Real-Time Network Traffic Classification in IoT Networks using Hybrid AI Techniques

Farzam Rezaei, Jorge E. López de Vergara
Escuela Politécnica Superior, Universidad Autónoma de Madrid,
Madrid, Spain
farzam.rezaei@estudiante.uam.es, jorge.lopez_vergara@uam.es

***Abstract*—The development of IoT across various fields emphasizes the need for real-time network traffic classification to maintain security, simplify resources, and address evolving threats. Traditional methods like port-based and deep packet inspection fail with encrypted traffic, privacy constraints, and slow processing, while deep learning solutions face limitations with insufficient data and latency in IoT environments. To overcome these challenges, this research presents a hybrid AI framework that seamlessly combines AI techniques for fast and accurate classification. Through a five-phase process—data preparation, feature engineering, model design, interpretability and optimization, and deployment and validation—it merges synthetic traffic generation and semi-supervised methods to improve data scarcity. Optimized deep neural networks and GAN networks will be used for classification and anomaly detection. Moreover, the results will be enhanced by applying explainable artificial intelligence for transparency. This framework aims to improve latency and accuracy, outperforming current approaches.**

***Index Terms*—IoT Network Traffic Classification, Real-Time Classification, Feature Engineering, Anomaly Detection, Machine Learning**

I. Introduction

The Internet of Things (IoT) has developed rapidly by linking many devices in medical, industrial, smart cities, and consumer electronics. This advancement has also resulted in an increase in different types of network traffic, which brings forth distinct issues in management, security, and resource utilization [1], [2].

Port-based and payload-based techniques are today inadequate in facing traffic classification in the ever-changing, complex, and limited IoT environments. For example, port-based traffic classification suffers from dynamic ports using problems. Additionally, payload-based techniques are computationally slow and incompetent in addressing privacy issues. Moreover, if network traffic is encrypted, this is no longer an option. Therefore, this situation calls for better and more applicable techniques [2], [3].

Deep learning (DL) has emerged as a transformative solution, offering the ability to learn complex patterns directly from raw network traffic data. By leveraging models like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), DL enables the effective classification of network traffic, even in the face of IoT-specific challenges such as intermittent connectivity and diverse communication protocols. Moreover, DL approaches can integrate spatial and temporal dependencies, making them well-suited for the classification of traffic patterns common in IoT environments [4], [5].

The classification of IoT network traffic is not only vital for optimizing network performance, but also necessary for enhancing security. Efficient traffic classification supports anomaly detection, intrusion prevention, and the enforcement of security policies, safeguarding IoT systems against potential vulnerabilities and threats. Furthermore, as IoT adoption continues to scale, traffic classification enables targeted resource allocation and quality-of-service (QoS) mechanisms, improving both operational efficiency and user experience [4], [6].

Despite its promise, deploying DL for IoT traffic classification faces several challenges. These include the computational demands of deep learning models, the scarcity of labeled datasets for IoT-specific applications, and the "black-box" nature of DL models, which complicates explainability and accountability. Addressing these challenges requires innovative methodologies that balance model performance, resource efficiency, and interpretability [1], [4], [7].

Recent research highlights using Generative Adversarial Networks (GANs) for traffic classification and anomaly detection. GANs, an unsupervised model, consist of a generator creates realistic data and a discriminator judging it, improving through a competitive process. They have gained attention for two reasons: they can generate challenging anomalous data and learn normal system patterns, making them effective at spotting outliers. This dual capability makes GANs a promising tool for anomaly detection, though challenges remain [8].

This research sets out to address these challenges by creating a powerful machine learning framework tailored for classifying IoT network traffic. By utilizing cutting-edge architectures and real-world datasets, the study aims to push the boundaries of network management and security within the IoT space.

The remainder of this paper is organized as follows. Section II surveys the related work in the network traffic analysis and IoT networks. Section III clarifies the weaknesses of the existing methods and their implications for research.

This Ph.D. work is partially funded by a grant from the Department of Electronics and Communication Technologies at Universidad Autónoma de Madrid, as well as by the Community of Madrid, Spain, under the project RAMONES-CM (TEC-2024/COM-504).

Section IV presents the main research questions. Section V outlines our initial methodology, followed by a comprehensive approach. Section VI concludes the paper, highlighting and showing future directions in this research.

II. Related work

In recent years, various approaches have been used by researchers in IoT networks analysis. Early network traffic classification used port numbers to identify protocols, based on standards set by the Internet Assigned Numbers Authority (IANA). Ports are divided into system (0–1023), user (1024–49 151), and dynamic (49 152–65 535) ranges, with 0–1023 being standard for protocols like HTTP (port 80) or DNS (port 53). This method checks TCP and UDP packet port numbers to classify traffic. Studies found it only 70% accurate [2]. This method is simple, fast, and uses little computing power but only identifies protocols, not specific applications.

The deep packet inspection (DPI) method, also called signature-based identification, fixes problems with port-based classification by not relying on port numbers, avoiding issues like masking and random ports. Instead, it looks at the payload (content) of network traffic to identify the applications behind it. It uses signature patterns like characters or bits taken from packets and matches them to a library of known application signatures. This makes it more accurate than port-based methods, even with non-standard ports [1], [2]. DPI looks at packet content for better accuracy and app detection. However, it is slow, needs many resources, cannot spot new applications without updated signatures, and raises privacy issues. Moreover, it is not applicable to encrypted traffic.

The statistical classification method uses statistics and machine learning (ML) to sort network traffic. First, it groups packets into flows based on five header fields: source and destination IP addresses, source and destination port numbers, and the protocol (TCP or UDP). Next, it pulls out statistic features. At the packet level, it looks at single or grouped packets to get things like packet size and time between packets. At the flow level, it checks the whole flow for features like total packets, total bytes, and how long the flow lasts. The solutions split into three types: supervised, unsupervised, and semi-supervised learning [2].

Supervised learning analyzes traffic flow stats for high accuracy without needing to deal with packet content. It is easy to set up, but it relies on quality training data and expert feature selection. Unsupervised learning skips labeled data, reducing effort but with lower accuracy. Semi-supervised learning combines both, improving accuracy but requiring more computing power.

Deep learning (DL) methods, notably Neural networks (NN), are systems of connected units, called neurons, that process information based on inputs. These neurons are linked, and each link has a weight adjusted using a method like back-propagation. Deep learning is a type of NN with many layers. Common deep learning tools for classifying network traffic include Multi-Layer Perceptrons (MLP), Recurrent Neural Networks (RNN), Auto Encoders (AE), and Convolutional Neural Networks (CNN). Recently, researchers have used deep learning to sort network traffic into applications or services. It takes raw traffic as input to spot spatial or temporal patterns, skipping the need to manually pick features like older methods, making it easier and not requiring deep technical knowledge [9]. Deep Learning skips feature selection and gives high accuracy with fine detail. It needs lots of data and computing power, especially for training.

CNN, a popular deep learning type, finds spatial patterns and works well in image recognition. It has also been applied to recognize network traffic patterns [10]. It turns an image into a pixel vector, uses a filter to slide over it step-by-step, and pulls out features without losing key details. These features go to a pooling layer to shrink their size, then to a fully connected NN for classification. Researchers use CNN for spatial traffic patterns and RNN for time-based patterns, feeding in either raw traffic or statistics from flows [9].

Generative Adversarial Networks (GANs) play versatile roles in anomaly detection, categorized as generating abnormal data, generating both normal and abnormal data, learning normal system behavior, learning both normal and abnormal behavior, and learning complex data distributions. These roles split into data augmentation and representation learning, with GANs excelling at creating synthetic data to address scarcity and imbalance, especially for rare anomalies, and modeling normal patterns for outlier detection [11]. In networking, GANs leverage their generator-discriminator duo to detect deviations, often focusing on normal data due to imbalance challenges [12].

III. Gaps and Challenges

Real-time network traffic classification in IoT networks faces significant constraints due to the unique constraints of IoT environments and the limitations of existing methods. Below are the key challenges impacting effective classification:

- **Limited Computational Resources:** IoT devices, such as sensors, have restricted processing power and memory, yet machine learning (ML) and deep learning (DL) models demand substantial resources. While tools like model compression help, energy-efficient deployment remains difficult, hindering real-time performance on edge devices [13].
- **Latency in Real-Time Processing:** DL models often require significant computation, leading to delays in classification that disrupt time-sensitive IoT tasks, such as immediate threat detection, reducing operational efficiency [13].
- **Lack of Interpretability:** DL models operate as black boxes, obscuring decision-making processes. In IoT applications like security monitoring, where transparent and trustworthy decisions are critical, this lack of clarity limits adoption [13].
- **Device Heterogeneity:** IoT devices vary widely in hardware, protocols, and operating systems, complicating the

development of universal classification solutions. Tailored adaptations for diverse devices increase complexity [14].

- **Dynamic Threats:** Dynamic and emerging cyber threats, like novel malware, require classification systems to continuously learn. Current methods often rely on manual updates, slowing response times and weakening security [14].
- **Different Traffic Patterns:** IoT traffic patterns vary depending on the environment. For instance, IoT traffic patterns in smart city use cases feature four main types: periodic, normal aperiodic, payload exchange, and event-driven patterns [15]. The classification model needs training on all these IoT traffic types, each with varying complexity, to perform effectively.
- **Dependence on Labeled Data:** Supervised learning achieves high accuracy but relies on extensive labeled datasets, which are rare in IoT due to labeling costs. Unsupervised methods lack precision, and semi-supervised approaches require careful tuning [2].

These challenges highlight the need for innovative, hybrid AI solutions that balance accuracy, speed, and adaptability to advance real-time traffic classification in IoT networks.

IV. Research questions

Considering the gaps mentioned earlier, this research aims to address the following research questions:

1) How can deep learning be made faster and lighter for real-time traffic classification?
2) How can CNNs, RNNs, Transformers, and GANs be effectively combined into a hybrid deep learning framework to boost accuracy and strength in classifying IoT network traffic?
3) How can Explainable AI (XAI) techniques be added to the framework to ensure clear understanding and improve classification accuracy?
4) Which setup, edge, fog, cloud, or hybrid is the best mix of speed, computing efficiency, and scalability for IoT traffic classification?
5) How can Generative Adversarial Networks (GANs) simulate tricky traffic scenarios to improve the framework's anomaly detection abilities?

V. Methodology

This research will develop a hybrid machine learning framework focusing on real-time IoT traffic classification and anomaly detection. The methodology unfolds in five phases, combining generative AI (GAN, LLM) for generating synthetic data, semi-supervised methods to improve data scarcity and labeling, deep neural networks (DNNs) to develop our model, and explainable AI (XAI) for interpretability. The goal is to create a scalable, interpretable and efficient system that addresses the data complexity, heterogeneity of devices, and dynamic threats of IoT.

A. Data Collection and Preprocessing

First of all, we focus on gathering and refining diverse IoT traffic and log datasets to ensure a solid foundation for the hybrid framework. By collecting real-world data and enhancing it with synthetic traffic and logs, this phase addresses the challenges of data complexity and scarcity in IoT.

- **Dataset Gathering:** Collect public IoT datasets (e.g., CICIoT2023 [16], CICAPT-IIoT [17], UNSW-TON-IoT [18]) and real-time traffic and logs from simulated IoT networks.
- **Synthetic Data:** Train a GAN model on normal traffic to generate synthetic flows, including rare anomalies. The generator creates fake traffic; the discriminator validates realism, producing balanced datasets with normal and anomalous samples.

  Use an LLM to generate synthetic traffic data, such as packet sequences or metadata.

B. Feature Engineering

In this phase, we focus on feature engineering to extract and select the most relevant traffic features for real-time IoT network classification. By refining raw and synthetic data into an optimized feature set, this phase increases model efficiency and accuracy under IoT resource constraints.

1) Extract statistical features from raw and LLM-generated traffic at packet level (e.g., length, inter-arrival time) and flow level (e.g., total bytes, duration).
2) Perform feature selection to identify the most relevant features for classification and anomaly detection.
3) Validate feature quality for generated traffic to ensure synthetic data aligns with real patterns.

C. Hybrid Model Design

In this phase, we develop a hybrid framework for real-time IoT traffic classification and anomaly detection using machine learning and deep learning techniques.

1) Apply semi-supervised learning techniques for traffic classification, combining a small labeled dataset with a larger unlabeled set. Initial clustering groups traffic patterns, followed by self-training with labeled samples to propagate labels (e.g., ”normal” vs. ”anomaly”) across unlabeled flows. This method adapts to IoT's limited labeled data while maintaining real-time performance.
2) Integrate deep learning models include Convolutional Neural Networks (CNNs) for spatial pattern recognition, Recurrent Neural Networks (RNNs) for temporal sequence analysis, and Autoencoders (AEs) for feature learning from engineered traffic features. CNNs process packet-level features, RNNs analyze flow sequences, and AEs reconstruct patterns to support semi-supervised classification, using real and synthetic generated traffic data. These models ensure accurate classification with minimal labeled data.
3) Use Generative Adversarial Networks (GANs) for anomaly detection by modeling benign traffic distributions with selected features. GANs train on benign

flows, with the discriminator identifying deviations as anomalies, and the generator refining synthetic traffic for training. This enhances security without classifying traffic types directly.

4) Optimize the hybrid model for real-time execution using lightweight architectures.

D. Optimization and Interpretability

This phase concentrates on optimizing the hybrid framework for real-time performance while ensuring interpretability in IoT traffic classification. By refining model efficiency and implanting explainable AI techniques, this phase balances speed, resource use, and transparency for practical deployment.

1) Integrate explainable AI (XAI) techniques to interpret the impact of extracted features on classifications and anomaly detections. XAI analyzes our model outputs, producing explanations for operational transparency.
2) Reduce model complexity through compression techniques applied to our model. Reducing weights in the neural network used in our model ensures edge-compatible efficiency with inference times.
3) Tune hyperparameters (e.g., learning rates, GAN weights) for fast inference.

E. Real-Time Deployment and Testing

This phase focuses on deploying and testing the optimized framework in real-time IoT environments to evaluate its performance. By simulating various scenarios, this phase ensures the system's efficiency, robustness, and suitability for practical network traffic classification.

1) Implement the optimized framework on edge devices, fog nodes, and cloud platforms within simulated IoT networks.
2) Evaluate performance metrics including accuracy, F1-score, anomaly detection rate, latency, and resource consumption. Metrics assess classification precision and speed across real and synthetic generated traffic, ensuring operational efficiency.
3) Robustness test with encrypted traffic, adversarial input, and anomalies.

VI. Conclusions and Further Research

This research introduced a hybrid AI framework for real-time traffic classification in IoT networks, combining generative and discriminative techniques to tackle the complexities of dynamic network environments. The framework will effectively process traffic patterns under strict time constraints by integrating synthetic data generation with advanced classification methods. Developed through a multi-phase approach—data preparation, feature extraction, model design, optimization, deployment, and testing. This aims to provide efficient and accurate classification across diverse IoT environments, enhancing with interpretability tools and optimized for low-latency execution, the system exceeds traditional methods, proving its value for real-time applications, like security monitoring and network management in IoT ecosystems.

Future studies could optimize traffic classification by refining hybrid techniques for lower latency and higher accuracy, leveraging quantum machine learning (QML) to enhance feature selection and model efficiency in IoT networks.

References

[1] H. Tahaei, F. Afifi, A. Asemi, F. Zaki, and N. B. Anuar, "The rise of traffic classification in IoT networks: A survey," *Journal of Network and Computer Applications*, vol. 154, 3 2020.

[2] A. Azab, M. Khasawneh, S. Alrabaee, K. K. R. Choo, and M. Sarsour, "Network traffic classification: Techniques, datasets, and challenges," *Digital Communications and Networks*, vol. 10, pp. 676–692, 6 2024.

[3] M. Lotfollahi, M. J. Siavoshani, R. S. H. Zade, and M. Saberian, "Deep packet: a novel approach for encrypted traffic classification using deep learning," *Soft Computing*, vol. 24, pp. 1999–2012, 2020. [Online]. Available: https://doi.org/10.1007/s00500-019-04030-2

[4] A. Thakkar and R. Lohiya, "A review on machine learning and deep learning perspectives of IDS for IoT: Recent updates, security issues, and challenges," *Archives of Computational Methods in Engineering*, vol. 28, pp. 3211–3243, 6 2021.

[5] G. D'Angelo and F. Palmieri, "Network traffic classification using deep convolutional recurrent autoencoder neural networks for spatial–temporal features extraction," *Journal of Network and Computer Applications*, vol. 173, p. 102890, 1 2021.

[6] F. Ullah, S. Ullah, G. Srivastava, and J. C.-W. Lin, "IDS-INT: Intrusion detection system using transformer-based transfer learning for imbalanced network traffic," *Digital Communications and Networks*, vol. 10, pp. 190–204, 2 2024.

[7] E. Luis-Bisbé, V. Morales-Gómez, D. Perdices, and J. E. López de Vergara, "No pictures, please: Using explainable artificial intelligence to demystify cnns for encrypted network packet classification," *Applied Sciences*, vol. 14, p. 5466, 6 2024.

[8] W. Lim, K. S. C. Yong, B. T. Lau, and C. C. L. Tan, "Future of generative adversarial networks (GAN) for anomaly detection in network security: A review," *Computers & Security*, vol. 139, p. 103733, 2024.

[9] T. Sasi, A. H. Lashkari, R. Lu, P. Xiong, and S. Iqbal, "An efficient self attention-based 1d-cnn-lstm network for iot attack detection and identification using network traffic," *Journal of Information and Intelligence*, 9 2024.

[10] T. Shapira and Y. Shavitt, "Flowpic: Encrypted internet traffic classification is as easy as image recognition," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2019, pp. 680–687.

[11] M. Sabuhi, M. Zhou, C.-P. Bezemer, and P. Musilek, "Applications of generative adversarial networks in anomaly detection: A systematic literature review," 2021. [Online]. Available: https://arxiv.org/abs/2110.12076

[12] M. Zhao and Y. Zhang, "GAN-based deep neural networks for graph representation learning," *Engineering Reports*, vol. 4, p. e12517, 2022.

[13] S. Sai, M. Kanadia, and V. Chamola, "Empowering IoT with generative AI: Applications, case studies, and limitations," *IEEE Internet of Things Magazine*, vol. 7, pp. 38–43, 5 2024.

[14] F. Alwahedi, A. Aldhaheri, M. A. Ferrag, A. Battah, and N. Tihanyi, "Machine learning techniques for IoT security: Current research and future vision with generative ai and large language models," *Internet of Things and Cyber-Physical Systems*, vol. 4, pp. 167–185, 2024.

[15] A. S. Ibrahim, K. Y. Youssef, H. Kamel, and M. Abouelatta, "Traffic modelling of smart city internet of things architecture," *IET Communications*, vol. 14, pp. 1275–1284, 5 2020.

[16] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment," *Sensors*, vol. 23, 7 2023.

[17] E. Ghiasvand, S. Ray, S. Iqbal, S. Dadkhah, and A. A. Ghorbani, "CICAPT-IIOT: A provenance-based apt attack dataset for iiot environment," 7 2024. [Online]. Available: https://arxiv.org/abs/2407.11278

[18] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-IoT dataset," 2018. [Online]. Available: https://arxiv.org/abs/1811.00701