Energy-Efficient Hybrid IDS for IoT Using Autoencoder and Random Forest

M. Abdullah Ashraf
Dept. of Information and Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
mabdullah5599183@gmail.com

Hashir Imran
Dept. of Information and Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
ihashir007@gmail.com

Aoun Muhammad
Dept. of Electrical and Electronics Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
aoun.muhammad@iub.edu.pk

Sana Tariq
Dept. of Information and Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
sana.tariq@iub.edu.pk

***Abstract*— In recent years , The use of Internet of Things (IoT) devices increase rapidly which point of the challenges related security, computational complexity and power consumption. Traditional intrusion detection systems in cyber physical systems often fail to provide efficient and scalable solutions for resource-constrained environments. This paper proposed an intrusion detection system (IDS) based on hybrid model (Autoencoder and Random) and this model hybrid model is energy efficient. The Autoencoder is used for feature reduction to minimize computational complexity, while the Random Forest classifier is employed for accurate threat detection. The hybrid model is trained using IoT datasets and achieve high detection accuracy 97.73% and inference time is 0.1458 seconds for 20,000 records.Experimental results shows that our hybrid model reduces computational cost and improve energy efficiency and maintaining detection performance also. Therefore, the proposed model is suitable for intrusion detection in resource-constrained IoT environments.**

I. Introduction

A. Overview of IoT and Security Challenges

The Internet of Things (IoT) is the network of physical devices embedded with sensors and software which are connect with each other and exchange data over the internet. [1]. In the Fourth Industrial Revolution IoT will play significant role and become innovative technology [2]. Wani et al. [3] predict that 29.42 bilion IoT devices active now a days and the number of active devices raise from 29.42 billion to 30 billion.

There are many challenges in IoT development and deployment that are: (i) Computational complexity (ii) Energy consumption (iii) Security Challenges (iv) Energy Efficient Lightweight Models and algorithms (v) Cyber attacks (vi) Big data handling and heterogeneous environment [4]. Due to these challenges, IoT systems become vulnerable to cyber attacks. Possible attacks include channel attacks, session hijacks, routing attacks, flooding [5],Malicious activities, zero-day attacks, malware, ransomware, distributed denial of service (DDoS), phishing attacks, unauthorized access, data breaches, and cryptographic attacks [6]

B. Limitations of Traditional Intrusion Detection Systems

Intrusion Detection System (IDS) is a system software that monitors the network traffic (data,controls) and actions, detect any suspicious or malicious traffic which behavior deviates from the traditional or custom security system designs [7]. IDS techniques are mainly signature-based and rule-based which are works only predefined attack patterns that are stored in datasets used to train these models so they are unable to detect new or zero day attack.Many IDS solutions require continuous updates (to get latest knowledge about new types of attacks), high computational power, and large memory resources, making them unsuitable for resource-constrained IoT devices mainly remote devices. Figure 1 Describes the types of IoT attacks (threats) [8].

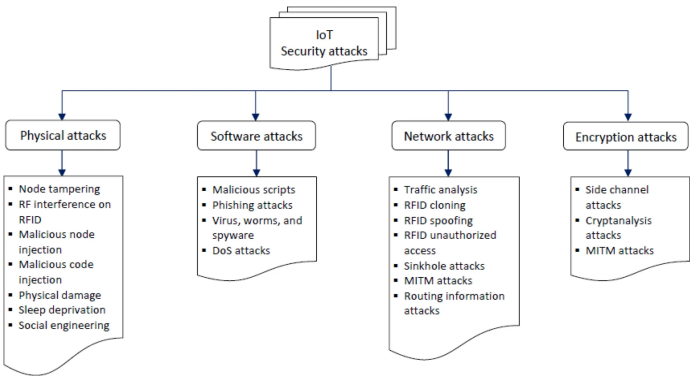

Fig. 1: Types of IoT attacks (threats)

C. Machine Learning and Hybrid IDS Approaches

AI techniques like Machine learning and Deep learning is used in Intrusion Detection Systems (IDS) to achieve accuracy

DOI: https://doi.org/10.67535/tsp.000003.016
Published under the ICACNC 2026 Proceedings

in detecting malicious traffic using rule based techniques [9]. Using ML(Random Forest) and DL(Autoencoder) models IDS can detect new attacks and Zero-day attacks, so ML and DL have become vital at this point [10].

D. Proposed Energy-Efficient Hybrid IDS

To address the limitations of most of the available approaches, this paper proposes a new energy efficient hybrid intrusion detection system (IDS) for IoT. The proposed model combines an Autoencoder for feature reduction with a Random Forest classifier for accurate threat detection,low computational cost, and less power consumption suitable for IoT devices with limited resoures .

The Autoencoder reduces network traffic through removing extra features , which lowers the computational cost and energy consumption.Then Random Forest analyzes the network traffic pattern and detects normal or malicious activities.

II. Related Works

A. Traditional and Machine Learning-Based IDS

Most real time IDS are signature-based systems [11] and rule-based systems which can not detect new and zero day attacks [12].To overcome these issues , ML models such as SVM, Decision Trees and Random Forest are used and these models improve threat detection and can detect and identify unknown patterns which look suspicious .However, these models increase computational cost and power consumption which can be a problem.

B. Deep Learning and Hybrid IDS Approaches

Deep learning models, such as Deep Neural Network (DNN) [13], Autoencoder [14], and Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN) and their hybrid models are now being widely used in IDS due to their very good quality that they can learn from complex datasets and improving detection. However, these models need high computational and power consumption again limitation.

C. Future Research Directions

Although existing ML and DL based IDS models achieve high detection accuracy, the only problem is their high computational cost and energy consumption. As IoT environments are harsh ,remote so they require lightweight and adaptive security solutions due to resource constraints. Furthermore, most existing hybrid models have main goal to improve accuracy rather than making balance between energy efficiency and performance. This creates a research gap for designing a new lightweight hybrid IDS solutions that can operate efficiently in these IoT environments.

III. Methodology

A. System Architecture Overview

The proposed system is an energy-efficient hybrid based on two models intrusion detection framework designed for IoT environments. It consists of data preprocessing [15], feature reduction using Autoencoder [16], and classification with the help of Random Forest. The objective is to reduce computational cost(low processing power) while maintaining high detection accuracy.The objective is to detect attacks, reduce power consumption and computational complexity.

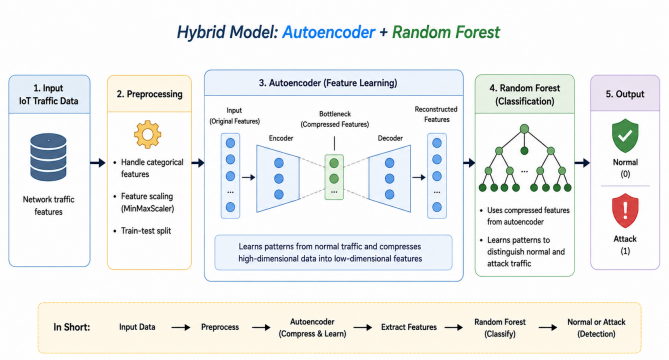

Fig. 2: Workflow of Hybrid Model(Autoencoder+Random Forest

B. Data Preprocessing

Tawakuli et al. [17] stated "Data preprocessing is the set of operations that convert raw data into good quality data for input such as dimensionality reduction, normalization of numerical features and handling missing value".

Let the dataset be represented as:

$$\{y_1, y_2, ..., y_n\}$$

Each input sample is normalized using Min Max scaling:

$$z' = \frac{z - z_{\min}}{z_{\max} - x_{\min}}$$

C. Feature Reduction Using Autoencoder

Autoencoder is a deep learning model which compress(high dimensional data into low dimensional latent) the input using encoder and reconstructs original input using decoder. [18].

Encoder function:

The encoder compress the input data $\mathbf{x}$ to a latent space represented by a mean μ and a standard deviation σ:

$$\mu, \sigma = f_\phi(\mathbf{x})$$

where f_ϕ is the neural network parameterized by ϕ.

Decoder Function:

The decoder reconstructs the input data from the latent variable:

$$\hat{x} = g_\theta(z)$$

where g_θ is the neural network parameterized by θ.

Loss Function:
Loss function is a function of machine learning models use to find out the difference between actual and predict target values. For autoencoder it is known as mean square error which minimize the reconstruction error between actual value y_i and predict value $\hat{y}_i$ [19].

D. Classification Using Random Forest

In 2001 Leo Breiman proposed Random Forest algorithm [20]. Random Forest consist of multiple decision trees collect the data from autoencoder combine data and detect if the traffic is normal or suspecious.

$$\hat{z} = \text{mode}\left(s_1(y), s_2(y), \ldots, s_t(y)\right)$$

where i shows the decision tree, $s_i(y)$ is the prediction of the decision trees and t is the total number of trees.

E. Energy Efficiency Consideration

Energy consumption is equal to the computation and computation time.Energy efficiency increases definitely if the computation and time reduce.

$$E \propto C_{\text{compute}} \times T_{\text{processing}}$$

where E represents energy consumption, C_{compute} represents computational cost, and $T_{\text{processing}}$ is processing time.

Using Autoencoder in this new system reduces computational complexity and processing time, thus reducing energy consumption, and makes the system much more energy efficient.

IV. EXPERIMENTAL SETUP

A. Dataset Description

The proposed model is trained using the TON_IoT dataset (train_test_network.csv), and evaluated on the UNSW_2018_IoT_Botnet_Full5pc_3.csv test dataset [21] and download it from the website "https://research.unsw.edu.au/projects/toniot-datasets" which is publicly available here for research purposes.To make the models more efficient and reduce energy consumption, we adopted preprocessing and feature selection strategies inspired by [22] and [23] .It contains mostly malicious and normal traffic data,multiple attack categories,network flows which train the models to detect attacks accurately efficiently.

B. Data Splitting Strategy

The Dataset is Divided into two categories inspiring by [24], [25] and [26], First is training data is a set of data which is used to train the proposed model, and second is testing data is a set of data used to test the proposed model and told about model accuracy.

- 80% Training dataset
- 20% Testing dataset

C. Model Training Configuration

We make energy efficient Hybrid IDS for IoT devices, the hybrid model is configured with lightweight parameters best for iot devices.

- Autoencoder: Autoencoder configured with feature reduction, compressing the high dimensional input space (40 features) to latent space of 15 features , and model was trained only for 30 epochs.

TABLE I: **Sensitivity Analysis of Latent Space Dimensionality**

Latent Dimension	Validation Loss (MSE)
1	0.87961
5	0.24180
10	0.06554
15	0.01185
20	0.02475

A sensitivity analysis was performed on the latent space dimension, and the results demonstrate that 15 latent features achieve the best balance between reconstruction accuracy and feature compression. Consequently, a latent dimension of 15 was selected as the optimal configuration.

- Random Forest: Autoencoder performed feature reduction and then Random Forest works and detects attack.As we made lightweight models, we use the less number of decision trees to increase efficiency, performance and stability.

D. Evaluation Metrics

Evaluation Metrics use to calculate the performance of the models by measuring Accuracy, Precision, Recall, and F1-score to get results about the detection performance. Inference Time [27] using to measure computational complexity and energy efficiency.

Accuracy measure the percentage of correct predictions out of all predictions , Precision tell me about how many actually attacks are performed out of attack predictions and Recall is that how many actual attack is detected by model out of all attacks.

E. Implementation Environment

The training phase of hybrid model performed by google colab to make sure model train efficiently using cloud based computational resources. Real time testing and GUI deployment performed by local system(Hp,Intel Core i5-6th generation,8 GB RAM).Python 3.11 used to develop and train models.

Deep Learning Model (Autoencoder) developed using python libraries TensorFlow and Keras, and Machine Learning Model (Random Forest) developed by library Scikit-learn (v1.6.1) and Data Preprocessing techniques also performed by this library. Pandas and NumPy libraries used for data manipulation, and a real-time monitoring dashboard devoloped by Tkinter library.In VS Code proposed models test by using different dataset and the accuracy is 97.73%.

V. RESULTS AND DISCUSSION

A. Classification Performance

This section evaluates the detection capability of the proposed hybrid intrusion detection system using Autoencoder and Random Forest. The performance is analyzed using training loss, classification metrics, and confusion matrix analysis.

1) Autoencoder Training Analysis: Figure 3 disclose Autoencoder training loss and validation loss on multiple epochs.Pattern of Blue and Orange lines shows model is accurate, quickly and successfully learned, not overfitting data and also stable.

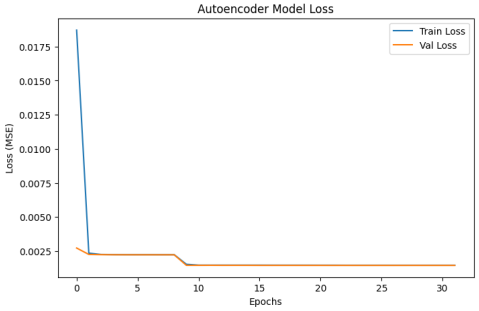

Fig. 3: Autoencoder Training Loss

2) Classification Metrics Evaluation: The proposed model achieved high performance with an overall accuracy of 97.73% . The classification report includes precision, recall, and F1-score for both normal and malicious traffic.

TABLE II: Classification Report of the Proposed Hybrid IDS

Traffic Class	Precision	Recall (TPR)	F1-Score	Support
Normal Traffic	0.99	0.97	0.98	10,000
Attack Traffic	0.90	0.99	0.95	2,500
Overall Accuracy				**97.73%**

3) Confusion Matrix Analysis: Figure 4 shows the Confusion Matrix of the proposed model and results shows model predict correct traffic with very few false predictions.

TABLE III: Confusion Matrix of the Proposed Hybrid IDS

	Predicted Normal (0)	Predicted Attack (1)
Actual Normal (0)	9,725	275
Actual Attack (1)	9	2,491

Five independent runs were performed to evaluate model stability. The proposed Hybrid IDS achieved 97.81% ± 0.04% accuracy, 99.77% ± 0.08% TPR, 2.68% ± 0.04% FPR, and 0.9361 ± 0.0011 MCC, demonstrating consistent performance across multiple runs.

B. Computational Efficiency

A key objective of this research is energy efficiency. The model processed a test batch of 12,500 network records in just **0.1458 seconds**. The average inference time per sample was 0.0073 milliseconds, indicating high-speed real-time processing capability and proving that the proposed

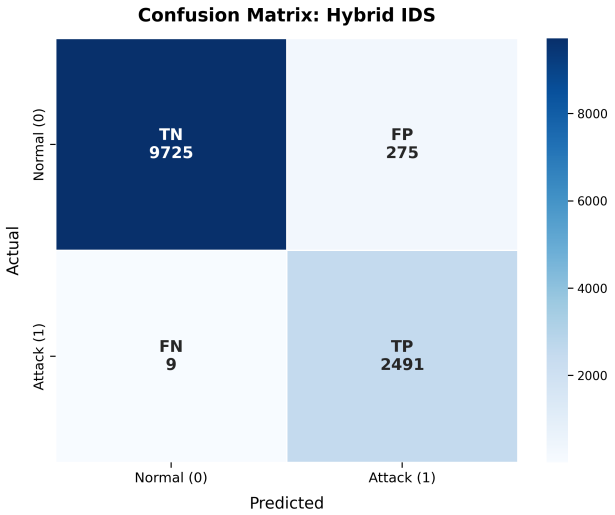

Fig. 4: Confusion Matrix of Proposed Model

lightweight Autoencoder and Random Forest combination is highly efficient and perfectly suited for resource-constrained IoT environments.Less computation,less CPU usage and less power consumption.

TABLE IV: Hardware and Energy Profiling Results

Metrics Evaluated	Empirical Value Captured
Total Pipeline Inference Time	2.3880 Seconds
Latency per Network Packet	0.191037 ms/packet
Peak CPU Utilization	94.10%
Net Memory (RAM) Allocated	0.0547 MB
Estimated Operational Power	6.79 Watts
Total Energy Consumption	16.222583 Joules

C. Comparative Analysis

We compare our Hybrid-IDS model (AE + RF) with SVM, Naive Bayes and Standalone Random Forest and result shows our proposed model is efficient, accurate, reduce computational complexity, very low inference time and consume less power.

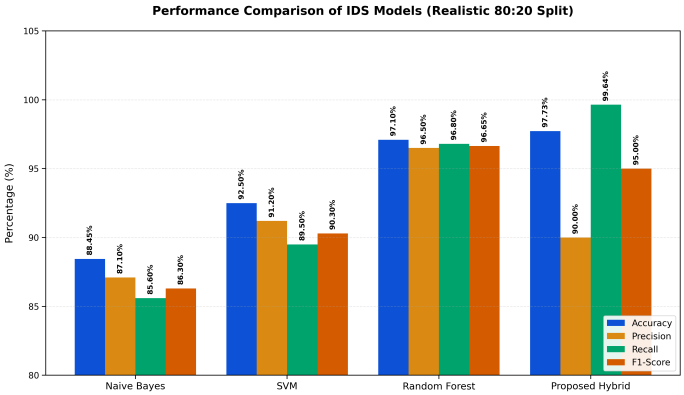

Fig. 5: Performance Comparison of IDS Models

VI. Future Work

Although the proposed energy-efficient hybrid intrusion detection system achieves high accuracy and low inference time, there are still several directions for further improvement. In future work, the model can be deployed and tested on real IoT hardware platforms such as Raspberry Pi or ESP8266 to evaluate its performance in practical environments.

Additionally, the integration of advanced deep learning techniques, such as lightweight neural networks or federated learning approaches, can further enhance detection accuracy while ensuring data privacy. The model can also be optimized for even lower energy consumption by applying advanced compression techniques and hardware-aware optimization methods.

Furthermore, testing the proposed system on larger and more diverse datasets will help improve its generalization capability and robustness against evolving cyber threats in real-world IoT networks.

VII. Conclusion

In this paper, an energy-efficient hybrid intrusion detection system (IDS) based on Autoencoder and Random Forest is proposed for securing Internet of Things (IoT) environments. The number of IoT devices is increasing but coomputational costs limited which make traditional security less effective. To overcome this challenge, the proposed model utilizes an Autoencoder for feature reduction and a Random Forest classifier for accurate threat detection.

According to experimental results, proposed hybrid model IDS achieves detection accuracy 97.73% and inference time is 0.1458 seconds which is very low for 20,000 records. Results shows that the model is capable of performing real-time intrusion detection while maintaining computational efficiency. Furthermore, the reduction in feature dimensionality significantly improves processing speed and reduces energy consumption, making the system suitable for resource-constrained IoT devices.

Compared to traditional machine learning models such as Support Vector Machine(SVM) and standalone Random Forest, our hybrid model provides better performance in terms of accuracy, efficiency, and scalability. Therefore, the proposed model offers a reliable and efficient solution for securing IoT networks against cyber threats.

References

[1] P. Gokhale, O. Bhat, and S. Bhat, "Introduction to iot," *International Advanced Research Journal in Science, Engineering and Technology*, vol. 5, no. 1, pp. 41–44, 2018.

[2] J. A. Faysal, S. T. Mostafa, J. S. Tamanna, K. M. Mumenin, M. M. Arifin, M. A. Awal, A. Shome, and S. S. Mostafa, "Xgb-rf: A hybrid machine learning approach for iot intrusion detection," in *Telecom*, vol. 3, no. 1. MDPI, 2022, pp. 52–69.

[3] A. Wani, N. K. Basha, M. Mohammed, I. Hussain, C. Ananth, and H. M. Rai, "Ai-driven botnet detection in iot networks: A comprehensive research review," *Computer Science Review*, vol. 61, p. 100941, 2026.

[4] K. Kumar, M. M. Shukla, B. Pandey, and W. Bakar, "Power and area efficient hardware architecture of lightweight cryptography for iot and embedded systems," *International Journal of Information Technology*, vol. 17, no. 6, pp. 3739–3749, 2025.

[5] H. Damghani, L. Damghani, H. Hosseinian, and R. Sharifi, "Classification of attacks on iot," in *4th international conference on combinatorics, cryptography, computer science and computation*, 2019, pp. 245–255.

[6] H. M. R. U. Rehman, S. Liaquat, M. J. Gul, M. Z. Jhandir, D. Gavilanes, M. M. Vergara, and I. Ashraf, "A systematic literature study of machine learning techniques based intrusion detection: datasets, models, challenges, and future directions," *Journal of Big Data*, vol. 12, no. 1, p. 264, 2025.

[7] M. M. Rahman, S. Al Shakil, and M. R. Mustakim, "A survey on intrusion detection system in iot networks," *Cyber Security and Applications*, vol. 3, p. 100082, 2025.

[8] H. El-Sofany, S. A. El-Seoud, O. H. Karam, and B. Bouallegue, "Using machine learning algorithms to enhance iot system security," *Scientific Reports*, vol. 14, no. 1, p. 12077, 2024.

[9] M. Sajid, K. R. Malik, A. Almogren, T. S. Malik, A. H. Khan, J. Tanveer, and A. U. Rehman, "Enhancing intrusion detection: a hybrid machine and deep learning approach," *Journal of Cloud Computing*, vol. 13, no. 1, p. 123, 2024.

[10] S. K. Jangam and P. S. R. P. Muntala, "Role of artificial intelligence and machine learning in iot device security," *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, vol. 3, no. 1, pp. 77–86, 2022.

[11] K. Shafi, H. Abbass, and W. Zhu, "An adaptive rule-based intrusion detection architecture," in *Proceedings of the 2006 RNSA security technology conference. Canberra, Australia*, 2006, pp. 307–319.

[12] A. Ojugo, A. Eboka, O. Okonta, R. Yoro, and F. Aghware, "Genetic algorithm rule-based intrusion detection system (gaids)," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 3, no. 8, pp. 1182–1194, 2012.

[13] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE access*, vol. 7, pp. 41 525–41 550, 2019.

[14] N. Thapa, Z. Liu, D. B. Kc, B. Gokaraju, and K. Roy, "Comparison of machine learning and deep learning models for network intrusion detection systems," *Future Internet*, vol. 12, no. 10, p. 167, 2020.

[15] S. García, S. Ramírez-Gallego, J. Luengo, J. M. Benítez, and F. Herrera, "Big data preprocessing: methods and prospects," *Big data analytics*, vol. 1, no. 1, p. 9, 2016.

[16] Y. Wang, H. Yao, and S. Zhao, "Auto-encoder based dimensionality reduction," *Neurocomputing*, vol. 184, pp. 232–242, 2016.

[17] A. Tawakuli, B. Havers, V. Gulisano, D. Kaiser, and T. Engel, "Survey: Time-series data preprocessing: A survey and an empirical analysis," *Journal of Engineering Research*, vol. 13, no. 2, pp. 674–711, 2025.

[18] S. Jamshidi, F. Erfan, O. Abdul-Wahab, M. Bellaiche, and F. Khomh, "Lightweight autoencoder-isolation forest anomaly detection for green iot edge gateways," *arXiv preprint arXiv:2511.18235*, 2025.

[19] E. S. Alomari, S. Manickam, and M. Anbar, "Adaptive hybrid deep learning model for real-time anomaly detection in iot networks," *Journal of Advanced Research Design*, vol. 137, no. 1, pp. 278–289, 2026.

[20] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[21] N. Moustafa, "A new distributed architecture for evaluating ai-based security systems at the edge: Network ton_iot datasets," *Sustainable Cities and Society*, vol. 72, p. 102994, 2021.

[22] N. Moustafa, M. Keshky, E. Debiez, and H. Janicke, "Federated ton_iot windows datasets for evaluating ai-based security applications," in *2020 IEEE 19th international conference on trust, security and privacy in computing and communications (TrustCom)*. IEEE, 2020, pp. 848–855.

[23] N. Moustafa, "New generations of internet of things datasets for cybersecurity applications based machine learning: Ton_iot datasets," in *Proceedings of the eResearch Australasia Conference, Brisbane, Australia*, 2019, pp. 21–25.

[24] A. Nurhopipah and U. Hasanah, "Dataset splitting techniques comparison for face classification on cctv images," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 14, no. 4, p. 341, 2020.

[25] K. M. Kahloot and P. Ekler, "Algorithmic splitting: A method for dataset preparation," *IEEE access*, vol. 9, pp. 125 229–125 237, 2021.

[26] V. R. Joseph and A. Vakayil, "Split: An optimal method for data splitting," *Technometrics*, vol. 64, no. 2, pp. 166–176, 2022.

[27] I. J. Ratul, Y. Zhou, and K. Yang, "Accelerating deep learning inference: A comparative analysis of modern acceleration frameworks," *Electronics*, vol. 14, no. 15, p. 2977, 2025.