IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 28 February 2025, accepted 19 March 2025, date of publication 24 March 2025, date of current version 2 May 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3554083

RESEARCH ARTICLE

Intrusion Detection in IoT and IIoT: Comparing Lightweight Machine Learning Techniques Using TON_IoT, WUSTL-IIOT-2021, and EdgeIIoTset Datasets

SHEREEN ISMAIL[1,2], (Senior Member, IEEE), SALAH DANDAN[3], AND ALA'A QUSHOU[4]
[1]Merit Network Inc., Ann Arbor, MI 48108, USA
[2]University of Michigan, Ann Arbor, MI 48108, USA
[3]School of Electrical Engineering and Computer Science, University of North Dakota, Grand Forks, ND 58202, USA
[4]Department of Computer Engineering, University of Jordan, Amman 11942, Jordan

Corresponding author: Shereen Ismail (drismail@umich.edu)

This work was supported in part by Merit Network Inc.

ABSTRACT The security of Internet of Things (IoT) and Industrial Internet of Things (IIoT) systems has been significantly enhanced through the integration of effective intrusion detection systems (IDSs). Machine learning (ML) has emerged as a highly efficient approach for designing cyber-attack detection systems to improve the security. This study reviewed recent advancements in the literature utilizing the TON_IoT, WUSTL-IIoT-2021, and Edge-IIoTset datasets. A comprehensive performance analysis of various supervised ML classification techniques was conducted to identify lightweight models suitable for deployment in resource-constrained IoT and IIoT environments. The performance of Decision Tree (DT), Random Forest (RF), and three ensemble techniques: Bagging, Stacking, and LightGBM (LGBM), was evaluated. The TON_IoT, WUSTL-IIOT-2021, and Edge-IIoTset imbalanced datasets, representing three distinct IIoT environments and encompassing numerous samples of different attack types, were used. The impact of imbalanced class distributions on model performance was analyzed. The imbalanced datasets were customized for training and testing ML models, with feature selection performed using Mutual Information (MI). Model performance was assessed using several metrics: Precision, Recall, Micro-F1, Model Size, and Training Time. Furthermore, a cross-dataset transfer learning approach was applied to evaluate how models trained on the TON_IoT dataset generalize when tested on the WUSTL-IIoT-2021 dataset, demonstrating the ability of the models to generalize across datasets with common features and attack labels. For real-time intrusion detection and network traffic analysis, we set up an experiment to deploy the trained ML models in a live network environment. The experiment provided real-time insights into CPU usage, memory consumption, and network activity, with predictions continuously logged for monitoring and further analysis.

INDEX TERMS Internet of Things, Industrial Internet of Things, security, cyber-attacks, intrusion detection, machine learning, lightweight, data balancing, cross-dataset transfer learning.

I. INTRODUCTION

The rapid expansion of the Internet of Things (IoT) and Industrial Internet of Things (IIoT) has transformed various sectors by enabling seamless connectivity and smart automation. However, this growth has also introduced significant security challenges. By 2025, it is estimated that there will be over 55.7×10^9 connected IoT devices globally, highlighting the sheer scale of the ecosystem according to IDC [1]. A report by Nokia showed that IoT devices were responsible for 33%

The associate editor coordinating the review of this manuscript and approving it for publication was Stefano Scanzio.

of all infected devices in 2020, underlining the vulnerability of these systems [2].

Industry 4.0, characterized by the intersection of Information Technology (IT) and Operational Technology (OT), plays a key role in shaping the IIoT integrated with Cyber-Physical Systems (CPS). Such systems form the backbone of smart factories, energy grids, intelligent transportation systems, and healthcare facilities in smart cities. However, such Industry 4.0 technologies also exposes critical infrastructure to cyber threats. Cybersecurity Ventures predicts that cybercrime will cost the world $10.5 trillion annually by 2025, with a significant portion attributable to attacks on IoT and IIoT systems. The average cost of a data breach in industrial environments is estimated at $4.24 million per incident, as reported by IBM, demonstrating the high stakes involved in securing these systems [3].

Figure 1 illustrates the layered architecture of IIoT systems, highlighting five distinct layers: Edge Layer, Middleware Layer, Application Layer, IT and OT Environment Layer, and Cloud Services Layer. Each layer has specific functions and associated security threats. The Edge Layer focuses on data collection and real-time interactions, facing threats such as device tampering and physical attacks. The Middleware Layer manages data flow and interactions, vulnerable to data leakage and unauthorized Application Programming Interface (API) access. The Application Layer includes business-specific applications and is prone to malware attacks and unauthorized configuration changes. The IT and OT Environment Layer integrates traditional IT and OT systems, with insider threats and legacy system vulnerabilities being primary concerns. The Cloud Services Layer provides scalability and advanced analytics but is at risk of data breaches and advanced persistent threats.

Several high-profile attack incidents documented in the literature underscore the importance of robust IoT and IIoT security [4], [5]. For example, the Mirai botnet attack in 2016, which infected IoT devices to perform a massive Distributed Denial of Service (DDoS) attack, took down major websites like Twitter, Netflix, and Reddit. The Stuxnet worm in 2010 targeted SCADA systems in industrial environments, damaging Iran's nuclear centrifuges and highlighting vulnerabilities in IIoT systems. The Triton/Trisis malware in 2017 targeted industrial safety systems, aiming to disrupt and potentially cause physical damage to industrial processes. These incidents illustrate the potential for significant disruption and damage from cyber-attacks on IoT and IIoT systems.

Machine learning (ML)-based Intrusion Detection System (IDS) have emerged as a powerful tool for enhancing security. Traditional IDS systems rely on signature-based detection, which can only identify known threats. While ML-based IDS can detect novel and unknown threats by analyzing patterns and anomalies in data. This capability is crucial for IoT and IIoT environments, where the threat landscape is constantly evolving. ML models can process large volumes of data in real-time and adapt to new attack vectors as they learn from new data. This real-time analysis and adaptation are essential for maintaining security in dynamic environments.

ML-based IDS significantly reduce false positives by distinguishing between normal and malicious activity with high precision [6], allowing security teams to focus on genuine threats. These systems offer scalability, as they can leverage cloud computing resources and distributed data processing frameworks to accommodate the growing number of IoT and IIoT devices.

Several comprehensive datasets emulating or simulating attacks in IoT/IIoT environments are available in the literature to develop and validate ML models for IoT/IIoT security, including UNSW-NB15 [7], CICIDS, DS2OS, N_BaIoT 2018, Bot-IoT [8], X-IIoTID [9], and LATAM-DDOS-IOT [10]. We selected TON_IoT, WUSTL-IIOT-2021, and Edge-IIoTset which are relatively new, imbalanced, and have been collected through emulating real-world industrial systems in diverse scenarios. However, developing ML-based IDS for datasets like TON_IoT, WUSTL-IIOT-2021, and Edge-IIoTset poses significant challenges due to several factors inherent to IIoT environments. These datasets form a fundamental component for training ML classifiers aimed at securing IIoT systems against cyber-attacks, yet they typically contain smaller amount of abnormal data compared to normal instances. In IIoT settings, normal operations vastly outnumber abnormal events, making the detection of anomalies more complex and prone to false positives.

ML algorithms rely heavily on the quality and quantity of data for effective training. In IIoT environments, collecting real-world data can be challenging due to the diverse and dynamic nature of industrial operations. Also, anomalies in IIoT networks can be subtle and context-dependent, requiring sophisticated feature extraction techniques to distinguish them from normal behavior. Adding to the complexity is the requirement to deploy IDS on resource-constrained IIoT devices, which limits the allowable computational complexity and energy consumption for ML models.

TON_IoT, WUSTL-IIOT-2021, and Edge-IIoTset datasets come with their own variations. For instance, TON_IoT and WUSTL-IIOT-2021 datasets provide insights into traffic patterns and anomalies in IIoT networks, while EdgeIIoTset focuses on edge computing environments. A wise preprocessing for meaningful comparison between the three datasets, along with consideration of the imbalancing issue, is typically required to analyze the characteristics of IIoT data, such as sensor readings or network traffic patterns, and extract meaningful features for ML classifiers

The main contributions of this study are as follows:

- **Comprehensive Literature Review:** This paper provides a thorough literature review of prior works utilizing the TON_IoT, WUSTL-IIoT-2021, and Edge-IIoTset datasets. The review highlights the contributions of existing studies while identifying the lack of unified evaluations across these three datasets, particularly in addressing their unique characteristics and challenges

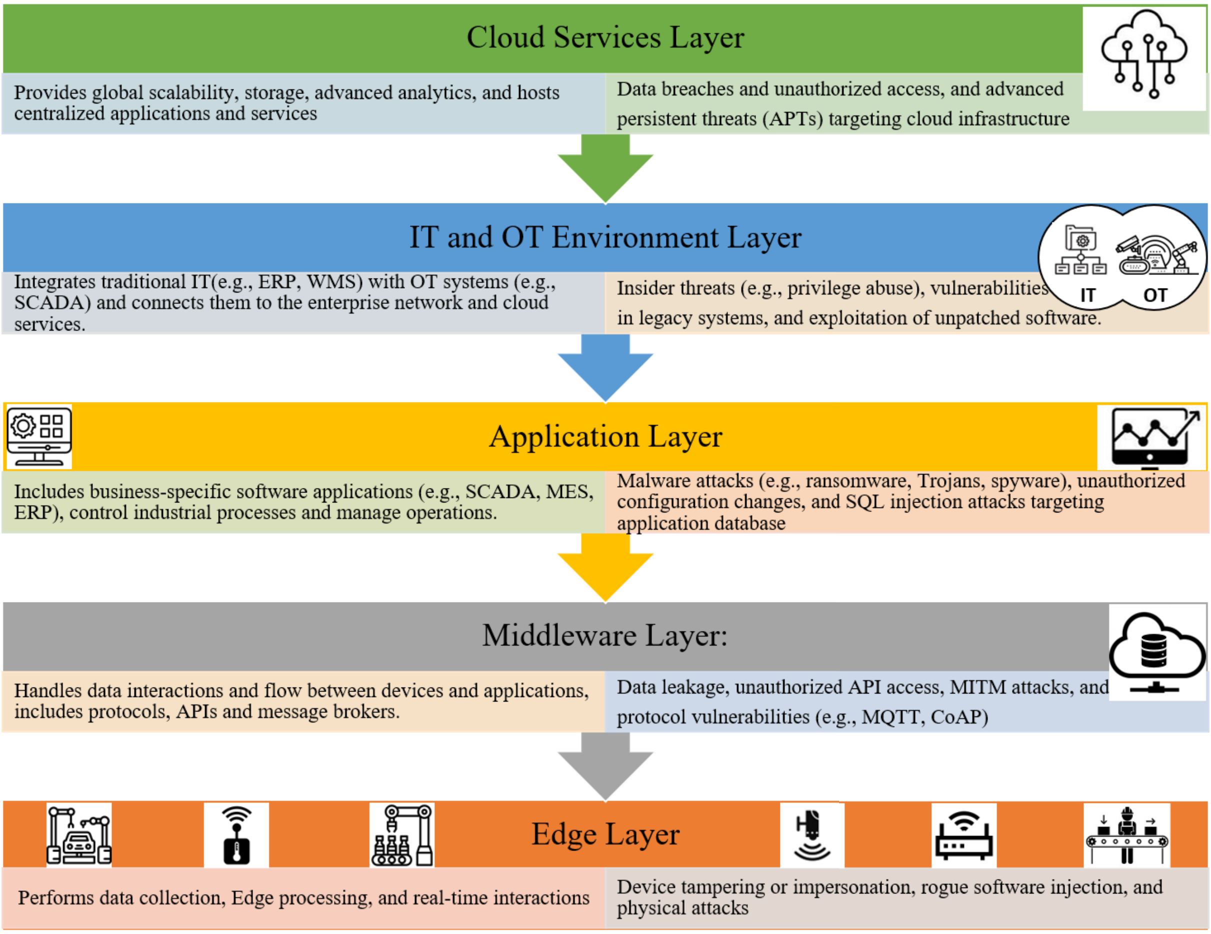

Cloud Services Layer
Provides global scalability, storage, advanced analytics, and hosts centralized applications and services
Data breaches and unauthorized access, and advanced persistent threats (APTs) targeting cloud infrastructure
IT and OT Environment Layer
Integrates traditional IT(e.g., ERP, WMS) with OT systems (e.g., SCADA) and connects them to the enterprise network and cloud services.
Insider threats (e.g., privilege abuse), vulnerabilities in legacy systems, and exploitation of unpatched software.
IT
OT
Application Layer
Includes business-specific software applications (e.g., SCADA, MES, ERP), control industrial processes and manage operations.
Malware attacks (e.g., ransomware, Trojans, spyware), unauthorized configuration changes, and SQL injection attacks targeting application database
Middleware Layer:
Handles data interactions and flow between devices and applications, includes protocols, APIs and message brokers.
Data leakage, unauthorized API access, MITM attacks, and protocol vulnerabilities (e.g., MQTT, CoAP)
Edge Layer
Performs data collection, Edge processing, and real-time interactions
Device tampering or impersonation, rogue software injection, and physical attacks

FIGURE 1. The IIoT layered architecture.

associated with imbalanced datasets and cross-dataset generalization.

- **Comparative Study and Performance Analysis of ML Techniques:** This paper conducts a comparative study and performance analysis of various ML supervised techniques across multiclass classification: Decision Trees (DT), Random Forest (RF) vs. ensemble algorithms: Bagging, Stacking, and Light GBM (LGBM) to detect cyber-attacks targeting IoT and IIoT using TON_IoT, WUSTL-IIOT-2021, and Edge-IIoTset datasets. The aim is to analyze the impact of imbalanced class distributions on model performance. We identify lightweight ML models that strike a balance between efficiency and resource usage, making them well-suited for deployment in resource-constrained IIoT. We apply lightweight methodology in terms of pre-processing and feature extraction to ensure minimal computational overhead. These datasets are characterized by insufficient data samples for certain attack classes and a large imbalance with a prevalence of normal samples. Thus, we customize the datasets to ensure improved class representation for training and testing ML models. The evaluation includes metrics such as Precision, Recall, Micro-F1, in addition to Model Size (MS) and Training Time (TT).
- **Cross-Dataset Transfer Learning Experiment:** A cross-dataset transfer learning approach is employed to analyze the robustness and generalization capability of the ML models. Models trained on the TON_IoT dataset are evaluated on the WUSTL-IIoT-2021 dataset, leveraging common features and attack labels across the datasets. The study highlights the impact of domain shift and dataset variations on model performance and underscores the importance of preprocessing techniques to ensure reliable generalization across different IIoT environments.
- **Real-Time Network Traffic Monitoring and Intrusion Prediction:** We sets up an experiment to continuously monitor network traffic and predict intrusions using the trained ML models. Live packet data is captured from the network interface, processed into flow records, and key features are extracted to classify traffic as legitimate or malicious. The system operates in real-time, providing continuous insights into CPU

usage, memory consumption, and network activity, with predictions logged for monitoring and further analysis.

The remainder of this paper is organized as follows: Section II provides a comprehensive review of the literature, focusing on recent studies utilizing each of the three datasets. Section III details the proposed lightweight methodology applied to each of the three datasets. Section IV presents the simulation results along with an in-depth discussion. Section V illustrates the cross-dataset transfer learning experiment. Section VI describes the experiment of real-time network traffic monitoring and intrusion prediction. Finally, Section VII concludes the paper and outlines directions for future work.

II. RELATED WORK

Significant research efforts are apparent in securing IoT, with a focus on industrial sectors such as manufacturing, logistics and supply chain management. This emphasis is highlighted by multiple surveys, including [11], [12], [13], [14], [15], exploring potential techniques and solutions to improve the security of IIoT applications in these specific industrial environments by employing different datasets to propose effective ML-based IDS.

The TON_IoT, WUSTL-IIOT-2021, and Edge-IIoTset datasets have been widely adopted for training and evaluating IDS models due to their representation of real-world network traffic and attack patterns. However, these datasets often exhibit significant class imbalances, posing challenges in building robust and effective IDSs. To mitigate these challenges, preprocessing steps such as data balancing techniques, feature normalization, and dimensionality reduction are commonly employed. Recent studies have explored various ML techniques, Deep learning (DL) approaches, and feature selection algorithms, to enhance detection performance while addressing class imbalance and maintaining high detection accuracy (ACC). This section provides a summary of the most recent and relevant work in the literature, highlighting the methodologies, results, and challenges encountered in the development of IDS for IoT and IIoT environments using these datasets.

A. TON_IoT

As shown in table 1, recent work using the TON_IoT dataset highlights significant findings. Gad et al. [16] proposed a distributed detection system utilizing ML to identify and mitigate IoT attacks. They employed the TON_IoT dataset, encompassing data from cloud, fog, and edge layers, for training and testing. Four ML techniques were used: K-Nearest Neighbors (KNN), XGBoost, DT, and RF. Feature selection via Chi2 reduced features to 20, and the SMOTE technique addressed class imbalance. XGBoost achieved the highest ACC, Recall, Precision, and F1-score: 0.983, 0.984, 0.967, and 0.97, respectively. In a related study, Gad et al. [17] used multiple ML techniques: Logistic Regression (LR), Gaussian Naive Bayes (NB), DT, RF, AdaBoost, KNN, Support Vector Machine (SVM), and XGBoost, on the TON_IoT dataset for binary and multi-class classification. Chi2 was used for feature selection, and Min-Max normalization standardized the dataset. XGBoost achieved superior performance, with Precision, Recall, F1-score, and false positive rate of 0.991, 0.984, 0.987, and 0.009, respectively.

Guo et al. [18] developed an IDS framework for IoT systems, testing 10 ML algorithms, including AdaBoost, CatBoost, Extra Trees, XGBoost, and Stacking ensembles, on the TON_IoT dataset. The Stacking-ensemble model achieved the highest Matthews Correlation Coefficient (MCC) of 0.9971 and 0.9909 for binary and multi-class classification. Belarbi et al. [19] used Federated Learning (FL) with Deep Belief Networks (DBN) and Deep Neural Network (DNN) on the TON_IoT dataset. They preprocessed the data by removing `NaN` values, duplicates, and irrelevant features and converted categorical features into numeric form. DBN outperformed DNN, achieving Precision, Recall, and F1-score of 91, 74, and 78, respectively.

Oseni et al. [20] enhanced DL-based IDS transparency using Shapley additive explanation (SHAP) for the TON_IoT dataset. Their framework achieved 99.15% ACC and 98.83% F1-score, showcasing its capability in protecting IoT networks. Latif et al. [21] proposed a Dense Random Neural Network (DNRANN) for IoT intrusion detection, tested on the TON_IoT dataset. The model achieved ACCs of 99.14% and 99.05% for binary and multi-class classification, respectively.

Malik et al. [22] integrated blockchain and DL to secure smart city environments, achieving a 99.8% detection rate for backdoor attacks using TON_IoT and BoT-IoT datasets. Their approach combined a Feistel structure and Deep Reinforcement Learning (DRL) for enhanced privacy and accountability. While Shtayat et al. [24] developed an explainable ensemble DL-based IDS using SHAP and Local Interpretable Model-Agnostic Explanations (LIME). Their approach achieved 99.96% ACC on the TON_IoT dataset.

Musleh et al. [23] employed VGG-16 and DenseNet for feature extraction, combined with SMOTE for imbalance correction. Stacked models achieved the highest ACC of 98.3%.

Wang et al. [25] introduced a Transformer-based Network IDS leveraging self-attention for feature embeddings, achieving 98.39% ACC for binary classification when incorporating IoT telemetry data.

Li et al. [26] proposed an anomaly-based IDS combining convolutional and recurrent layers with SVM for precise classification. Their approach demonstrated significant improvements on TON_IoT, attaining high detection ACC for various attack types.

In [27], the authors proposed a self-attention-based deep convolutional neural network (SA-DCNN) for monitoring IIoT networks and detecting malicious activities. The SA mechanism assigned significance values to each input feature, and the DCNN leverages these values for classification.

TABLE 1. Summary of recent work using the TON_IoT dataset.

Ref.	Year	Contribution	Balancing technique	Feature Selection	ML Techniques	Performance Metrics
[16]	2022	Proposed distributed ML-based IDS for IoT.	SMOTE	Chi2	XGB, DT, KNN, RF	ACC: 0.983, Recall: 0.984, Precision: 0.967, F1-score: 0.97
[17]	2021	Proposed ML-based IDS for VANETs.	SMOTE	Chi2	XGB, LR, NB, SVM, DT, AdaBoost, KNN	ACC: 0.991, Recall: 0.991, Precision: 0.984, F1-score: 0.987
[18]	2023	Developed ML-based IDS framework for IoT systems to monitor abnormal network activities.	Imbalanced	Chi2, Spearman rank correlation coefficient	AB, CB, ET, DT, XGB, RF, GB, KNN, EnS, EnV	MCC: NA, Other metrics: NA
[19]	2023	FL-based IDS for IoT.	Imbalanced	NA	DNN, DBN	Precision: 91, Recall: 74, F1-score: 78
[20]	2023	Explainable DL-based IDS for IoT.	NA	NA	CNN	ACC: 99.15, F1-score: 98.83
[21]	2022	Lightweight DNRANN for IoT-IDS.	NA	NA	DnRaNN	ACC: 99.14
[22]	2023	Developed a secure platform for digital governance, interoperability, and data exchange.	NA	NA	DRL, NB, DT, TP2SF	ACC: Over 95%, Recall: Over 97%, Precision: Over 98%, F1-score: Over 97%
[23]	2023	Used feature extraction with ML algorithms for IoT intrusion detection.	SMOTE	VGG-16 and DenseNet	RF, KNN, SVM, stacked models	ACC: 98.3%, Other metrics: NA
[24]	2023	Explainable ensemble DL-based IDS.	NA	NA	CNNs, ELM, Ensemble	ACC: 99.96, Precision/Recall/F1-score: 100
[25]	2023	Proposed Transformer-based IoT Network IDS.	NA	NA	FT-Transformer	ACC: 97.95% (binary classification), 95.78% (multi-class classification)
[26]	2023	Proposed CRSF framework for IDS in IIoT.	NA	Automatic feature extraction eliminates manual selection.	CNN2D-RNN, SVM	ACC: 0.9959, F1-score: 0.9959
[27]	2024	Designed a novel DL model to detect attacks in IIoT networks.	Imbalanced	Feature filtering using Mutual Information (MI)	SA-DCNN	ACC: 99.95, Recall: 99.61, Precision: 99.46, F1-score: 99.53
[28]	2022	Used Inception-Time and DenseNet with sliding window approach to improve ACC and memory usage.	Class Weights	Chi-squared	DenseNet	ACC: 94.94, Recall: 92.4, Precision: 98.3, F1-score: 95.3
[29]	2024	Proposed a hybrid IDPS for combating 6LoWPAN attacks.	NA	NA	Hybrid IDPS	ACC: 99.94%
[30]	2023	FL-IDS for wireless IIoT networks	Imbalanced	Recursive Feature Elimination	FL	ACC: 93.92%

A two-step cleaning procedure removed data duplication, considering both intra-class and cross-class samples. The model used MI-based feature filtering to tackle underfitting. Performance evaluation on the IoTID20 and Edge-IIoTset datasets showed 96.89% ACC for IoTID20 and 99.95% ACC for Edge-IIoTset, outperforming classic ML and DL models.

In [28], the authors assessed the DenseNet and Inception Time models for multi-class classification of cyber-attacks using TON_IoT, Edge-IIoT, and UNSW 2015 datasets. DenseNet achieved 99.9% ACC for Windows 10, while Inception Time reached 100% ACC on network data. On Edge-IIoT, Inception Time achieved 94.94% ACC, and UNSW-NB15 showed 98.4% ACC, improving to 98.6% with the sliding window technique. The study highlighted Inception Time's relevance for cyber-attack detection.

In [29], the authors presented a hybrid Intrusion Detection and Prevention System (IDPS) for detecting 6LoWPAN attacks in IoT environments. The system simulated three types of 6LoWPAN attacks using Cooja, creating a dataset with six scenarios—three with and three without malicious nodes. The IDPS was tested on various ML classifiers, with the ANN achieving the highest ACC. DT-based models were lighter and suitable for edge or fog computing, handling large IoT network data efficiently.

In [30], the authors introduced an IoT IDS-based on FL, enhancing security and privacy. The system trains local IoT device data and shares parameter updates with a central server, which aggregates and redistributes the model. Evaluation on the Edge-IIoTset dataset showed an ACC of 92.49%, comparable to centralized ML models, proving the efficacy of the FL-based method.

B. WUSTL-IIOT-2021 DATASET

As shown in table 2, recent work using the WUSTL-IIOT-2021 dataset highlights significant findings. Eid et al. [31] presented an IDS specifically designed for IIoT scenarios with an optimized Convolutional Neural Network (CNN) model. The model was trained using a dataset that was equilibrated through an innovative multi-class application of the SMOTE, guaranteeing uniform representation of all classes. Furthermore, systematic optimization was employed to refine the hyperparameters of the CNN model and alleviate the impact of the expanded training dataset size.

TABLE 2. Summary of recent work using WUSTL-IIOT-2021 dataset.

Ref.	Year	Contribution	Balancing technique	Feature Selection	ML Techniques	Performance Metrics
[31]	2024	Presented an optimized CNN model for IDS	SMOTE	NA	CNN	ACC: 99.99, Precision: 99.94, Recall: 99.65, F1: 99.8
[32]	2023	Developed an efficient IDS model for IIoT systems.	NA	NA	ANN, SVM, DT, RF, LR, NB	ACC: 99.97
[33]	2023	Employed ML techniques on the WUSTL-IIOT dataset.	NA	NA	LDA, SVM, SVM+LDA, QDA	ACC: 100, Precision: 92.54, Recall: 98.48, F1: 94.84
[34]	2024	Suggested an innovative IDS model using RF technique for classification of hostile activities in IIoT-based networks	NA	PSO, BA	RF	ACC: 0.999, Precision: 0.996, Recall: 0.996, F1: 0.996
[35]	2024	Introduced a blockchain-enabled information security framework incorporating a ML security model within a multi-layered strategy.	RandomUnderSampler	MI, ET	NB, KNN, RF, DT, Bagging, Stacking, Boosting	ACC: 100, Precision: Over 95, Recall: Over 95, F1: 100
[36]	2024	Examined the efficacy of six ML models for IDS system in industrial Internet.	SMOTE, RandomUnderSampler, RandomOverSampler	NA	RF, DT, KNN, LR, SVM, NB	ACC: 99.99, Precision: 99.98, Recall: 99.99, F1: 99.99
[37]	2023	Introduced an innovative intrusion detection method for the identification of unusual network traffic.	SMOTE	Recursive Feature Elimination	XGBoost	ACC: 100, Precision: 100, Recall: 100, F1: 100
[38]	2024	Introduced a DRL framework for anomaly detection within the SCADA network.	NA	NA	Q-network	ACC: 99.36
[39]	2024	Developed a high-ACC IDS with minimal payload.	NA	PSO	MARS, GAM	ACC: 100
[40]	2023	Developed a network-IDS model for IIoT security.	NA	PCC	IF	ACC: 92.17%, MCC, AUC

The evaluation results indicated significant performance enhancement when the improved CNN model was trained on a balanced version of WUSTL-IIOT-2021 dataset, followed by an evaluation of its generalization capacity with the non-domain specific UNSW_NB15 dataset. The model's efficacy was assessed by ACC, Precision, Recall, and F1-score metrics. The findings indicated that the suggested IDS was exceptionally successful, achieving performance surpassing 99.9% across all criteria. The IDS has significant efficacy in identifying intrusions within generic IT networks, with enhancements over 30% relative to the default baseline model.

Eid et al. [32] used six ML algorithms: Artificial Neural Networks (ANN), SVM, KNN, NB, LR, and DT to propose several IDS models. These models were tested on the WUSTL-IIOT-2021 dataset. They converted the categorical features to numeric features using one-hot encoding and applied the normalization method. The model built on the RF algorithm achieved an ACC of 99.97%, surpassing previous models. This highlights the potential of ML-based IDS models in advancing the security and safety of IIoT systems, where all interactions are harmful and nothing is controlled. Tareq et al. [33] utilized a parallel approach using ML techniques, including Linear Discriminant Analysis (LDA), SVM, SVM combined with LDA, and Quadratic Discriminant Analysis (QDA), on the WUSTL-IIOT dataset, and compared these methods with traditional methodologies. Prior to classification, the data underwent pre-processing. Features such as [’StartTime,’ ‘LastTime,’ ’SrcAddr,’ ‘DstAddr,’ ’sIpId,’ ‘dIpId’] were removed, as mentioned by the authors in their study [41]. ’Target’ was also eliminated to facilitate multi-classification. Features with a correlation exceeding 0.95 were excluded. The data was split into separate training datasets and trained concurrently. Experiments demonstrated that this parallel training system identifies and predicts cyber threats with greater Precision. The detection speed of the parallel ML models was elevated, achieving optimal ACC of 100% with the SVM+LDA model. Gaber et al. [34] presented a novel IDS model based on Particle Swarm Optimization (PSO) and Bat algorithm (BA) for feature selection, and a RF classifier for classifying malicious behaviors in IIoT-based network traffic. The model's performance was evaluated using the WUSTL-IIOT-2021 dataset based on four evaluation metrics: ACC, Precision, Recall, and F1-score. The results showed that the RF and BA classifiers proved to be the best, achieving 0.999, 0.9966, 0.996, and 0.996, respectively.

Ismail et al. [35] introduced a blockchain-enabled information security framework incorporating a ML security model within a multi-layered strategy. They examined the efficacy of Gaussian-NB, KNN, RF, DT, and three ensemble methods: Bagging, Stacking, and boosting. They used the WUSTL-IIOT-2021 unbalanced dataset. MI and Extra-trees (ET) were used as a single-stage ensemble feature selection method. The efficacy of the ML models was assessed by classification ACC, Precision, Recall, F1-score, MCC, MS, TT, and Prediction Time (PT). RF, Bagging, Stacking, and CatBoost demonstrated strong performance across most metrics, with the Stacking model achieving the highest ACC, Recall, micro F1-score, macro F1-score, and MCC, while CatBoost exhibited the greatest Precision.

Eid et al. [36] examined the efficacy of six notable ML models: RF, DT, KNN, LR, SVM, and NB, for IDSs in IIoT settings using the WUSTL-IIOT-2021 dataset. They applied data preprocessing methods such as feature engineering, data normalization, recording, and the handling of missing data. They also examined multi-class attack identification with a novel SMOTE-based multi-class balancing method to address dataset imbalances. The findings demonstrated that data preprocessing and intelligent dataset balancing consistently improved classification performance in both binary and multi-classification tasks. The RF algorithm stood out for its consistently strong performance and computational efficiency.

Xu et al. [37] introduced an innovative IDS that effectively identifies unusual network traffic. They utilized the binary grey wolf optimizer (BGWO) heuristic approach and recursive feature elimination (RFE) to reduce the feature dimensions and identify the most pertinent feature subset. The SMOTE method was employed to augment the minority class and alleviate the effects of data imbalance on classification outcomes. The preprocessed data were classified via XGBoost, with hyperparameters improved by Bayesian optimization with a tree-structured Parzen estimator (BO-TPE) to maximize detection performance. The findings indicated that their suggested strategy surpassed state-of-the-art techniques in four of five datasets. Their method attained flawless ACC, Precision, Recall, and an F1-score of 1.0 on the BoT-IoT and WUSTL-IIOT-2021 datasets, substantiating the efficacy of their methodology.

Mesadieu et al. [38] introduced a DRL framework for anomaly detection within SCADA networks. Their model employed a Q-network, enabling high performance in pattern recognition for complex tasks. They validated their solution using WUSTL-IIoT-2021 and its predecessor, WUSTL-IIoT-2018.. The results indicated that their model attained an ACC of 99.36% in attack detection, underscoring the potential of DRL to improve the security of critical IIoT infrastructure.

Tiwari et al. [39] developed a high-ACC IDS with minimal payload. The experimental process utilized WUSTL-IIOT-2021. The feature selection method employed was PSO, alongside feature reduction techniques including Principal Component Analysis (PCA), LDA, and t-distributed stochastic neighbor embedding (t-SNE). Furthermore, the Generalized Additive Model (GAM) and Multivariate Adaptive Regression Splines (MARS) were implemented to identify payloads that could disrupt normal application operations. The combination of PSO and PCA with MARS yielded predictive results with high ACC. The trained ML model was quantized to 4-bit and 8-bit formats and deployed on Azure IoT Edge to emulate edge devices. Experimental findings indicated a 25% reduction in model latency following quantization.

To mitigate the computational and temporal expenses associated with the dataset's high dimensionality, Mohy-eddine et al. [40] advocated the utilization of Pearson's correlation coefficient (PCC) and isolation forest (IF). The IF was utilized to eliminate outliers, while the PCC was employed for feature selection. Furthermore, they employed the MCC to examine the influence of their proposed model on the imbalanced datasets: Bot-IoT and WUSTL_IIOT_2021. The RF classifier improved the performance of the IDS. Their methodology demonstrated exceptional outcomes, achieving 99.99% and 99.12% ACC, 92.17% and 93.96% MCC, and 92.48% and 99.3% Area Under the Curve (AUC) scores on the Bot-IoT and WUSTL_IIOT_2021 datasets, respectively.

C. EDGE-IIoTSET DATASET

As summarized in table 3, recent studies using the Edge-IIoTset dataset yielded significant findings. Ferrag et al. [42] introduced Edge-IIoTset, a cybersecurity dataset for IoT and IIoT. This dataset incorporates features extracted from logs, warnings, and traffic, categorizing 14 attacks into five types. Among various ML models evaluated, DNN demonstrated the best performance across ACC, Precision, Recall, and F1-score.

Ahmed et al. [43] introduced EdgeGuard, a ML-based framework for proactive intrusion detection in edge networks. Utilizing CNNs in a residual configuration, EdgeGuard efficiently captured patterns in network traffic. Evaluated on the Edge-IIoTset dataset, it achieved a micro-average ROAUC of 0.96 and macro-average of 0.88, demonstrating its capability to enhance IoT security.

Bhandari et al. [44] applied multiple ML models: DT, RF, NB, SVM, Gradient Boosting, and DNN, to Edge-IIoTset and IoT-23 datasets. They balanced classes via undersampling and oversampling, achieving optimal results with SVM and DT based on accuracy, Precision, Recall, and F1-score.

Benamor et al. [45] evaluated RF, DT, and XGBoost with feature selection techniques including, Chi2 and MI. XGBoost outperformed others on Edge-IIoTset and BoTNeTIoT datasets, highlighting its efficacy in intrusion detection.

Popoola et al. [46] explored Federated Deep Learning (FDL) on Edge-IIoTset, demonstrating ACC (99.60%), Precision (92.50%), Recall (95.42%), and F1-score (93.51%) while reducing TT by up to 75.87% compared to centralized models.

Ramaiah et al. [47] tested models like Extremely Randomized Trees (ERT), XGBoost, LSTM, and AdaBoost on Edge-IIoTset. Using SMOTE for balancing and Z-score normalization, ERT achieved 99.93% accuracy, outperforming other Network IDS solutions.

Khacha et al. [48] designed a hybrid CNN-LSTM model for intrusion detection. Preprocessing steps included filtering and normalization. The hybrid model achieved 100% accuracy in binary classification and 98.69% in multiclass classification on Edge-IIoTset.

Hamza et al. [49] used supervised ML methods like KNN, DT, LR, SVM, and RF for malware detection on Edge-IIoTset. RF achieved the highest accuracy (94%) among tested methods.

TABLE 3. Summary of recent work using edge-IIoTset dataset.

Ref.	Year	Contribution	Balancing Technique	Feature Selection	ML Techniques	Performance Metrics
[43]	2024	Introduced EdgeGuard based on ML designed for proactive intrusion detection on edge networks.	Imbalance	NA	CNN	ROCAUC
[44]	2022	Presented the framework of a cyberattack prediction using several ML methods.	Imbalance	NA	DNN, DT, RF, SVM, NB, Gradient Boosting	Accuracy: 0.95, Precision: 0.96, Recall: 0.92, 0.87, 0.97, 0.98, F1: 0.94, 0.93
[45]	2023	Assessed the effectiveness of supervised ML methods in identifying threats in the IoT network.	Imbalance	Chi2, MI	RF, DT, XGBoost	Accuracy: 100, Precision: 100, Recall: 100, F1: 100
[42]	2022	Evaluated ML-based intrusion detection systems in centralized and FL modes.	SMOTE	Zeek and TShark tools	FL, Centralized learning	Accuracy: 100, Precision: 100, Recall: 100, F1: 100
[46]	2024	Proposed a robust security solution for designing and deploying a resilient CIoT using FDL framework.	NA	NA	Centralized DL	Accuracy: 99.60, Precision: 95.42, Recall: 92.50, F1: 93.51
[47]	2024	Demonstrated potential attack vectors to IIoT using EdgeIIoT-2021 with ML and DL.	SMOTE	PCA	ERT, XGB, LSTM, AdaBoost, RandNN	Accuracy: 99.93, Precision: 97.7, Recall: 95.1, F1: 93.5
[48]	2022	Proposed a hybrid DL-based model using CNN and LSTM.	NA	NA	CNN-LSTM	FPR, Accuracy: 100% for binary, 98.69 for multiclass
[49]	2023	Dataset Evaluation for malware detection.	NA	NA	KNN, DT, LR, SVM, RF	Accuracy: 94%
[50]	2023	Developed an innovative MEC architecture to enhance the security of IoT applications via FL.	NA	NA	DETECT	Accuracy: 99, Precision: 99, Recall: 99, F1: 99
[51]	2023	Used ensemble approaches for attack detection on IoT networks	NA	Remove non-numeric and zero values features	CatBoost, XGBoost, RF, DT	Accuracy: 99.53, Precision: 99.07, Recall: 99.22, F1: 99.14
[52]	2024	Proposed a DL-based Network-IDS	NA	NA	CNN, AE	Accuracy: 92.34, Precision: 90.28, Recall: 91.69, F1: 89.08
[53]	2023	Employed DL models to create an anomaly-based IDS	NA	NA	CNN, GRU, CNN + GRU	Accuracy: 98.7%, Precision: NA, FPR: 91.89%
[54]	2023	Proposed a malware threats detection system for varied and distributed contexts.	NA	NA	DNN, ANN, DT, RF, SVM	Accuracy: 96%, Precision: 85%, Recall: 76%, F1: 86%

Abou El Houda et al. [50] proposed DETECT, an edge-based FL framework for IoT security. Tested on Edge-IIoTset and NSL-KDD, DETECT achieved 99% accuracy and F1-score on Edge-IIoTset, securing IoT applications while preserving privacy.

Keserwani et al. [51] employed CatBoost and XGBoost for IoT attack detection on Edge-IIoTset. Feature preprocessing and selection reduced dimensionality to 20 key features. XGBoost achieved superior performance with 99.53% accuracy and related metrics.

Angelin et al. [52] presented a CNN-AutoEncoder hybrid for IIoT intrusion detection. The model improved detection efficiency by minimizing data features and achieved 92.34% accuracy, 91.69% Precision, 90.28% Recall, and 89.08% F1-score on Edge-IIoTset. Saadouni et al. [53] proposed an anomaly-based detection system using DL, combining CNN and GRU. They evaluated the model on binary and multiclass classification and the results showed the CNN-GRU model outperformed existing approaches in accuracy and Precision. It surpassed the standalone GRU model, achieving an 88% detection cost in multiclass classification for a single data flow.

In [54], the authors presented an Artificial Intelligence (AI)-based method for detecting malware in IoT devices in smart environments. The system monitored network traffic to identify malware and improve security. The DNN model deployed on IoT gateways showed minimal increases in network bandwidth, which was less than 30 kb/s, and CPU consumption of 2%. The model achieved 93% ACC and a 92% F1-score, demonstrating its effectiveness in detecting malware in smart environments.

III. METHODOLOGY

This section outlines the detailed methodology applied to each of the three datasets, encompassing data pre-processing, feature selection, data balancing and train-test splitting, and the evaluation of ML models' performance. The primary objective is to ensure consistent and meaningful analysis across all datasets under study.

A. DATASET PRE-PROCESSING AND FEATURE SELECTION

For each of the three datasets, we identified and removed redundant identity features as well as irrelevant attributes specific to each dataset to ensure optimal model performance. We also referenced the original papers that introduced these datasets to maintain consistency with their proposed methodologies and feature selections. A threshold of 0.1 was selected for MI-based feature selection to retain the most relevant features while significantly reducing computational complexity and mitigating overfitting. Preprocessing and

TABLE 4. Description of TON_IoT, WUSTL-IIOT-2021, and Edge-IIoTset original and model-ready datasets.

Dataset	Original Data				Model-Ready Data		
	Class Label	Sample Size	Percentage	Total Size	Sample Size	Percentage	Total Size
TON_IoT	Normal	50,000	23.7%	211,043	106,613	23.7%	449,996
	XSS	20,000	9.48%		42,645	9.48%	
	DDoS	20,000	9.48%		42,645	9.48%	
	DoS	20,000	9.48%		42,645	9.48%	
	Password Cracking	20,000	9.48%		42,645	9.48%	
	Reconnaissance	20,000	9.48%		42,645	9.48%	
	MITM	1,043	0.49%		2223	0.49%	
	Ransomware	20,000	9.48%		42,645	9.48%	
	Backdoor	20,000	9.48%		42,645	9.48%	
	Injection Attacks	20,000	9.48%		42,645	9.48%	
WUSTL-IIOT	Normal	1,107,448	92.7%	1,194,464	417,217	92.7%	449,997
	Command Injection	259	0.022%		97	0.022%	
	Reconnaissance	8240	0.69%		3,104	0.69%	
	DoS	78305	6.56%		29,500	6.56%	
	Backdoor	212	0.018%		79	0.018%	
Edge-IIoTset	Normal	24301	15.4%	157,800	69,299	15.4%	449,991
	DDoS-UDP	14498	9.19%		41,344	9.19%	
	DDoS-ICMP	14090	8.92%		40,180	8.92%	
	SQL-Injection	10311	6.53%		29,403	6.53%	
	DDoS-TCP	10247	6.49%		29,221	6.49%	
	Vulnerability Scanner	10076	6.38%		28,733	6.38%	
	Password	9989	6.33%		28,485	6.33%	
	DDoS-HTTP	10561	6.69%		30,116	6.69%	
	Uploading	10269	6.50%		29,284	6.50%	
	Backdoor	10195	6.46%		29,284	6.46%	
	Port-Scanning	10071	6.38%		28,719	6.38%	
	XSS	10052	6.37%		28,665	6.37%	
	Ransomware	10925	6.92%		31,154	6.92%	
	Fingerprinting	1001	0.63%		2,854	0.63%	
	MITM	1214	0.77%		3,461	0.77%	

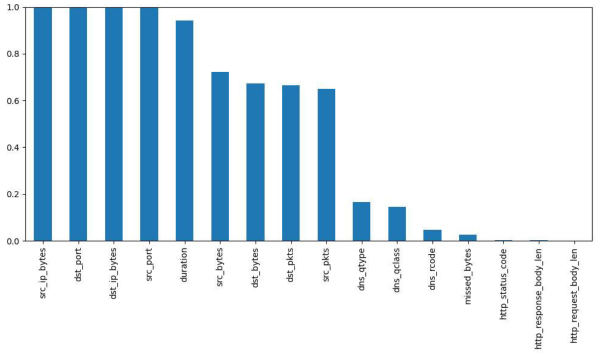


FIGURE 2. MI feature selection results for TON_IoT.

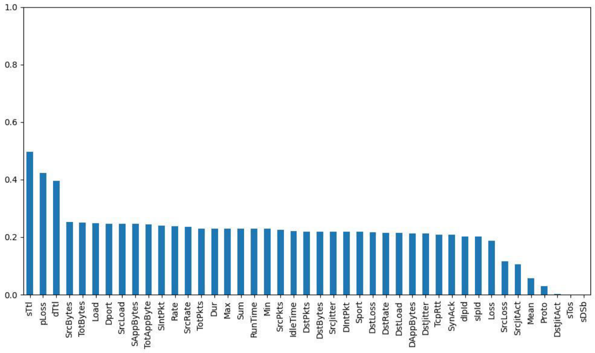


FIGURE 3. MI feature selection results for WUSTL-IIOT-2021.

feature selections steps leading to a reduction in feature dimensionality by 72%, 20%, and 77% for the TON_IoT, WUSTL-IIoT-2021, and Edge-IIoTset datasets, respectively.

1) TON_IoT

The data samples were collected from a large-scale testbed network comprising virtual machines, physical systems, hacking platforms, cloud and fog platforms, and IoT sensors. This setup was designed to simulate the complexity and scalability of IIoT and Industry 4.0 networks [55]. The dataset includes a variety of attacks, such as Denial of Service (DoS)/DDoS, reconnaissance, ransomware, backdoor, data injection, command injection, password cracking, and Man-in-the-Middle (MITM) attacks (table 4) [56].

We started with removing the following features: src_ip', 'dst_ip', 'src_port', 'dst_port'. All remaining features were

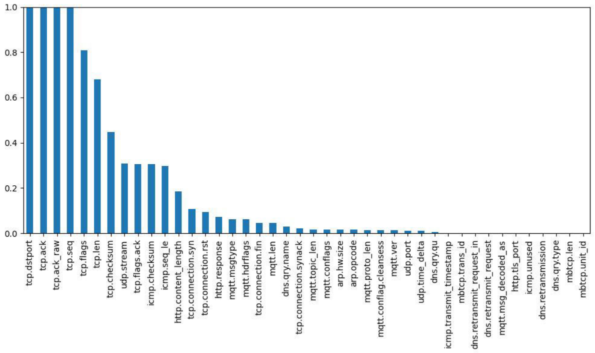


FIGURE 4. MI feature selection results for Edge-IIoTset.

label encoded using scikit-learn's LabelEncoder. After computing MI scores, we removed all features with a score lower than 0.1 [35], [57] (fig. 2). These features were: 'dns_rcode', 'missed_bytes', 'http_status_code', 'http_response_body_len', 'http_request_body_len'.

2) WUSTL-IIOT-2021

We used the original imbalanced WUSTL-IIOT-2021 dataset [41] that consists of 1,194,464 sample observations and 48 feature to extract a representative customized dataset for our simulations. For WUSTL-IIOT-2021 dataset, we deal with less than 8% attacks samples, which is closer to real world scenarios for IIoT. The data samples were collected using a IIoT network testbed over an extended period from numerous sensors, IoT devices, and machines with varying sampling frequencies, resulting in high-dimensional datasets, as detailed in [41]. This dataset focuses on four types of cyber-attacks: DoS, reconnaissance, command injection, and backdoor (table 4). The following features were removed: 'StartTime', 'LastTime', 'SrcAddr', 'DstAddr', 'sIpId', and 'dIpId' [41]. After computing MI scores, we removed all features with a score less than 0.1 (fig. 3). These features were: 'Mean', 'Proto', 'DstJitAct', 'sTos', and 'sDSb'.

3) EDGE-IIoTSET

This dataset was collected using a purpose-built IoT/IIoT testbed with a large representative set of devices, sensors, protocols, and cloud/edge configurations. The dataset considers five threats: DoS/DDoS attacks, information gathering, MITM attacks, injection attacks, and malware attacks (table 4) [42].

This dataset was collected using a purpose-built IoT/IIoT testbed with a diverse set of devices, sensors, protocols, and cloud/edge configurations. It addresses five major threat categories: DoS/DDoS attacks, information gathering, MITM attacks, injection attacks, and malware attacks (table 4) [42].

During pre-processing, we removed features in two stages:

- **Initial Removal**: Features related to specific protocol fields, timestamps, and other irrelevant attributes were excluded based on the dataset documentation. These features are summarized in table 5.
- **Post-Feature Selection**: MI scores were computed for all remaining features, and those with scores below 0.1 were excluded (fig. 4). These features are summarized in table 6.

TABLE 5. Features removed during the initial stage of preprocessing.

frame.time	ip.src_host	ip.dst_host
arp.src.proto_ipv4	arp.dst.proto_ipv4	http.request.full_uri
icmp.transmit_timestamp	http.file_data	http.request.uri.query
tcp.options	tcp.payload	tcp.srcport
tcp.dstport	udp.port	mqtt.msg

TABLE 6. Features removed based on MI scores below 0.1.

tcp.connection.rst	http.response	mqtt.msgtype
mqtt.hdrflags	tcp.connection.fin	mqtt.len
tcp.connection.synack	dns.qry.name	mqtt.topic_len
mqtt.conflags	arp.hw.size	arp.opcode
mqtt.conflag.cleansess	mqtt.proto_len	mqtt.ver
udp.port	udp.time_delta	dns.qry.qu
icmp.transmit_timestamp	mbtcp.trans_id	mqtt.msg_decoded_as
http.tls_port	icmp.unused	dns.retransmission
dns.qry.type	mbtcp.len	mbtcp.unit_id
dns.retransmit_request_in		

B. DATA BALANCING AND TRAIN-TEST SPLITTING

Data balancing techniques were applied to each dataset to address class imbalance while preserving attack class distribution. A two-step resampling approach was used, where *SMOTE* first generated synthetic samples for underrepresented classes only when the target sample count exceeded the original class count, ensuring selective oversampling. The `sampling_strategy` parameter defined the target number of samples per class, and a `random_state` of 42 was set for reproducibility. Following this, *RandomUnderSampler* reduced the size of majority classes to align with the target distribution while maintaining class proportions, using the same `random_state` for consistency. This resampling approach maintained a consistent dataset size of approximately 450,000 records (table 4) while ensuring that original class proportions were retained.

This approach effectively mitigated class imbalance while maintaining dataset representativeness. The key benefits include:

- **Preserving Class Proportions:** Balancing without distorting the original attack distribution.
- **Preventing Synthetic Data Bias:** SMOTE applied only when necessary to avoid model over-reliance on synthetic samples.
- **Reducing Overfitting Risks:** A combination of SMOTE and RandomUnderSampler improved model generalization.

A 70:30 train-test split was applied for model evaluation [58], a ratio widely recommended for efficient training and optimal predictions [59].

C. MACHINE LEARNING ALGORITHMS

We investigated the performance of traditional supervised ML techniques: DT and RF, as well as ensemble methods like Bagging, Stacking, and LGBM. The Stacking model used DT and RF as base estimators, with a Multi-Layer Perceptron (MLP) as the final estimator. Our objective was to identify a lightweight model that balances efficiency and resource utilization for effectively detecting attacks in IIoT. DT and RF offer versatility and interpretability, while LGBM is known for its efficiency and scalability. Ensemble methods such as Bagging and Stacking leverage the strengths of multiple models to enhance predictive performance [60].

IV. SIMULATION RESULTS AND DISCUSSION

We utilized Google Colaboratory and Python to evaluate the performance of ML algorithms on three model-ready datasets (table 4). The performance of the trained models was assessed using Scikit-learn's classification report, which provides metrics such as Precision, Recall, and Micro-F1, mathematically expressed as follows:

$$\text{Precision} = \frac{T_P}{T_P + F_P}$$

$$\text{Recall} = \frac{T_P}{T_P + F_N}$$

$$\text{Micro-F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Here T_P, T_N, F_P, and F_N are true positive, true negative, false positive, and false negative, respectively. Precision and Recall are calculated for each class individually, and then an overall score is averaged. While Micro-F1 specifically used to evaluate the performance of classification models in multi-class and/or multi-label settings.

Figures 5 to 7 present the simulation results for the five ML algorithms in terms of Precision, Recall, and Micro-F1 For the TON_IoT dataset, WUSTL-IIOT-2021, and Edge-IIoTset, respectively.

For the TON_IoT dataset results in fig. 5, the Precision and Recall scores show that LGBM outperforms the other models across most attack classes, particularly DoS/DDoS and Ransomware. LGBM achieves high Precision, such as 98.5% for DoS/DDoS and 97.7% for Ransomware, while also maintaining strong Recall values, particularly for Ransomware at 98.5% and Backdoor at 97.2%. This combination of high Precision and Recall contributes to the best Micro-F1 score of 97.8%, demonstrating LGBM's ability to balance false positives and false negatives effectively across these attack types.

RF also performs well, particularly for DoS/DDoS with 97.8% Precision and Ransomware with 97.7% Recall, though it lags slightly behind LGBM in Micro-F1, recording a score of 97.3%. This slight reduction in performance is due to the inherent complexity of RF in handling imbalanced datasets and potentially less effective detection of certain attack types.

DT performs significantly worse compared to the ensemble methods. DT achieves lower Precision at 89.3% and Recall at 87.6% across all attack types, with particularly poor performance on Backdoor and MITM attacks, where it fails to capture many true positives. This indicates that DT is less effective for handling complex attack patterns of imbalanced classes in the TON_IoT dataset.

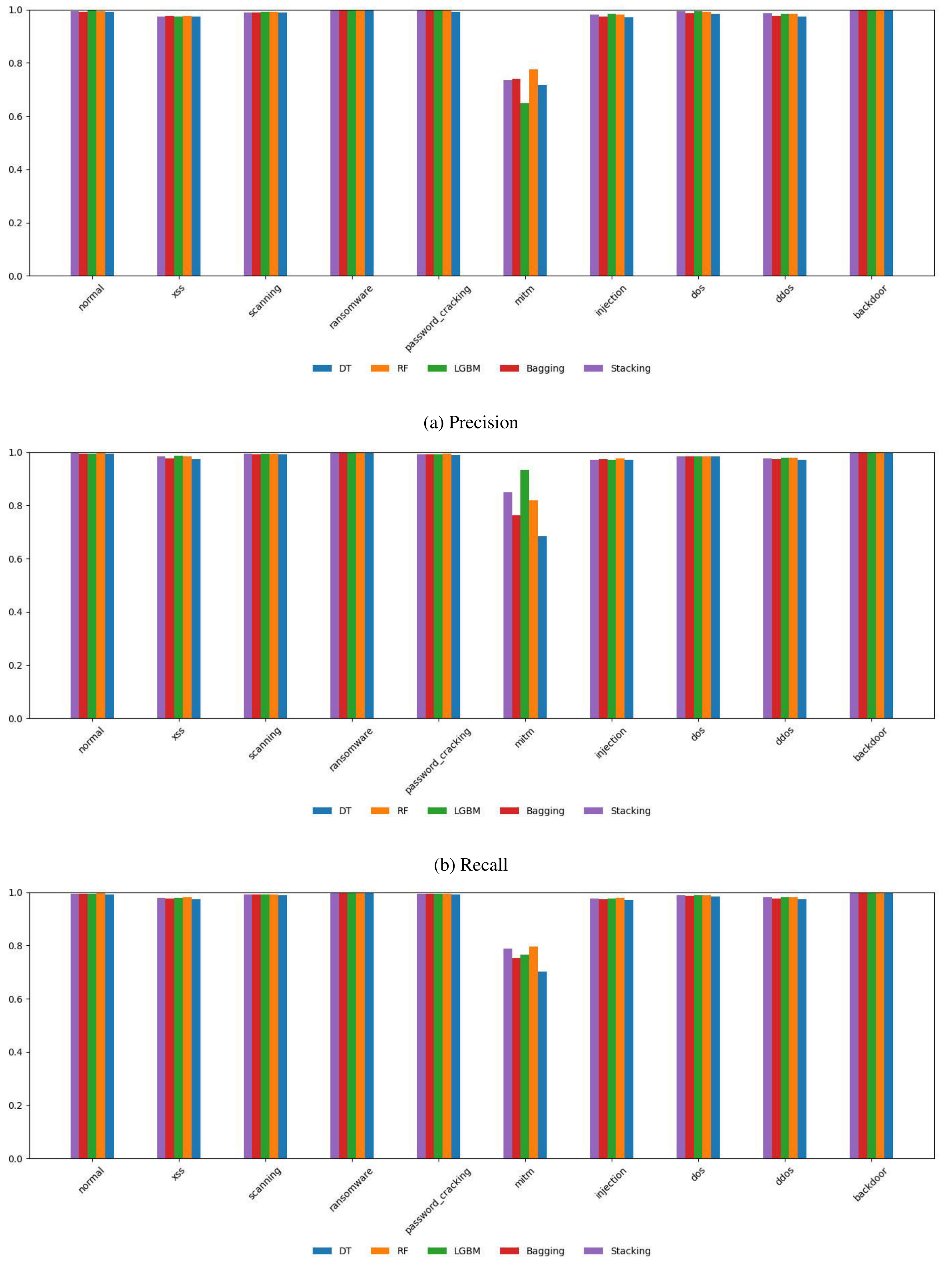
The WUSTL-IIOT-2021 dataset results in fig. 6 show that the performance difference is most evident in the detection of the Backdoor minority class, where Stacking performs the best for Precision at around 98%, while LGBM performs the worst at 81%. For Recall, LGBM performs the best in detecting Command Injection with approximately 98% and Backdoor with 81%, while RF shows the worst Recall for Backdoor at around 70%. Additionally, the results indicate that Stacking achieves the highest Micro-F1 score for Backdoor, around 87%, while DT records the lowest at approximately 78%, but the highest for Command Injection with 98.4%.

For the Edge-IIoTset results in fig. 7, Precision and Recall show the best performance across all ML models in detecting DDoS_UDP and DDoS_ICMP with around 100%. However, Precision is the lowest for LGBM, RF, and DT in detecting MITM, around 25%. In terms of Recall, the worst performance for detecting MITM is observed with Bagging and Stacking. Regarding Micro-F1, the best results for most attack classes are achieved by Bagging and Stacking, while the worst performance for detecting MITM is noted with LGBM.

Table 7 presents the simulation results for MS and TT metrics. The results reveal distinct trade-offs between the different ML techniques in terms of MS and TT across datasets. DT is the most lightweight, with MS ranging from 159KB for WUSTL-IIOT-2021 to 8.5MB for Edge-IIoTset and TT between 2–4 seconds, making it ideal for resource-constrained environments. RF shows robust predictive potential but is significantly more resource-intensive, with MS of 1.76MB for WUSTL-IIOT-2021 to 836MB for Edge-IIoTset and TTs from 25 seconds for TON_IoT to 50 seconds for Edge-IIoTset. LGBM strikes a balance with Mem, ranging from 1.7MB for WUSTL-IIOT-2021 to 5.2MB for Edge-IIoTset, and relatively fast TT of 3–6 seconds, making it scalable and efficient compared to other ensemble models. Bagging and Stacking exhibit higher resource demands. Bagging has MS ranging from 105KB for WUSTL-IIOT-2021 to 54.5MB for Edge-IIoTset with TT of 9–27 seconds. On the other hand, Stacking is the most resource-intensive, with MS spanning 1.65MB for WUSTL-IIOT-2021 to 772MB for Edge-IIoTset and TT between 46 seconds for TON_IoT and 122 seconds for Edge-IIoTset. Overall, DT and LGBM are best suited for scenarios with resource constraints, while RF and Stacking are preferable when computational resources are abundant, and predictive accuracy is the priority.

V. CROSS-DATASET TRANSFER LEARNING EXPERIMENT

In this section, we applied a cross-dataset transfer learning approach to evaluate the performance of models trained

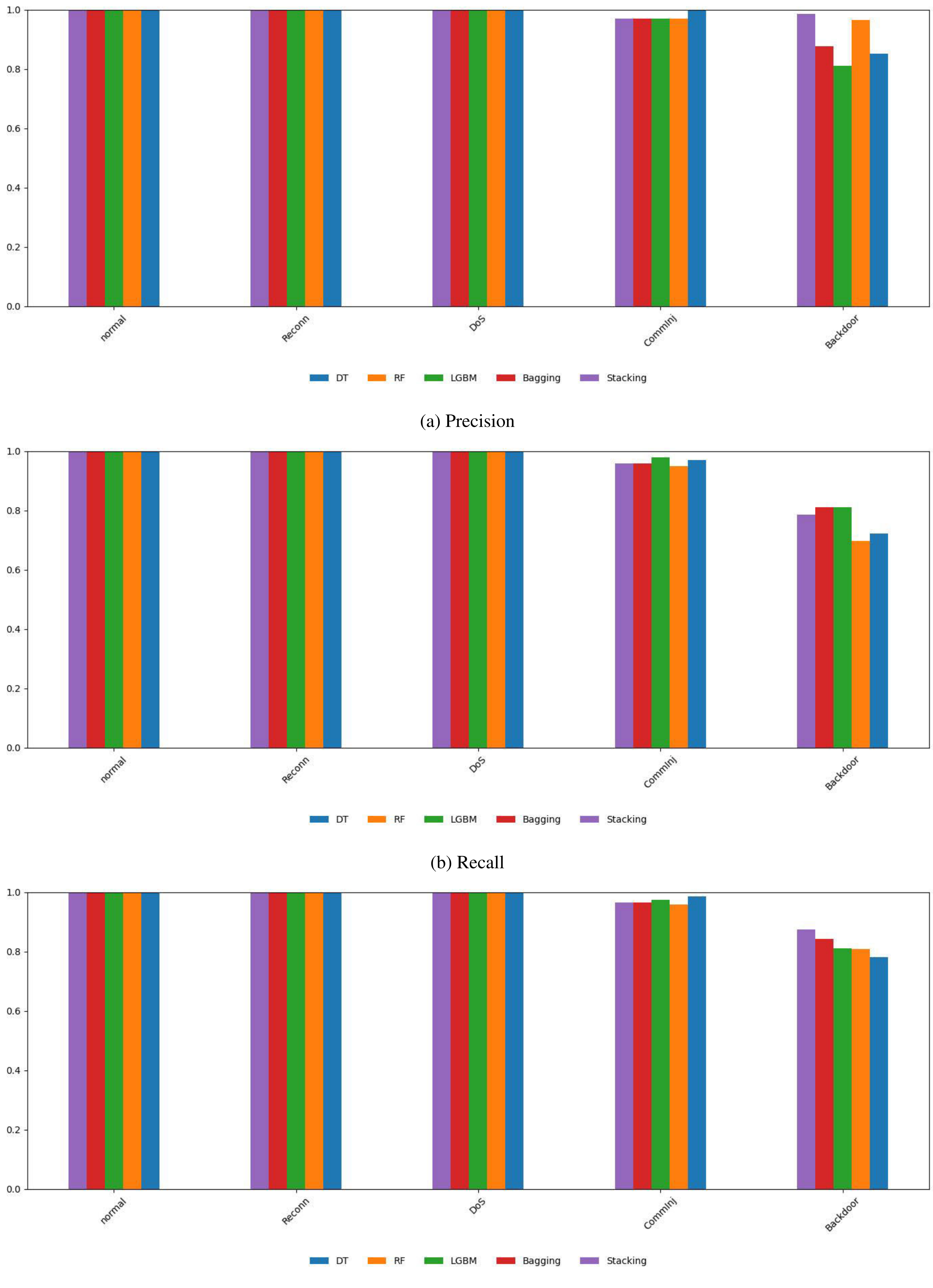


(a) Precision

(b) Recall

(c) Micro-F1

FIGURE 5. Simulation results for TON_IoT.

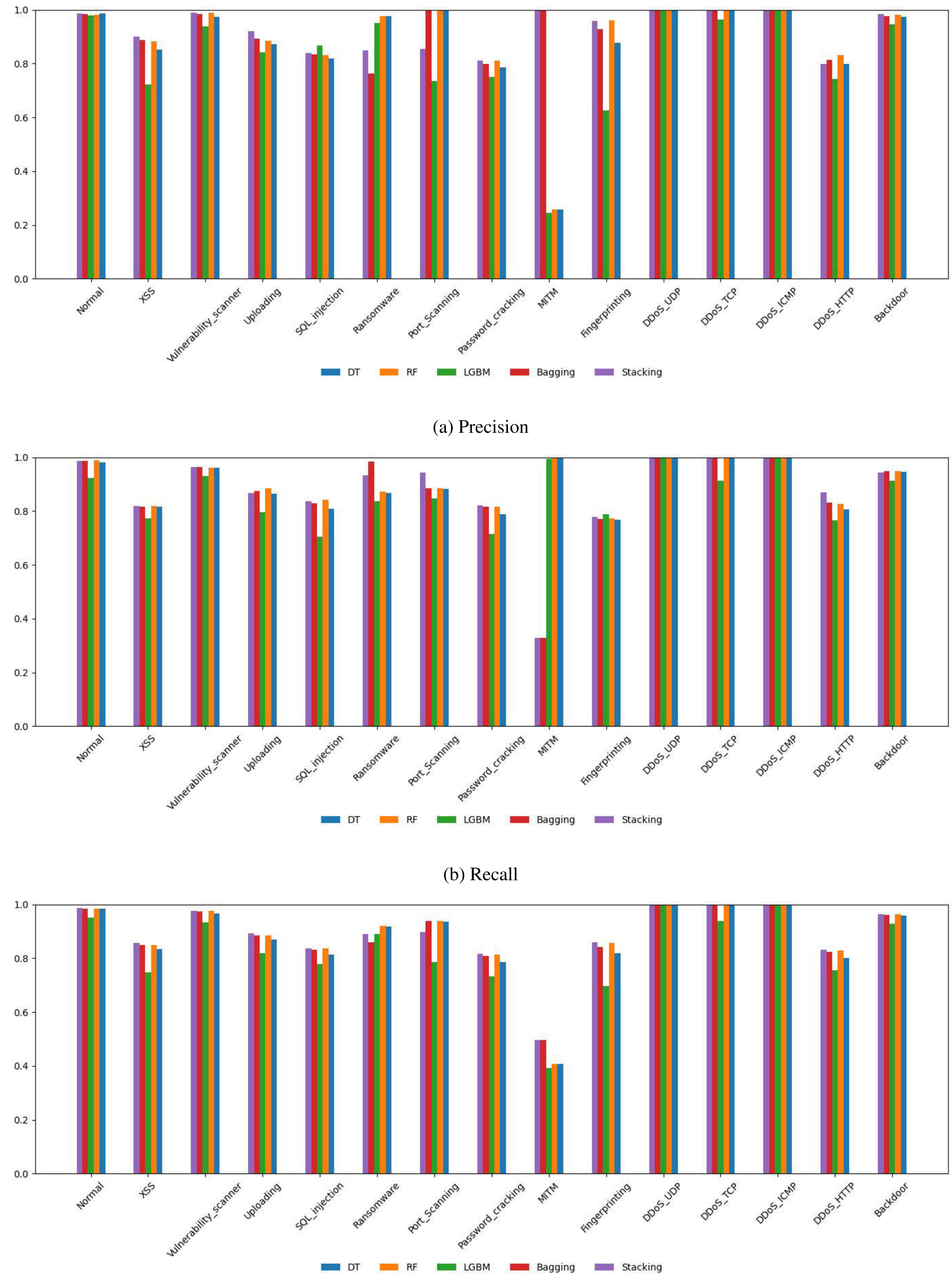


(a) Precision

(b) Recall

(c) Micro-F1

FIGURE 6. Simulation results for WUSTL-IIOT-2021.



(a) Precision

(b) Recall

(c) Micro-F1

FIGURE 7. Simulation results for Edge-IIoTset.

TABLE 7. ML results for Mem, TT metrics of TON_IoT, WUSTL-IIOT-2021, and Edge-IIoTset datasets.

ML Model	TON_IoT Dataset		WUSTL-IIOT-2021 Dataset		Edge-IIoTset Dataset	
	Mem	TT	Mem	TT	Mem	TT
DT	1.36MB	2s	159KB	3s	8.5MB	4s
RF	132MB	25s	1.76MB	4s	836MB	50s
LGBM	3.4MB	5s	1.7MB	3s	5.2MB	6s
Bagging	9.7MB	9s	105KB	27s	54.5MB	17s
Stacking	129MB	46s	1.65MB	58s	772MB	122s

on the TON_IoT dataset for predicting attack labels on the WUSTL-IIOT-2021 dataset. The TON_IoT dataset provides a realistic representation of a medium-scale network environment, while the WUSTL-IIOT-2021 testbed emulates real-world industrial systems.

The goal is to train the ML models using the TON_IoT dataset, leveraging seven common features that are among the high-importance features shared by both datasets, as well as the common attack labels: Normal, DoS, and Backdoor. These common features, listed in table 8, represent traffic characteristics relevant to distinguishing normal and attack samples. After training using TON_IoT dataset, the models were tested on the WUSTL-IIOT-2021 dataset to evaluate their ability to generalize across datasets with differing network and traffic conditions.

We evaluated the overall ACC to examine our approach. The overall ACC for the five ML models is presented in fig. 8. The DT achieved the highest accuracy of 96.25%, closely followed by Bagging with 96.02%. RF and Stacking both performed similarly, with accuracies around 93.3%. However, the LGBM model lagged behind, achieving 80.26%. These results demonstrate that tree-based ensemble models such as Bagging and Stacking show strong generalizability across datasets, whereas LGBM appears to struggle with domain shift.

TABLE 8. Mutual features between TON_IoT and WUSTL-IIOT-2021 Datasets.

TON_IoT	WUSTL-IIOT-2021
src_port	Sport
dst_port	Dport
src_pkts	SrcPkts
dst_pkts	DstPkts
src_bytes	SrcBytes
dst_bytes	DstBytes
duration	Dur
type	Traffic

The TON_IoT and WUSTL-IIOT-2021 datasets represent different environments (e.g., network vs. industrial testbed), leading to differences in traffic characteristics, noise, and feature distributions. Tree-based models like DT, RF, and Bagging are less sensitive to these shifts because they rely on splitting rules and thresholds that adapt well across domains. Conversely, LGBM uses gradient-based learning, which depends more on feature consistency and may be less robust to variations introduced by domain shift.

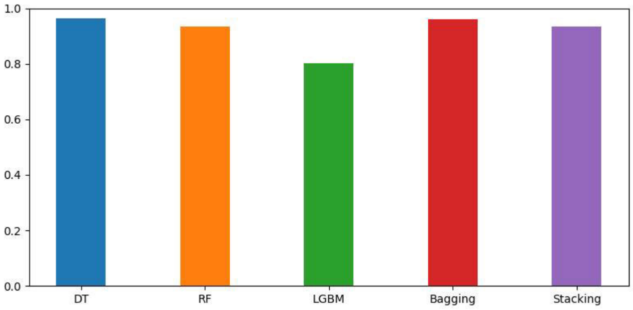


FIGURE 8. ACC results for transfer learning of TON_IoT and WUSTL-IIOT-2021.

While common features were selected, differences in their scales, distributions, or relationships with target labels between the two datasets could also impact model performance. As this approach involves cross-dataset validation, one of the primary challenges is the potential domain shift between the datasets. Datasets with minimal overlap in feature sets or attack labels could limit the ability to transfer knowledge effectively. Differences in network traffic characteristics, attack patterns, and noise in the datasets may impact the model's ability to generalize well. Other factors to consider include addressing discrepancies in data distributions, ensuring consistency in feature encoding, managing missing values, and mitigating overfitting through techniques such as domain adaptation or regularization. These considerations are crucial to improving the reliability and robustness of cross-dataset transfer learning in practical applications.

VI. REAL-TIME NETWORK TRAFFIC MONITORING AND INTRUSION DETECTION EXPERIMENT

In this section, we describe an experiment we setup to continuously monitor and analyze network traffic using Argus in combination with Tshark. Tshark captures live packet data from the network interface, while Argus processes and converts it into flow records (similar to NetFlow). The extracted flow data includes key features similar to the common features of used to train and validate the ML models in section V.

The processed flow samples are then passed to the pre-trained model to classify network traffic as either legitimate or malicious. The system operates in real-time, with predictions continuously logged for monitoring and further analysis.

The following pseudocode outlines the Python code that processes flow data for real-time network traffic monitoring, prediction, and logging, which supports the experiment's goals of continuous analysis and intrusion detection. We also employ the Linux *top* command that provides real-time insights into CPU usage, memory consumption, and network activity. As observed from table 9, RF and Stacking model use the highest memory at 191M and 190M, respectively, while the LGBM model uses the least at 142M. For all ML models, the CPU usage for the Python processing task is fixed at 15 threads, while the Tshark process uses 2 threads with a baseline memory consumption of 144M across all models.

Algorithm 1 Pseudocode for Real Time Network Traffic Monitoring and Prediction

```
BEGIN
LOAD ML model from “MLmodel.joblib”
DEFINE features = [’src_port’, ‘dst_port’, ‘src_pkts’, ‘dst_pkts’, ‘src_bytes’, ‘dst_bytes’, ’duration’]
SET command = “tshark -i ens3 -w - | argus -r - -w - | ra -n -r - -c ’,’ -s src_port dst_port src_pkts dst_pkts src_bytes dst_bytes duration”
START process using subprocess to execute command
OPEN ‘network-monitor.log’ in write mode
PRINT "Writing Results To Log..."
for each line in process standard output do
    SPLIT line into values using ’,’ delimiter
    if values contain exactly 6 elements then
        TRY
            CONVERT values into a NumPy array of floats
            PREDICT traffic type using ML model
            LOG ML prediction result with timestamp
            LOG flow feature values for reference
        CATCH
            SKIP erroneous or malformed data
    end if
end for
WAIT for process to finish
END
```

In short, the key insights of this study are presented as follows:

- Impact of Class Imbalance: Attack classes with lower representation in the dataset provide a more challenging evaluation scenario, making them better at differentiating the performance of ML models. This is because such underrepresented classes test the models' ability to correctly identify less frequent patterns, which can reveal differences in their generalization and robustness to class imbalance.

TABLE 9. Memory usage of ML models collected from btop.

ML Model	Memory Usage
DT	146M
RF	191M
LGBM	142M
Bagging	149M
Stacking	190M

for all ML models, CPU usage is 15 threads for the Python processing task

for all ML models, memory usage is 144M and CPU usage is 2 threads for the Tshark process.

- Model Size and Training Time: MS and TT increase with dataset complexity. For example, Edge-IIoTset consistently requires the largest MS and TT, reflecting its higher dimensionality and variability compared to TON_IoT and WUSTL-IIOT-2021.
- Model Efficiency: LGBM maintains low MS and TT across datasets, making it a practical choice for IoT and IIoT systems with constrained resources. In contrast, Stacking and RF require higher computational resources, particularly when working with larger datasets like Edge-IIoTset, which may limit their suitability for highly resource-constrained environments.
- Cross-Dataset Transfer Learning: The practical application of cross-dataset transfer learning was demonstrated by training models on the TON_IoT dataset (medium-scale network) and testing on the WUSTL-IIOT-2021 dataset (industrial systems). This approach highlighted challenges in transferring knowledge across datasets with minimal overlap in features or attack labels. Domain shifts between datasets require careful preprocessing to ensure reliable generalization, including feature normalization, encoding techniques, domain adaptation, and advanced sampling strategies to manage imbalanced class distributions.
- Real-Time Network Traffic Monitoring: An experiment was conducted to monitor and classify network traffic in real-time using Argus and Tshark for packet capture and flow analysis. By continuously processing and logging flow data, the system enables automated intrusion detection while maintaining system resource awareness through real-time monitoring of CPU and memory usage.

VII. CONCLUSION

In this study, we evaluated ML models across three distinct IIoT environments using the TON_IoT, WUSTL-IIOT-2021, and Edge-IIoTset datasets. These datasets, characterized by class imbalance and diverse attack scenarios, allowed us to assess the influence of class distribution on model performance. Data balancing techniques were applied to address these challenges while preserving attack class distributions.

We reviewed the literature to contextualize this study within prior works using these datasets. Our comparative analysis of supervised ML techniques for multiclass classification of cyber-attacks revealed the impact of class imbalance and identified lightweight models like LGBM as suitable for deployment in resource-constrained environments. Preprocessing and feature extraction methodologies were optimized to minimize computational overhead while improving class representation.

The study also explored cross-dataset transfer learning, highlighting the challenges posed by domain shifts and dataset variations. The results emphasize the importance of robust preprocessing techniques to ensure reliable model performance across diverse IoT and IIoT environments. Additionally, we set up a real-time network traffic monitoring and prediction experiment that enables automated intrusion detection while maintaining system resource awareness. Future work will focus on deploying an ML-based IDS on microcontroller platforms commonly used in IoT and IIoT environments, facilitating real-world evaluation of model performance, energy consumption, and latency under resource-constrained conditions.

REFERENCES

[1] IDC. (2024). *Future of Industry Ecosystems: Shared Data and Insights*. Accessed: Jun. 24, 2024. [Online]. Available: https://blogs.idc.com/2021/01/06/future-of-industry-ecosystems-shared-data-and-insights/

[2] Nokia. (2020). *Nokia Threat Intelligence Report 2020*. [Online]. Available: https://www.nokia.com/about-us/news/releases/2020/10/22/%nokia-threat-intelligence-report-warns-of-rising-cyberattacks-%on-internet-connected-devices/

[3] IBM Secur. (2023). *Cost of a Data Breach Report 2023*. [Online]. Available: https://www.ibm.com/security/data-breach

[4] S. Ismail and H. Reza, "Security challenges of blockchain-based supply chain systems," in *Proc. IEEE 13th Annu. Ubiquitous Comput., Electron. Mobile Commun. Conf. (UEMCON)*, Oct. 2022, pp. 1–6.

[5] X. Jiang, M. Lora, and S. Chattopadhyay, "An experimental analysis of security vulnerabilities in industrial IoT devices," *ACM Trans. Internet Technol.*, vol. 20, no. 2, pp. 1–24, May 2020.

[6] S. Ismail, T. T. Khoei, R. Marsh, and N. Kaabouch, "A comparative study of machine learning models for cyber-attacks detection in wireless sensor networks," in *Proc. IEEE 12th Annu. Ubiquitous Comput., Electron. Mobile Commun. Conf. (UEMCON)*, Dec. 2021, pp. 0313–0318.

[7] Z. E. Huma, S. Latif, J. Ahmad, Z. Idrees, A. Ibrar, Z. Zou, F. Alqahtani, and F. Baothman, "A hybrid deep random neural network for cyberattack detection in the industrial Internet of Things," *IEEE Access*, vol. 9, pp. 55595–55605, 2021.

[8] 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," *Future Gener. Comput. Syst.*, vol. 100, pp. 779–796, Nov. 2019.

[9] M. Al-Hawawreh, E. Sitnikova, and N. Aboutorab, "X-IIoTID: A connectivity-agnostic and device-agnostic intrusion data set for industrial Internet of Things," *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3962–3977, Mar. 2022.

[10] J. G. Almaraz-Rivera, J. A. Perez-Diaz, J. A. Cantoral-Ceballos, J. F. Botero, and L. A. Trejo, "Toward the protection of IoT networks: Introducing the LATAM-DDoS-IoT dataset," *IEEE Access*, vol. 10, pp. 106909–106920, 2022.

[11] D. Hamouda, M. A. Ferrag, N. Benhamida, and H. Seridi, "Intrusion detection systems for industrial Internet of Things: A survey," in *Proc. Int. Conf. Theor. Applicative Aspects Comput. Sci. (ICTAACS)*, Dec. 2021, pp. 1–8.

[12] K. Bansal and A. Singhrova, "Review on intrusion detection system for IoT/IIoT -brief study," *Multimedia Tools Appl.*, vol. 83, no. 8, pp. 23083–23108, Aug. 2023.

[13] M. Nuaimi, L. C. Fourati, and B. B. Hamed, "Intelligent approaches toward intrusion detection systems for industrial Internet of Things: A systematic comprehensive review," *J. Netw. Comput. Appl.*, vol. 215, Jun. 2023, Art. no. 103637.

[14] P. L. S. Jayalaxmi, R. Saha, G. Kumar, M. Conti, and T.-H. Kim, "Machine and deep learning solutions for intrusion detection and prevention in IoTs: A survey," *IEEE Access*, vol. 10, pp. 121173–121192, 2022.

[15] P. Jayalaxmi, R. Saha, G. Kumar, N. Kumar, and T.-H. Kim, "A taxonomy of security issues in industrial Internet-of-Things: Scoping review for existing solutions, future implications, and research challenges," *IEEE Access*, vol. 9, pp. 25344–25359, 2021.

[16] A. R. Gad, M. Haggag, A. A. Nashat, and T. M. Barakat, "A distributed intrusion detection system using machine learning for IoT based on ToN-IoT dataset," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 6, pp. 1–16, 2022.

[17] A. R. Gad, A. A. Nashat, and T. M. Barkat, "Intrusion detection system using machine learning for vehicular ad hoc networks based on ToN-IoT dataset," *IEEE Access*, vol. 9, pp. 142206–142217, 2021.

[18] G. Guo, X. Pan, H. Liu, F. Li, L. Pei, and K. Hu, "An IoT intrusion detection system based on TON IoT network dataset," in *Proc. IEEE 13th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Mar. 2023, pp. 0333–0338.

[19] O. Belarbi, T. Spyridopoulos, E. Anthi, I. Mavromatis, P. Carnelli, and A. Khan, "Federated deep learning for intrusion detection in IoT networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, vol. 3125, Dec. 2023, pp. 85–99.

[20] A. Oseni, N. Moustafa, G. Creech, N. Sohrabi, A. Strelzoff, Z. Tari, and I. Linkov, "An explainable deep learning framework for resilient intrusion detection in IoT-enabled transportation networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 1, pp. 1000–1014, Jan. 2023.

[21] S. Latif, Z. E. Huma, S. S. Jamal, F. Ahmed, J. Ahmad, A. Zahid, K. Dashtipour, M. U. Aftab, M. Ahmad, and Q. H. Abbasi, "Intrusion detection framework for the Internet of Things using a dense random neural network," *IEEE Trans. Ind. Informat.*, vol. 18, no. 9, pp. 6435–6444, Sep. 2022.

[22] V. Malik, R. Mittal, D. Mavaluru, B. R. Narapureddy, S. B. Goyal, R. J. Martin, K. Srinivasan, and A. Mittal, "Building a secure platform for digital governance interoperability and data exchange using blockchain and deep learning-based frameworks," *IEEE Access*, vol. 11, pp. 70110–70131, 2023.

[23] D. Musleh, M. Alotaibi, F. Alhaidari, A. Rahman, and R. M. Mohammad, "Intrusion detection system using feature extraction with machine learning algorithms in IoT," *J. Sensor Actuator Netw.*, vol. 12, no. 2, p. 29, Mar. 2023.

[24] M. M. Shtayat, M. K. Hasan, R. Sulaiman, S. Islam, and A. U. R. Khan, "An explainable ensemble deep learning approach for intrusion detection in industrial Internet of Things," *IEEE Access*, vol. 11, pp. 115047–115061, 2023.

[25] M. Wang, N. Yang, and N. Weng, "Securing a smart home with a transformer-based IoT intrusion detection system," *Electronics*, vol. 12, no. 9, p. 2100, May 2023.

[26] S. Li, G. Chai, Y. Wang, G. Zhou, Z. Li, D. Yu, and R. Gao, "CRSF: An intrusion detection framework for industrial Internet of Things based on pretrained CNN2D-RNN and SVM," *IEEE Access*, vol. 11, pp. 92041–92054, 2023.

[27] M. S. Alshehri, O. Saidani, F. S. Alrayes, S. F. Abbasi, and J. Ahmad, "A self-attention-based deep convolutional neural networks for IIoT networks intrusion detection," *IEEE Access*, vol. 12, pp. 45762–45772, 2024.

[28] I. Tareq, B. M. Elbagoury, S. El-Regaily, and E.-S.-M. El-Horbaty, "Analysis of ToN-IoT, UNW-NB15, and edge-IIoT datasets using DL in cybersecurity for IoT," *Appl. Sci.*, vol. 12, no. 19, p. 9572, Sep. 2022.

[29] A. H. Farea and K. Küçük, "Machine learning-based intrusion detection technique for IoT: Simulation with cooja," *Int. J. Comput. Netw. Inf. Secur.*, vol. 16, no. 1, pp. 1–23, Feb. 2024.

[30] M. M. Rashid, S. U. Khan, F. Eusufzai, M. A. Redwan, S. R. Sabuj, and M. Elsharief, "A federated learning-based approach for improving intrusion detection in industrial Internet of Things networks," *Network*, vol. 3, no. 1, pp. 158–179, Jan. 2023.

[31] A. M. Eid, B. Soudan, A. B. Nassif, and M. Injadat, "Enhancing intrusion detection in IIoT: Optimized CNN model with multi-class SMOTE balancing," *Neural Comput. Appl.*, vol. 36, no. 24, pp. 14643–14659, Aug. 2024.

[32] A. M. Eid, A. B. Nassif, B. Soudan, and M. N. Injadat, "IIoT network intrusion detection using machine learning," in *Proc. 6th Int. Conf. Intell. Robot. Control Eng. (IRCE)*, Aug. 2023, pp. 196–201.

[33] I. Tareq, S. Amin, B. M. Elbagoury, and E.-S. M. El-Horabty, "Machine learning for detecting Internet of Things network cyber-attacks," *Int. J. Intell. Comput. Inf. Sci.*, vol. 24, no. 2, pp. 18–27, Jun. 2024.

[34] T. Gaber, J. B. Awotunde, S. O. Folorunso, S. A. Ajagbe, and E. Eldesouky, "Industrial Internet of Things intrusion detection method using machine learning and optimization techniques," *Wireless Commun. Mobile Comput.*, vol. 2023, pp. 1–15, Apr. 2023.

[35] S. Ismail, S. Dandan, D. W. Dawoud, and H. Reza, "A comparative study of lightweight machine learning techniques for cyber-attacks detection in blockchain-enabled industrial supply chain," *IEEE Access*, vol. 12, pp. 102481–102491, 2024.

[36] A. M. Eid, B. Soudan, A. B. Nassif, and M. Injadat, "Comparative study of ML models for IIoT intrusion detection: Impact of data preprocessing and balancing," *Neural Comput. Appl.*, vol. 36, no. 13, pp. 6955–6972, May 2024.

[37] B. Xu, L. Sun, X. Mao, R. Ding, and C. Liu, "IoT intrusion detection system based on machine learning," *Electronics*, vol. 12, no. 20, p. 4289, Oct. 2023.

[38] F. Mesadieu, D. Torre, and A. Chennamaneni, "Leveraging deep reinforcement learning technique for intrusion detection in SCADA infrastructure," *IEEE Access*, vol. 12, pp. 63381–63399, 2024.

[39] R. S. Tiwari, D. Lakshmi, T. K. Das, A. K. Tripathy, and K.-C. Li, "A lightweight optimized intrusion detection system using machine learning for edge-based IIoT security," *Telecommun. Syst.*, vol. 87, no. 3, pp. 605–624, Nov. 2024.

[40] M. Mohy-Eddine, A. Guezzaz, S. Benkirane, and M. Azrour, "An effective intrusion detection approach based on ensemble learning for IIoT edge computing," *J. Comput. Virol. Hacking Techn.*, vol. 19, no. 4, pp. 469–481, Dec. 2022.

[41] M. Zolanvari, M. A. Teixeira, L. Gupta, K. M. Khan, and R. Jain, "Machine learning-based network vulnerability analysis of industrial Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6822–6834, Aug. 2019.

[42] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning," *IEEE Access*, vol. 10, pp. 40281–40306, 2022.

[43] Z. Ahmed and S. S. Askar, "EdgeGuard: Machine learning for proactive intrusion detection on edge networks," *Artif. Intell. Cybersecurity*, vol. 1, pp. 37–43, Jun. 2024.

[44] G. P. Bhandari, A. Lyth, A. Shalaginov, and T.-M. Grønli, "Artificial intelligence enabled middleware for distributed cyberattacks detection in IoT-based smart environments," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2022, pp. 3023–3032.

[45] Z. Benamor, Z. A. Seghir, M. Djezzar, and M. Hemam, "A comparative study of machine learning algorithms for intrusion detection in IoT networks," *Revue d'Intell. Artificielle*, vol. 37, no. 3, pp. 567–576, Jun. 2023.

[46] S. I. Popoola, A. L. Imoize, M. Hammoudeh, B. Adebisi, O. Jogunola, and A. M. Aibinu, "Federated deep learning for intrusion detection in consumer-centric Internet of Things," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 1610–1622, Feb. 2024.

[47] M. Ramaiah and M. Y. Rahamathulla, "Securing the industrial IoT: A novel network intrusion detection models," in *Proc. 3rd Int. Conf. Artif. Intell. For Internet Things (AIIoT)*, May 2024, pp. 1–6.

[48] A. Khacha, R. Saadouni, Y. Harbi, and Z. Aliouat, "Hybrid deep learning-based intrusion detection system for industrial Internet of Things," in *Proc. 5th Int. Symp. Informat. Appl. (ISIA)*, Nov. 2022, pp. 1–6.

[49] N. Hamza, H. Lakmal, M. Maduranga, and R. Kathriarachchi, "Malware detection of IoT networks using machine learning: An experimental study with edge IIoT dataset," in *Proc. 30th Annu. Tech. Conf. IET Sri Lanka Net.*, Colombo, Sri Lanka, 2023, pp. 1–18.

[50] Z. A. El Houda, B. Brik, A. Ksentini, and L. Khoukhi, "A MEC-based architecture to secure IoT applications using federated deep learning," *IEEE Internet Things Mag.*, vol. 6, no. 1, pp. 60–63, Mar. 2023.

[51] K. Keserwani, A. Aggarwal, and A. Chauhan, "Attack detection in industrial IoT using novel ensemble techniques," in *Proc. 2nd Int. Conf. Vis. Towards Emerg. Trends Commun. Netw. Technol. (ViTECoN)*, May 2023, pp. 1–6.

[52] J. A. Beauty Angelin and C. Priyadharsini, "Deep learning based network based intrusion detection system in industrial Internet of Things," in *Proc. 2nd Int. Conf. Intell. Data Commun. Technol. Internet Things (IDCIoT)*, Jan. 2024, pp. 426–432.

[53] R. Saadouni, A. Khacha, Y. Harbi, C. Gherbi, S. Harous, and Z. Aliouat, "Secure IIoT networks with hybrid CNN-GRU model using edge-IIoTset," in *Proc. 15th Int. Conf. Innov. Inf. Technol. (IIT)*, Nov. 2023, pp. 150–155.

[54] G. Bhandari, A. Lyth, A. Shalaginov, and T.-M. Grønli, "Distributed deep neural-network-based middleware for cyber-attacks detection in smart IoT ecosystem: A novel framework and performance evaluation approach," *Electronics*, vol. 12, no. 2, p. 298, Jan. 2023.

[55] A. Sharma, H. Babbar, and A. Sharma, "TON-IoT: Detection of attacks on Internet of Things in vehicular networks," in *Proc. 6th Int. Conf. Electron., Commun. Aerosp. Technol.*, Dec. 2022, pp. 539–545.

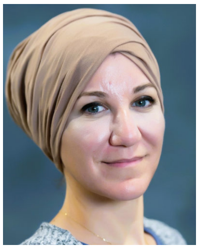
[56] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," *Sustain. Cities Soc.*, vol. 72, Sep. 2021, Art. no. 102994.

[57] S. Ismail, D. W. Dawoud, and H. Reza, "A comparative study of datasets for cyber-attacks detection in wireless sensor networks," in *Proc. IEEE 3rd Int. Conf. Comput. Mach. Intell. (ICMI)*, Apr. 2024, pp. 1–6.

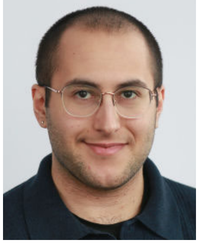
[58] I. Muraina, "Ideal dataset splitting ratios in machine learning algorithms: General concerns for data scientists and data analysts," in *Proc. 7th Int. Mardin Artuklu Sci. Res. Conf.*, 2022, pp. 1–18.

[59] B. Vrigazova, "The proportion for splitting data into training and test set for the bootstrap in classification problems," *Bus. Syst. Res. J.*, vol. 12, no. 1, pp. 228–242, May 2021.

[60] S. Ismail, Z. El Mrabet, and H. Reza, "An ensemble-based machine learning approach for cyber-attacks detection in wireless sensor networks," *Appl. Sci.*, vol. 13, no. 1, p. 30, Dec. 2022.



SHEREEN ISMAIL (Senior Member, IEEE) received the B.Sc. degree in computer engineering and the M.Sc. degree in computer engineering and networks from the University of Jordan, Amman, Jordan, and the Ph.D. degree in computer science from the University of North Dakota, USA. Throughout her doctoral studies, she was a Graduate Research Assistant with the School of Electrical Engineering and Computer Science. She taught undergraduate level with American University of Ras Al-Khaimah, the Al-Zaytoonah University of Jordan, and the Applied Science University of Jordan. Recently, she joined Merit Network Inc., as a Research Scientist in networking and cyber-security. Her research focuses on wireless networks and cyber-security. Her contributions have been recognized through scholarship awards at prestigious conferences, including 2023 IEEE International Symposium on Women in Services Computing (WISC 2023) and Women in CyberSecurity (WiCyS 2024).



SALAH DANDAN is currently pursuing the bachelor's degree in electrical engineering and computer science with the University of North Dakota. He is also an Undergraduate Research Assistant, contributing to projects focused on GPU acceleration, high-performance computing, and the application of machine learning in cyber-security.

ALA'A QUSHOU received the B.Sc. degree in computer engineering from Applied Science Private University, Amman, Jordan, and the M.Sc. degree in computer engineering and networks from the University of Jordan, Amman, in 2022. Since 2023, she has been a Lecturer and a Research Assistant with the University of Jordan. Her research interests include wireless sensor networks, the IoT, artificial intelligence, and cyber-security.

• • •