 1D-CNN module to extract local spatial features and reduce data dimensionality, which is then integrated with a GRU module to capture temporal dependencies and sequential attack patterns. Designed specifically for the resource-constrained environment of weather monitoring, where sensors act as publishers transmitting environmental data to a central broker, the model addresses common threats such as DDoS, Scanning, and Brute Force attacks. By utilizing a Softmax activation layer, the system effectively classifies network traffic into multiple categories. Experimental results demonstrate exceptional performance, with F1-scores, precision, and recall consistently exceeding 96% across all traffic categories. The model displays high generalizability, evidenced by a balanced precision-recall ratio (e.g., 0.988/0.988 for Normal traffic), and significant robustness in identifying diverse network-level threats. Specifically, the model excels in detecting Aggressive and UDP Scans (~0.988 F1) and MQTT Brute Force attacks (0.979 F1), while maintaining low false-positive rates to ensure network usability for legitimate weather monitoring data. These findings establish the Hybrid CNN-GRU architecture as a superior solution for securing IoT-based environmental monitoring systems against evolving cyber threats.**

Keywords—MQTT-IoT-IDS2020 Dataset, CNN-GRU Model, MQTT Security, Hybrid Deep Learning, Weather Monitoring Systems, Multi-class Classification, Malicious Traffic Detection

1. Introduction

The rapid expansion of the Internet of Things (IoT) has revolutionized critical infrastructure, particularly in environmental monitoring and smart cities [1,2]. Weather monitoring systems now rely on a vast network of resource-constrained sensors to transmit real-time data such as temperature and humidity [3]. Central to this infrastructure is the Message Queuing Telemetry Transport (MQTT) protocol, which utilizes a lightweight publish-subscribe topology to ensure efficient communication over TCP/IP [4,5]. However, the inherent openness and resource-limited nature of these devices make them prime targets for sophisticated cyber threats [6,7].

Modern MQTT-based IoT networks are increasingly vulnerable to specialized attacks, including Denial of Service (DoS), DDoS, Scanning, and Brute Force [8]. Traditional signature-based Intrusion Detection Systems (IDS) often fail to detect novel or complex attack patterns in these high-dimensional, heterogeneous environments [9]. Consequently, there is an urgent need for robust, multi-class detection mechanisms that can identify malicious traffic with high precision while maintaining the efficiency required for IoT applications [10].

To address these challenges, this research proposes a hybrid multi-class CNN-GRU model specifically tailored for MQTT-IoT weather monitoring networks. By leveraging the MQTT-IoT-IDS2020 dataset, this study integrates two powerful deep learning architectures; the Convolutional Neural Networks (CNN) and the Gated Recurrent Units (GRU) model. The CNN is utilized to capture local spatial patterns and reduce dimensionality through 1D convolutional layers while the GRU is employed to process sequential data and capture long-term dependencies essential for identifying time-sensitive attack patterns. This hybrid approach enables the system to classify traffic into multiple categories—including DDoS, Scanning, and Brute Force—ensuring the reliability of weather monitoring data in the face of evolving cyber-physical threats.

The Hybrid CNN-GRU model is validated using the MQTT-IoT-IDS2020 dataset, which provides a realistic, specialized benchmark for MQTT protocol attacks [11,12]. The key contributions of this study are: (i) the development of a hybrid CNN-GRU model tailored for high-accuracy multi-class malicious traffic detection in IoT; (ii) the adaptation of this system to the specific context of MQTT-based weather monitoring frameworks; and (iii) a comprehensive performance evaluation demonstrating superior accuracy, precision, and F1-score compared to standalone models.

2. Methodology

2.1 The MQTT-IoT Weather Monitoring Network

The study details the implementation of a hybrid multi-class CNN-GRU model for detecting malicious traffic in MQTT-based IoT networks, with a specific focus on weather monitoring applications. The system operates within an MQTT-based IoT framework designed for weather monitoring. The architecture of the MQTT a typical network is shown in Figure 1. The MQTT (Message Queuing Telemetry Transport) network follows a publish-subscribe topology designed for resource-constrained IoT devices [13]. In a weather monitoring context, sensors (Publishers) transmit environmental data (such

as temperature, humidity) to a central MQTT Broker. The Broker then routes these messages to registered clients (Subscribers) [13, 14]. This architecture is lightweight and operates over TCP/IP, making it efficient but vulnerable to specialized attacks like message spoofing or Denial of Service (DoS) [13, 15]. Some of the components of an MQTT-IoT weather monitoring network are presented in Table 1.

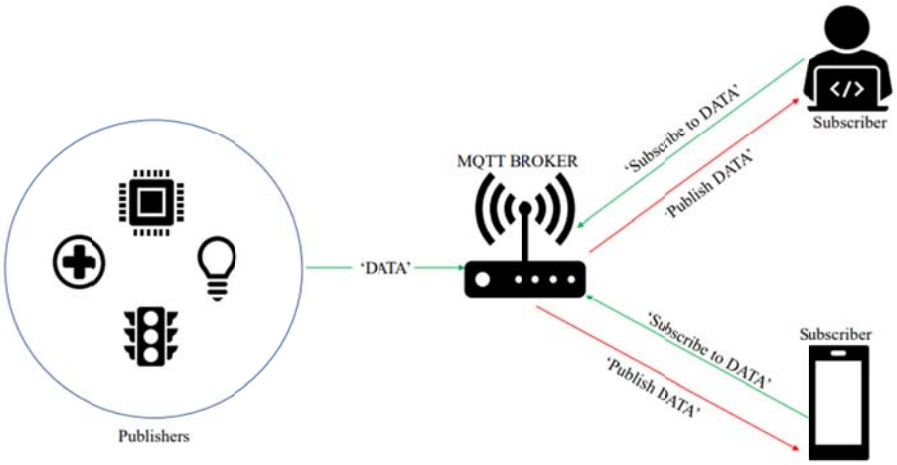

Figure 1. The Architecture of a Typical MQTT Protocol **[16]**

Table 1 The Components of an MQTT-IoT Weather Monitoring Network

Component	Description & Role	Examples
Sensors	Gather raw environmental data in real-time.	DHT11/DHT22 (Temp/Humidity), BMP180 (Pressure), Anemometer (Wind), Rain Gauge
Microcontroller	Processes sensor data and acts as the MQTT Client (Publisher).	NodeMCU ESP8266, ESP32, Arduino
MQTT Broker	The central hub that receives messages from publishers and routes them to subscribers.	Mosquitto, HiveMQ, EMQX, CloudMQTT
Subscribers	Applications or platforms that receive the data for display and storage.	Node-RED, ThingsBoard, Dashboards, Mobile App
Connectivity	Wireless network for transmitting data.	Wi-Fi, 4G, LoRa (for long-range)

2.2 The Hybrid CNN-GRU Model Structure

The model integrates two deep learning architectures to handle the complexity of IoT traffic. The CNN module uses 1D convolutional layers and max-pooling to identify local spatial patterns and reduce dimensionality while the GRU module which is the gated recurrent neural network processes sequential data to capture long-term dependencies and time-sensitive attack patterns. Then hybrid multiclass architecture enables the output from the CNN module to be fed into the GRU, followed by fully

connected layers and a Softmax activation to classify traffic into multiple attack categories (such as, DDoS, Scanning, Brute Force) or Normal traffic.

The architectural diagram of the CNN model is shown in Figure 2. Again, the architectural diagram of the GRU model is shown in figure 3 while the flow diagram for the hybrid CNN-GRU multiclass model is shown in Figure 4. The Convolutional Neural Network (CNN) layer is utilized for spatial feature extraction from the raw or preprocessed network traffic data [17, 18]. The convolutional layers is used to detect local patterns and significant features within the traffic flow [19, 20] while the pooling layers are used to reduce the dimensionality of the feature maps while retaining critical information [21, 22]. The Gated Recurrent Unit (GRU) is a streamlined version of an RNN that captures long-term temporal dependencies in the traffic sequences [24,24,25]. It uses two gates—the update gate and reset gate—to manage memory without the complexity of an LSTM, making it faster and more efficient for real-time IoT detection [24,24,25].

The hybrid model combines both strengths, while the CNN extracts high-level spatial features from the input data, which are then fed into the GRU to model the sequential and time-sensitive characteristics of the network traffic [24,24,25]. A final Softmax or fully connected layer performs the multi-class classification [24,25, 26].

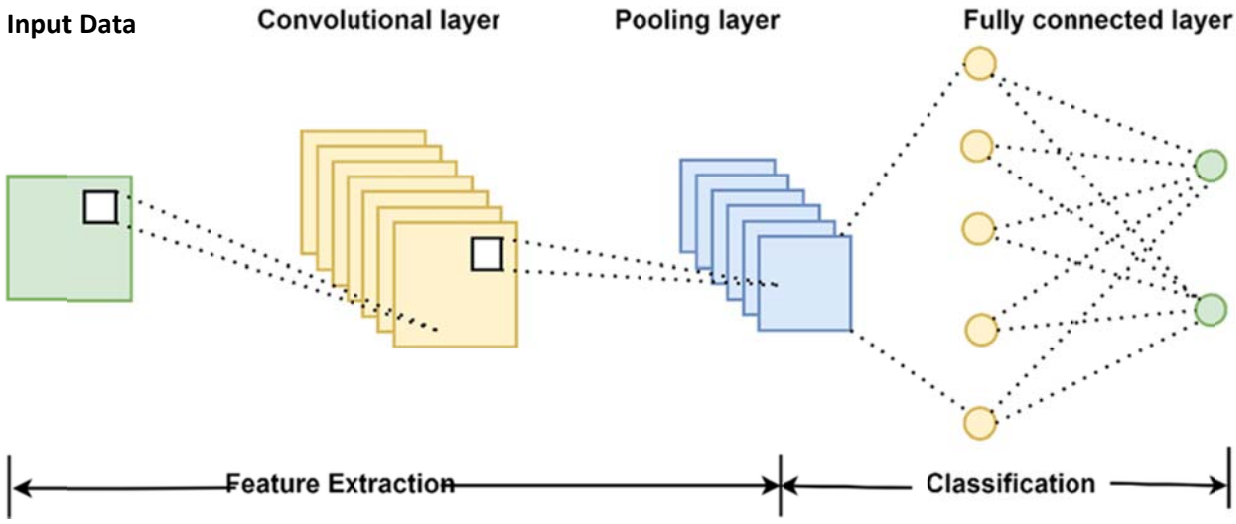

Figure 2 The architectural diagram of the CNN model [21]

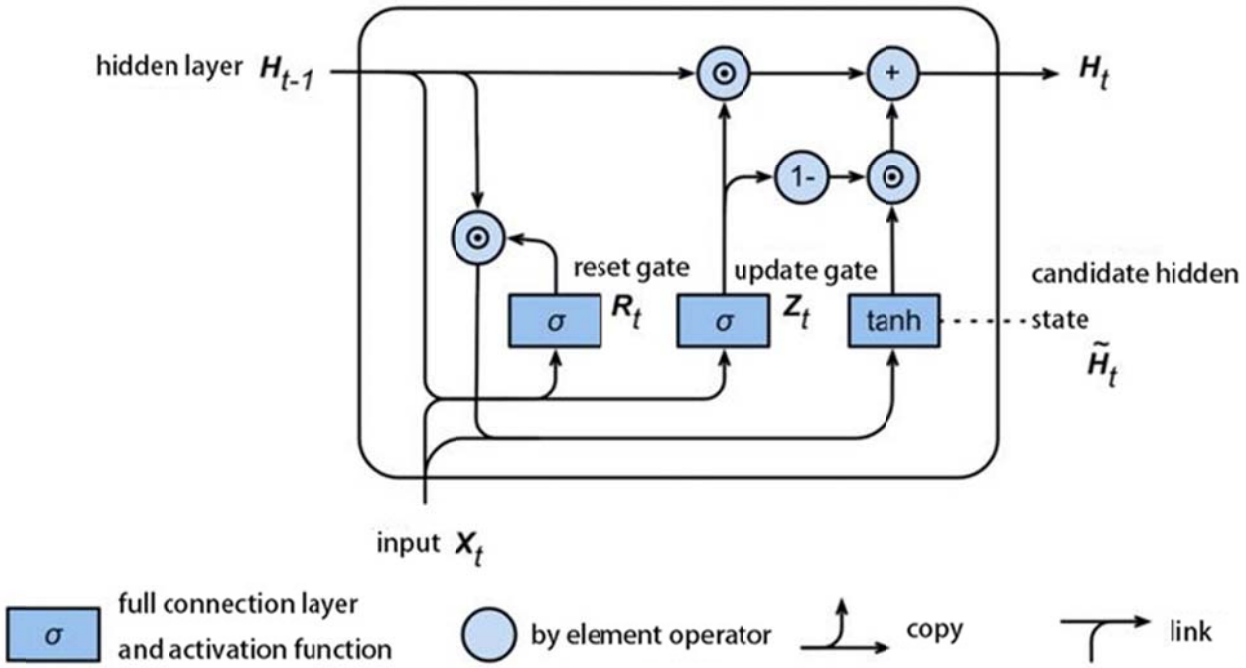

Figure 3 The architectural diagram of the GRU [27]

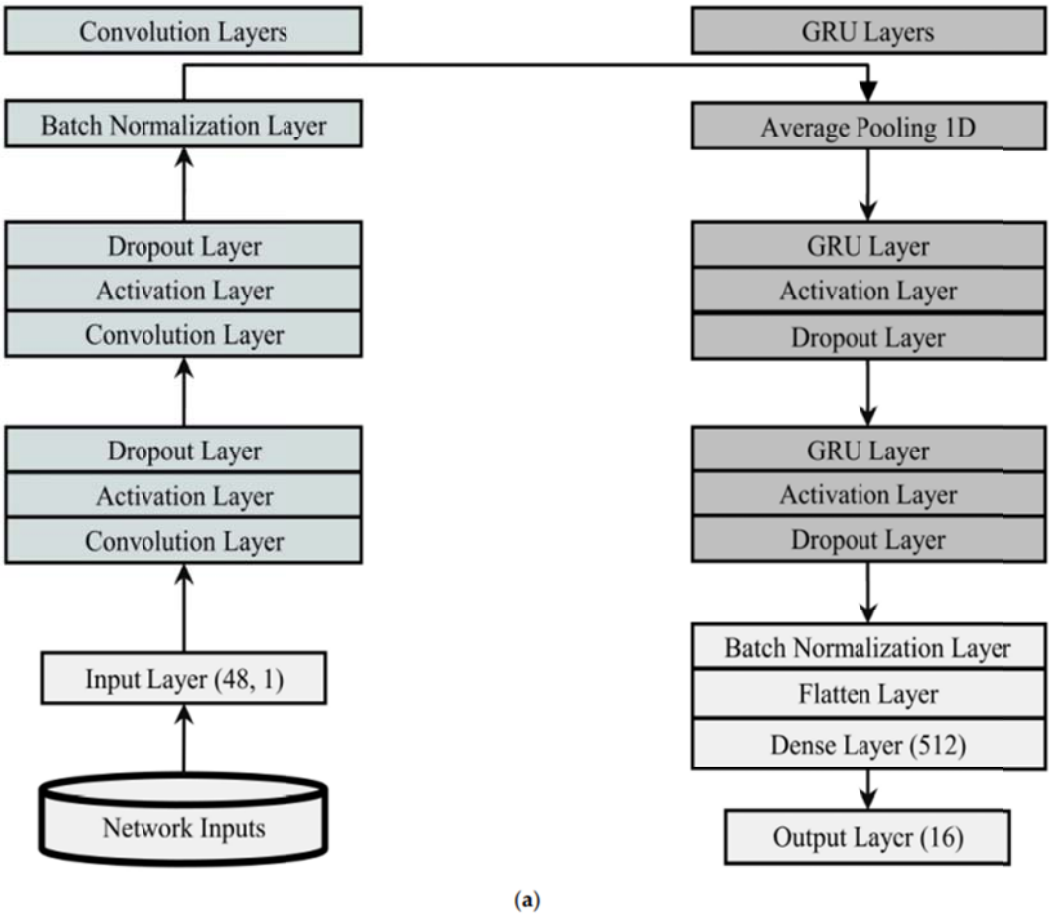

Figure 4 Flow diagram for the hybrid CNN-GRU multiclass model **[28]**

2.3 The case study dataset and its relevance for Weather Monitoring Network

The MQTT-IoT-IDS2020 dataset is a specialized, open-access dataset designed for evaluating Intrusion Detection Systems (IDS) in Internet of Things (IoT) environments that utilize the Message Queuing Telemetry Transport (MQTT) protocol. Developed by Hindy et al., it is one of the first datasets specifically focusing on MQTT-based attacks.

The MQTT-IoT-IDS2020 dataset is highly relevant for securing weather monitoring networks, as it specifically targets the MQTT protocol, which is the de facto standard for such IoT applications. It allows developers and researchers to build Intrusion Detection Systems (IDS) that can identify threats—such as MQTT broker, scanner, and brute-force attacks—designed to disrupt environmental data transmission.

The MQTT-IoT-IDS2020 dataset, which simulates an MQTT-based network for intrusion detection, includes five primary traffic scenarios (often labeled as five categories). These are categorized into normal operation and four distinct attack types, as shown in Table 2. The MQTT-IoT-IDS2020 is preprocessed following the steps summarized in Table 3. The flowchart of the hybrid CNN-GRU multiclass model is presented in Figure 5 while the performance evaluation metrics used to evaluate the model are presented in Table 4.

Table 2 The distribution of the packet-based data in the MQTT-IoT-IDS2020 dataset, aggregated with respect to the five traffic categories

Scenario/Class	Label	Number of Instances (Clean Data)
Normal (Baseline)	0	1,056,230 (approx.)
MQTT Brute Force	1	10,010,556 (approx.)
Aggressive Scan	2	40,488
UDP Scan	3	22,436
Sparta SSH Brute Force	4	19,728,943 (approx.)
Total	-	**~2,210,797** (Combined Scenarios)

Table 3 The Summary of the Preprocessing Steps for MQTT-IoT-IDS2020-based malicious traffic detection

Step	Technique	Description
1. Data Combination	CSV Aggregation	Merges multiple scenario-based files (e.g., standard operation, scanning, and brute-force) into a unified dataset for each flow level: Uni-flow, Bi-flow, or Packet-flow.
2. Data Cleaning	Deduplication & Null Handling	Removes duplicate records and addresses missing values or mislabeled entries (e.g., correcting "is attack" strings to 0 or 1) to improve data integrity.
3. Categorical Encoding	Label/One-Hot Encoding	Converts non-numerical features like source/destination IPs and attack types into integers. Common labels: 0 (Normal), 1 (Scan A), 2 (Scan sU), 3 (Sparta), 4 (MQTT BF).
4. Feature Selection	RF / RFE / Best First Search	Identifies the most relevant features (e.g., `num_pkts`, `num_psh_flags`) to reduce dimensionality and computational complexity.
5. Data Standardization	Min-Max Scaling	Scales numerical features (like inter-arrival times or packet lengths) to a standard range, typically [0, 1], to prevent bias from varying feature ranges.
6. Class Balancing	Oversampling / Undersampling	Balances the distribution between benign (typically ~76%) and malicious (~24%) traffic to avoid model bias toward the majority class.
7. Data Splitting	Train/Test Split	Typically involves an 80/20 **or** 70/30 split to ensure the model is evaluated on unseen data.

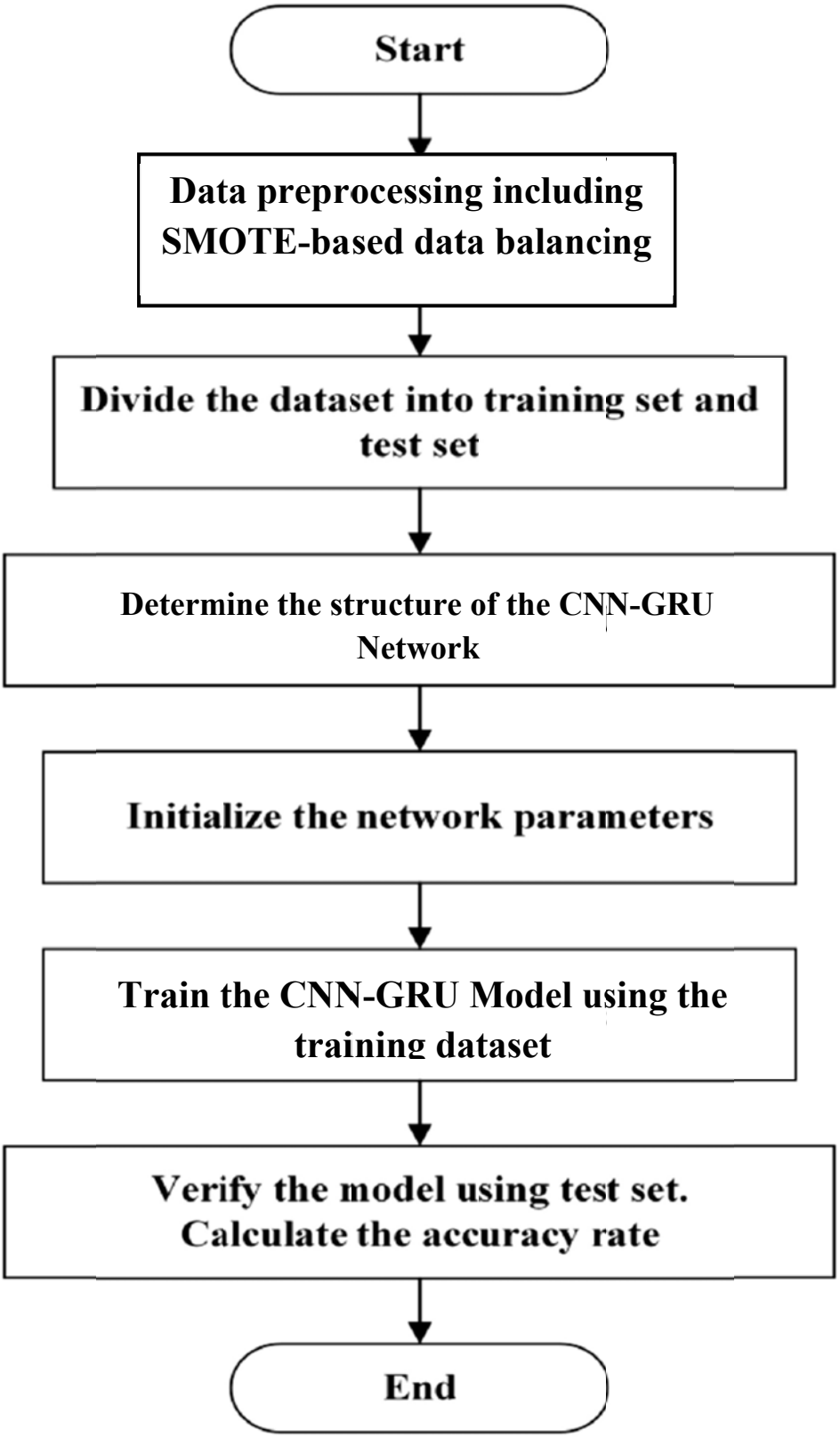

Figure 5 The flowchart of the hybrid CNN-GRU multiclass model

Table 4 Performance Evaluation metrics

SN	Metric	Formula
1	Recall	$\frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}$
2	Accuracy	$\frac{TN+TN}{TP+TN+FP+FN}$
3	Precision	$\frac{TP}{TP+FP}$
4	F1-Score	$\frac{2\times(Precision\times Recall)}{(Precision\times Recall)}$

3. Results and discussion

3.1 The results of the SMOTE Data balancing Approach

Based on the application of SMOTE (Synthetic Minority Over-sampling Technique) to balance the MQTT-IoT-IDS2020 dataset, which originally contained over 2 million records but suffered from significant class imbalance, the dataset is typically balanced by augmenting the minority classes to match the majority class. The class distribution after applying SMOTE to balance the training data is presented in Table 5 which typically resulted in an equalized number of samples per class. The pie chart for the imbalanced and the balanced datasets are presented in Figure 6 and Figure 7 respectively.

Table 5 The MQTT-IoT-IDS2020 Class Distribution After SMOTE

Class	Original Distribution Status	Samples After SMOTE (Approx.)
Normal	Majority Class	42,000
MQTT Brute Force	Minority Class	42,000
Aggressive Scan	Minority Class	42,000
UDP Scan	Minority Class	42,000
Sparta SSH Brute-force	Minority Class	42,000
Total Records (Balanced)	-	**210,000**

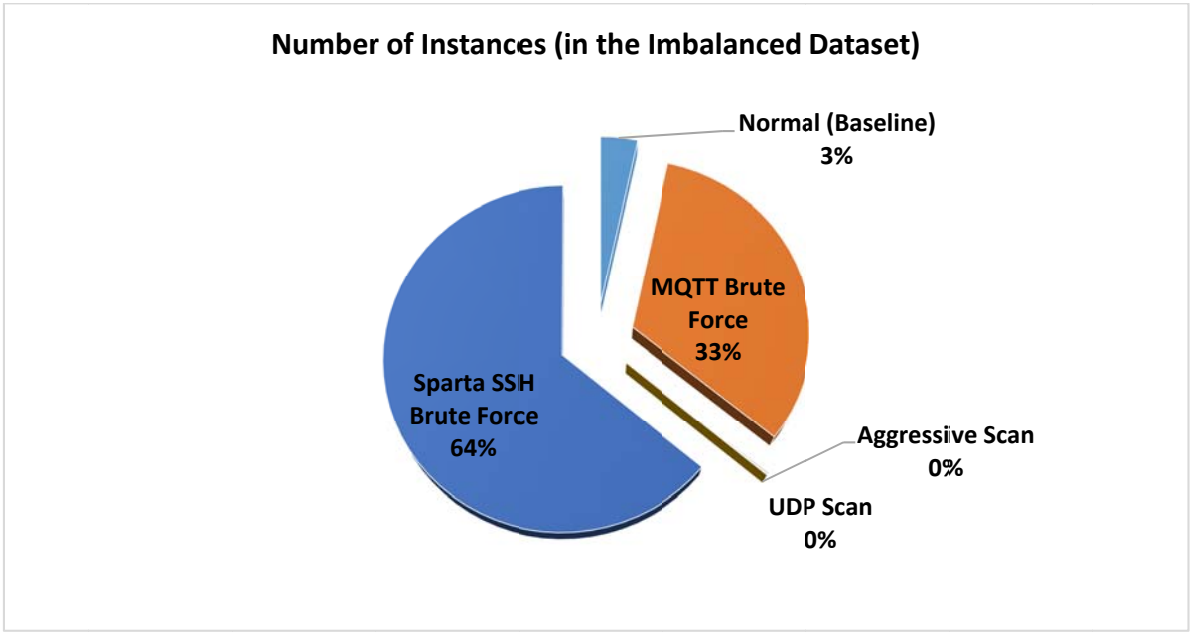

Figure 6 The class distribution before SMOTE data balancing was done

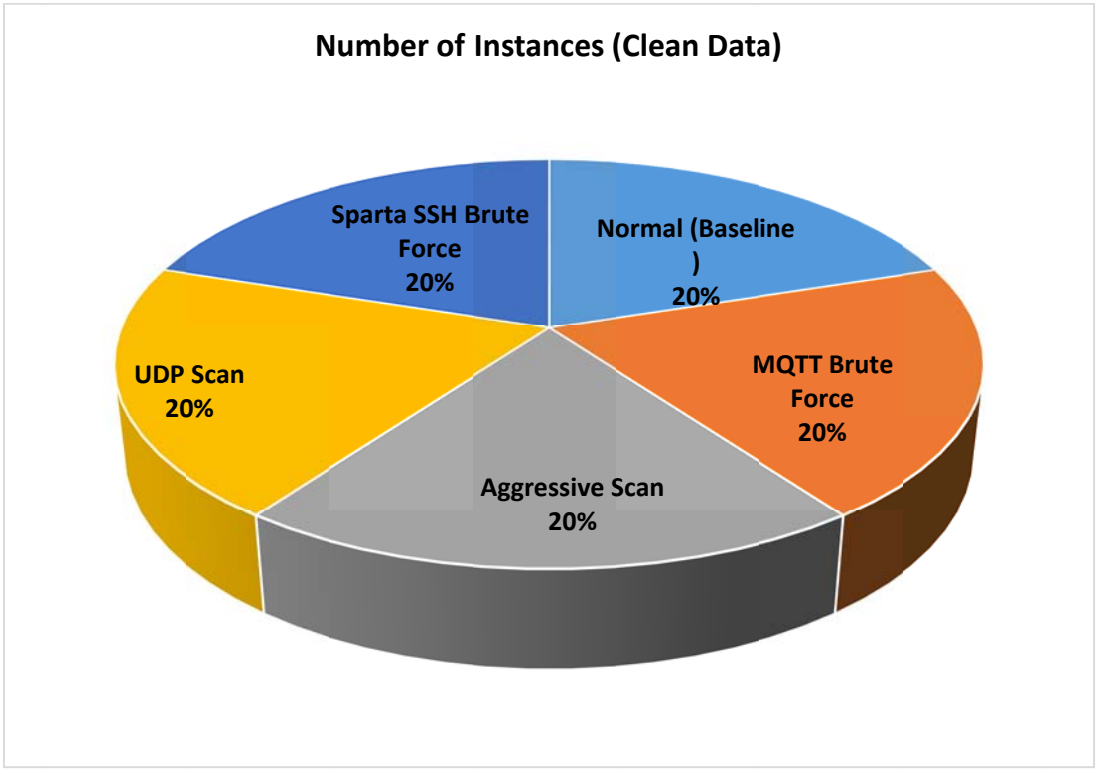

Figure 7 The class distribution after SMOTE data balancing was done

3.2 The performance of the hybrid CNN-GRU model for multi-class malicious traffic detection

The performance of the hybrid CNN-GRU model for multi-class malicious traffic detection is presented in Table 6 and Figure 8. The training loss and validation loss versus epoch is presented in

Figure 9 while Figure 10 shows the training accuracy and validation accuracy versus epoch. The multi-class malicious traffic detection model results indicate high-performing, stable, and reliable system across all evaluated traffic types. With F1-scores, precision, and recall consistently exceeding 96% for all classes, the model demonstrates high accuracy in both identifying malicious activity (Recall) and minimizing false alarms (Precision).

Generally, the models demonstrated high generalizability and robustness. The high generalizability is demonstrated by the close proximity of precision and recall scores (such as, Normal class 0.988 vs. 0.988) which suggests the model is not suffering from significant overfitting or extreme bias toward one type of error (false positives versus false negatives). The robustness is demonstrated by the fact that all the metrics are very high, which suggests that the model is adept at detecting various types of network-level attacks—both active scans and brute-force attempts—within the monitored environment.

The class-specific performance of the model is also very good. For the Normal class with 0.989 F1-score shows that there is excellent separation of legitimate traffic from malicious traffic. The 0.988 precision means very few legitimate connections are flagged as attacks (low False Positives), which is vital for network usability.

For the Aggressive Scan & UDP Scan (~0.988 F1), the model excels at identifying scanning behaviors. These attacks typically produce high-frequency packet patterns that are relatively easy to distinguish, which is reflected in the top-tier performance.

For the MQTT Brute Force (0.979 F1), while slightly lower than the scan detection, this is still a superior score. MQTT is often used in IoT scenarios, and this indicates the model effectively detects brute-force attacks on lightweight protocols.

For the Sparta SSH Brute-force (0.969 F1), the class shows the lowest performance, though it remains exceptionally high. It may indicate that the SSH brute-force attempts are more subtle, slower, or closely resemble "normal" encrypted traffic, making them slightly harder to distinguish than high-volume scanning, but still well-identified.

Furthermore, it can be stated that the model is not just accurate on the "Normal" class; it is consistently strong across all malicious categories, indicating good feature extraction. Also, the high precision across all malicious classes (such as, Sparta SSH at 0.967) implies the system can be trusted to trigger alerts without flooding administrators with false positives. In addition, the high recall means the model is catching most of the attacks, reducing the likelihood of a successful breach.

Table 6 The hybrid CNN-GRU model performance

Traffic Class	Precision	Recall	F1-Score	Accuracy (Avg)
Normal	0.988218	0.988911	0.988317	0.988218
MQTT Brute Force	0.978726	0.979412	0.978726	0.978628
Aggressive Scan	0.988119	0.988218	0.989208	0.988317
UDP Scan	0.98802	0.988317	0.988515	0.988416
Sparta SSH Brute-force	0.968448	0.969515	0.968545	0.968933

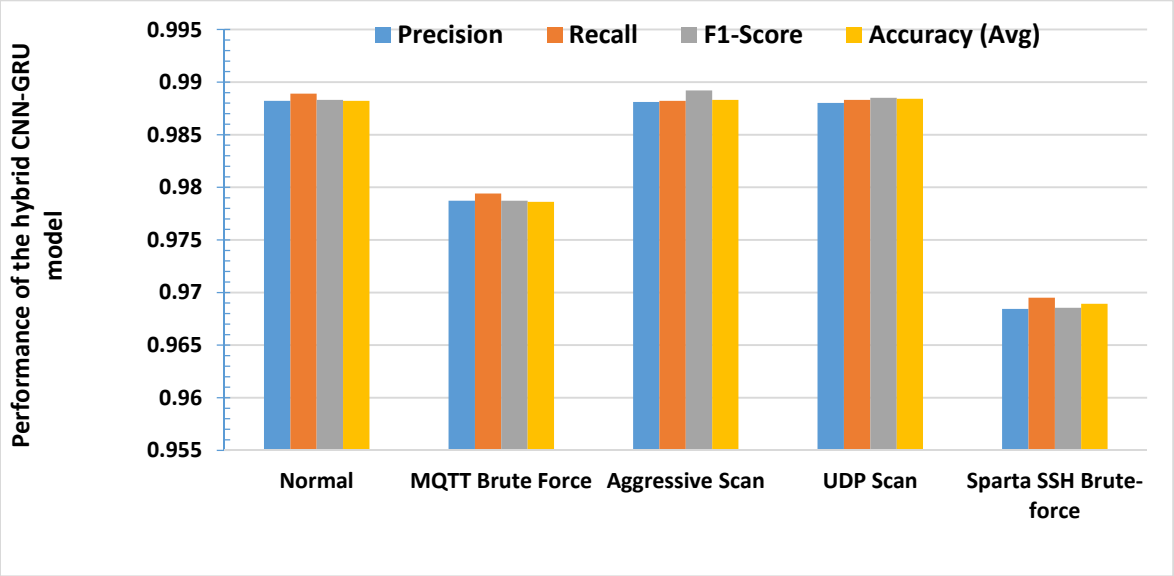

Figure 8 Performance of the hybrid CNN-GRU model

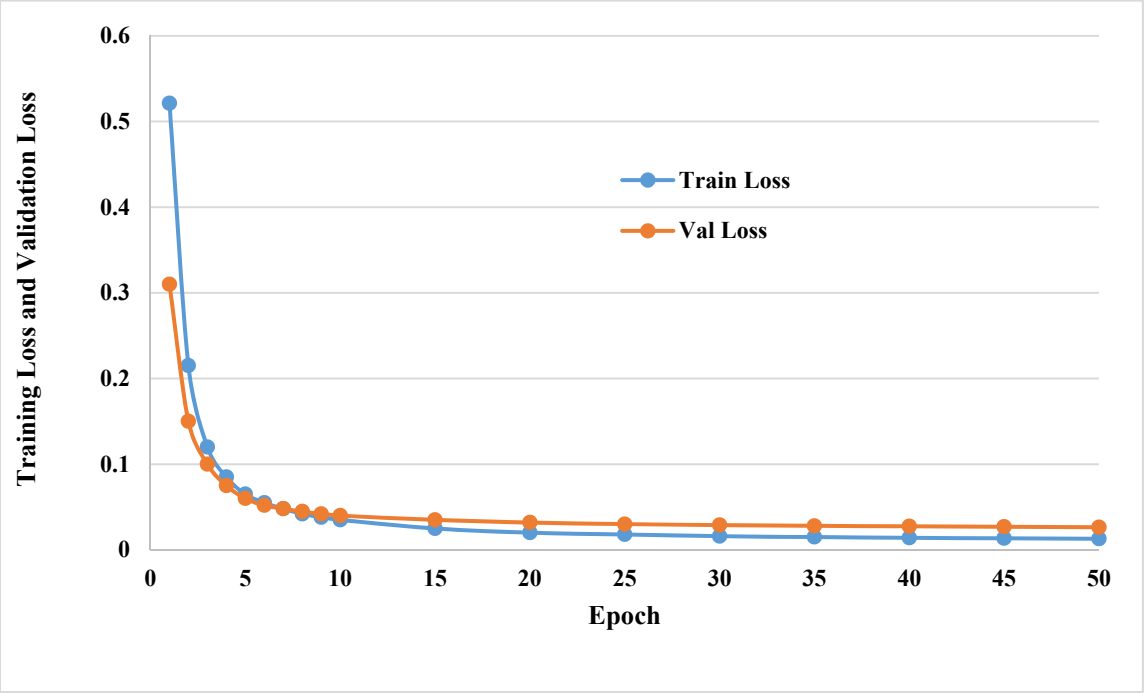

Figure 9 The Training Loss and Validation Loss Versus Epoch

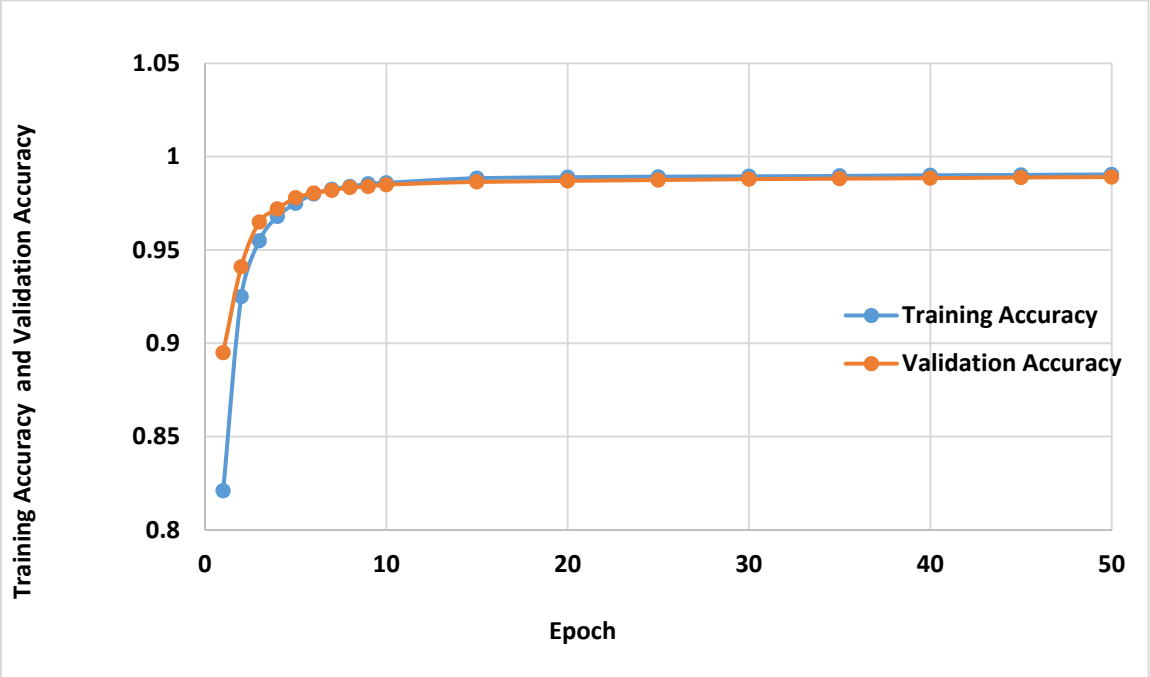

Figure 10 The Training Accuracy and Validation Accuracy Versus Epoch

4. Conclusion

This research successfully developed and evaluated a hybrid CNN-GRU model tailored for multi-class malicious traffic detection within MQTT-based IoT networks, specifically targeting applications in weather monitoring. By leveraging the spatial feature extraction capabilities of 1D Convolutional Neural Networks (CNN) and the temporal sequence processing of Gated Recurrent Units (GRU), the proposed architecture addresses the unique complexities of IoT traffic. The integration of these two deep learning techniques proved effective in identifying a wide range of network behaviors. The CNN module efficiently reduced dimensionality while capturing local spatial patterns, and the GRU module managed long-term dependencies with greater computational efficiency than traditional LSTM units. This synergy allows the model to accurately classify traffic into multiple categories—including DDoS, Scanning, Brute Force, and Normal traffic—providing a robust security layer for sensitive weather monitoring infrastructure.As IoT-based weather monitoring becomes increasingly critical for climate data integrity, the efficiency and real-time detection capabilities of the CNN-GRU model offer a scalable solution for securing MQTT protocols against evolving cyber threats.

References

1. Van Hoang, T. (2024). Impact of integrated artificial intelligence and internet of things technologies on smart city transformation. *Journal of technical education science*, *19*(Special Issue 01), 64-73.

2. Song, T., Cai, J., Chahine, T., & Li, L. (2021). Towards smart cities by Internet of Things (IoT)—a silent revolution in China. *Journal of the Knowledge Economy*, *12*(2), 1-17.
3. Arif, M., Maya, J. A., Anandan, N., Pérez, D. A., Tonello, A. M., Zangl, H., & Rinner, B. (2024). Resource-efficient ubiquitous sensor networks for smart agriculture: a survey. *IEEE Access*, *12*, 193332-193364.
4. Muchiri, I. N., Kiplimo, Y. C., & Aoki, S. (2024). Implementation of Message Queuing Telemetry Transport Protocol in Model Rocket.
5. Radwan, N. M., & Alves-Foss, J. (2024). MQTT in Focus: Understanding the Protocol and Its Recent Advancements. *International Journal of Computer Science and Security (IJCSS)*, *18*(1), 1-14.
6. Kornaros, G. (2022). Hardware-assisted machine learning in resource-constrained IoT environments for security: review and future prospective. *IEEE Access*, *10*, 58603-58622.
7. Amgbara, S. I., Akwiwu-Uzoma, C., & David, O. (2024). Exploring lightweight machine learning models for personal internet of things (IOT) device security. *ResearchGate Preprint*, *24*.
8. Syed, N. F. (2020). IoT-MQTT based denial of service attack modelling and detection.
9. Otoum, Y., & Nayak, A. (2021). As-ids: Anomaly and signature based ids for the internet of things. *Journal of Network and Systems Management*, *29*(3), 23.
10. Pantelakis, V., Bountakas, P., Farao, A., & Xenakis, C. (2023, August). Adversarial machine learning attacks on multiclass classification of iot network traffic. In *Proceedings of the 18th International Conference on Availability, Reliability and Security* (pp. 1-8).
11. Hindy, H., Bayne, E., Bures, M., Atkinson, R., Tachtatzis, C., & Bellekens, X. (2020, September). Machine learning based IoT intrusion detection system: An MQTT case study (MQTT-IoT-IDS2020 dataset). In *International networking conference* (pp. 73-84). Cham: Springer International Publishing.
12. Rakha, M. A., Khan, I. U., Ouaissa, M., Ouaissa, M., & Ayub, M. Y. (2024). Hybrid Model for IoT-Enabled Intelligent Towns Using the MQTT-IoT-IDS2020 Dataset. In *Cyber Security for Next-Generation Computing Technologies* (pp. 159-176). CRC Press.
13. NWANKWO, E. I. (2021). *INTEGRATION OF MESSAGE QUEUE TELEMETRY TRANSPORT PROTOCOL AND CONSTRAINED APPLICATION PROTOCOL FOR DATA COMMUNICATION IN WIRELESS SENSOR NETWORKS* (Doctoral dissertation).
14. Odejobi, O. D., Hammed, N. I., & Ahmed, K. S. (2020). IoT-Driven Environmental Monitoring Model Using ThingsBoard API and MQTT.

15. Murkomen, T. (2024). Performance, privacy, and security issues of TCP/IP at the application layer: A comprehensive survey. *GSC Advanced Research and Reviews*, *18*(3), 234-264.
16. Thantharate, A., Beard, C., & Kankariya, P. (2019, July). Coap and mqtt based models to deliver software and security updates to iot devices over the air. In *2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)* (pp. 1065-1070). IEEE.
17. Ranjan, N., Bhandari, S., Zhao, H. P., Kim, H., & Khan, P. (2020). City-wide traffic congestion prediction based on CNN, LSTM and transpose CNN. *Ieee Access*, *8*, 81606-81620.
18. Chen, G., Guo, Y., Zeng, Q., & Zhang, Y. (2023). A novel cellular network traffic prediction algorithm based on graph convolution neural networks and long short-term memory through extraction of spatial-temporal characteristics. *Processes*, *11*(8), 2257.
19. Ma, D., Song, X., & Li, P. (2020). Daily traffic flow forecasting through a contextual convolutional recurrent neural network modeling inter-and intra-day traffic patterns. *IEEE Transactions on Intelligent Transportation Systems*, *22*(5), 2627-2636.
20. Ren, Y., Zhao, D., Luo, D., Ma, H., & Duan, P. (2020). Global-local temporal convolutional network for traffic flow prediction. *IEEE Transactions on Intelligent Transportation Systems*, *23*(2), 1578-1584.
21. Gholamalinezhad, H., & Khosravi, H. (2020). Pooling methods in deep neural networks, a review. *arXiv preprint arXiv:2009.07485*.
22. Akhtar, N., & Ragavendran, U. (2020). Interpretation of intelligence in CNN-pooling processes: a methodological survey. *Neural computing and applications*, *32*(3), 879-898.
23. Salem, F. M. (2021). Gated RNN: the gated recurrent unit (GRU) RNN. In *Recurrent neural networks: from simple to gated architectures* (pp. 85-100). Cham: Springer International Publishing.
24. Zhang, Y., & Fill, H. D. (2024). TS-GRU: A stock gated recurrent unit model driven via neuro-inspired computation. *Electronics*, *13*(23), 4659.
25. ArunKumar, K. E., Kalaga, D. V., Kumar, C. M. S., Kawaji, M., & Brenza, T. M. (2022). Comparative analysis of Gated Recurrent Units (GRU), long Short-Term memory (LSTM) cells, autoregressive Integrated moving average (ARIMA), seasonal autoregressive Integrated moving average (SARIMA) for forecasting COVID-19 trends. *Alexandria engineering journal*, *61*(10), 7585-7603.
26. Mateus, B. C., Mendes, M., Farinha, J. T., Assis, R., & Cardoso, A. M. (2021). Comparing LSTM and GRU models to predict the condition of a pulp paper press. *Energies*, *14*(21), 6958.

27. Mishra, V., & Kane, L. (2023). A survey of designing convolutional neural network using evolutionary algorithms. *Artificial Intelligence Review*, *56*(6), 5095-5132.
28. Qianmin, S., Wei, P., Xiaoqiong, C., Hongxing, L., & Jihan, H. (2023). COVID-19 clinical medical relationship extraction based on MPNet. *IET Cyber-Physical Systems: Theory & Applications*, *8*(2), 119-129.
29. Ullah, I., Ullah, A., & Sajjad, M. (2021). Towards a hybrid deep learning model for anomalous activities detection in internet of things networks. *IoT*, *2*(3), 428-448.