Edge Learning for 6G-enabled Internet of Things

Edge Learning for 6G-enabled Internet of Things: A Comprehensive Survey of Vulnerabilities, Datasets, and Defenses

Mohamed Amine Ferrag, *Senior Member, IEEE*, Othmane Friha, Burak Kantarci, *Senior Member, IEEE*, Norbert Tihanyi, *Member, IEEE*, Lucas Cordeiro, Merouane Debbah, *Fellow, IEEE*, Djallel Hamouda, Muna Al-Hawawreh, Kim-Kwang Raymond Choo, *Senior Member, IEEE*

***Abstract*—The ongoing deployment of the fifth-generation (5G) wireless networks constantly reveals limitations concerning its original concept as a key driver of Internet of Everything (IoE) applications. These 5G challenges are behind worldwide efforts to enable future networks, such as sixth-generation (6G) networks, to efficiently support sophisticated applications ranging from autonomous driving capabilities to the Metaverse. Edge learning is a new and powerful approach to training models across distributed clients while protecting the privacy of their data. This approach is expected to be embedded within future network infrastructures, including 6G, to solve challenging problems such as resource management and behavior prediction. However, edge learning in general, and distributed deep learning, in particular, have been discovered to be susceptible to tampering and manipulation. This survey article provides a holistic review of the most recent research focused on edge learning vulnerabilities and defenses for 6G-enabled IoT. We summarize the existing surveys on machine learning for 6G–IoT security and machine learning-associated threats in three different learning modes: centralized, federated, and distributed. Then, we provide an overview of enabling emerging technologies for 6G–IoT intelligence. Moreover, we provide a holistic survey of existing research on attacks against machine learning and classify threat models into eight categories, including backdoor attacks, adversarial examples, combined attacks, poisoning attacks, Sybil attacks, byzantine attacks, inference attacks, and dropping attacks. In addition, we provide a comprehensive and detailed taxonomy and a side-by-side comparison of the state-of-the-art defense methods against edge learning vulnerabilities. Finally, as new attacks and defense technologies are realized, new research and future overall prospects for 6G-enabled IoT are discussed.**

M. A. Ferrag is the corresponding author.

M. A. Ferrag is with Technology Innovation Institute, 9639 Masdar City, Abu Dhabi, United Arab Emirates email: mohamed.ferrag@tii.ae

O. Friha is with Networks and Systems Laboratory (LRS), Badji Mokhtar-Annaba University, B.P.12, Annaba 23000, Algeria, email: othmane.friha@univ-annaba.org

B. Kantarci is with School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, ON, Canada email: Burak.Kantarci@uottawa.ca

N. Tihanyi is with Technology Innovation Institute, 9639 Masdar City, Abu Dhabi, United Arab Emirates email: norbert.tihanyi@tii.ae

L. Cordeiro is with the University of Manchester, UK and Technology Innovation Institute, UAE, email: lucas.cordeiro@manchester.ac.uk

M. Debbah is with Khalifa University of Science and Technology, P O Box 127788, Abu Dhabi, UAE email: merouane.debbah@ku.ac.ae

D. Hamouda is with Labstic Laboratory, Department of Computer Science, Guelma University, B.P. 401, 24000, Algeria e-mail: hamouda.djallel@univ-guelma.dz

Muna Al-Hawawreh is with School of Information Technology, Deakin University, Australia email: muna.alhawawreh@deakin.edu.au

K.-K. R. Choo is with the Department of Information Systems and Cyber Security, University of Texas at San Antonio, San Antonio, TX 78249-0631, USA. email: raymond.choo@fulbrightmail.org

***Index Terms*—Edge Learning, 6G, IoT, Federated Learning, AI vulnerabilities, Security.**

List of Abbreviations

5G	Fifth-Generation
6G	Sixth-Generation
AEA	Auto-Encoder with Attention
AI	Artificial Intelligence
APT	Advanced Persistent Threat
BCD	Bayesian Compromise Detection
CNN	Convolutional Neural Network
CoAP	Constrained Application Protocol
CPMS	Control Plane Micro Services
CVAE	Conditional Variational Autoencoder
CNDF	Core Network Data Analytics Function
DL	Deep Learning
DLT	Distributed Ledger Technologies
DP	Differential Privacy
DPI	Deep Packet Inspection
DQN	Deep Q-network
DNN	Deep Neural Network
DRL	Deep Reinforcement Learning
eMBB	Enhanced Mobile Broadband
FD	Federated Distillation
FDD	Frequency Division Duplexing
FGSM	Fast Gradient Sign Method
FL	Federated Learning
GAN	Generative Adversarial Network
GRUs	Gated Recurrent Units
HAR	Human Activity Recognition
HE	Homomorphic Encryption
HFL	Horizontal Federated Learning
HAI	Human-Centered Artificial Intelligence
IID	Independent and Identically Distributed
IoE	Internet of Everything
IoMT	Internet of Medical Things
IoT	Internet of Things
IIoT	Industrial Internet of Things
JSCC	Joint Source-Channel Coding
KPCA	Kernel Principal Component Analysis
KPI	Key Performance Indicators

LSTM	Long Short Term Memory
MAC	Medium Access Control
MDP	Markov Decision Process
MQTT	Message Queuing Telemetry Transport
MEC	Mobile Edge Computing
MIA	Membership Inference Attack
MIMO	Multiple-Input Multiple-Output
MTD	Moving Target Defense
mMIMO	massive MIMO
MTD	Moving Target Defense
NFV	Network Functions Virtualization
NOMA	Non-Orthogonal Multi-Access
NI	Network Intelligence
NIDS	Network Intrusion Detection Systems
NLP	Natural Language Processing
NOMA	Non-Orthogonal Multi-Access
Non-IID	Non-Independent and Identically Distributed
ODT	Opportunistic Data Transfer
P2P	Peer-to-Peer
PoF	Proof of Federation
RL	Reinforcement Learning
RNN	Recurrent Neural Network
SDN	Software-Defined Networking
SGD	Stochastic Gradient Descent
SNN	Self-Sustaining Network
SHA	Secure Hash Algorithm
SVM	Support Vector Machine
SONs	Self-Organizing Networks
TFL	Transfer Federated Learning
UE	User Equipment
UPMS	User Plane Micro Services
URLLC	Ultra-Reliable, Low-Latency Communications
VFL	Vertical Federated Learning
VRF	Verifiable Random Function
VLC	Visible Light Communication
VM	Virtual Machine
XAI	Explainable AI
XR	Extended Reality
xURLLC	eXtreme URLLC

I. Introduction

We are experiencing accelerated development, increasing adoption, and innovative combinations of information and communication technologies, such as cloud and edge computing, the Internet of Things (IoT), massive data analytics, and Artificial Intelligence (AI). This widespread endorsement is driven by many factors, including the widespread availability of broadband communications, which are expected to move the world into an all-connected zone. One instance of such a combination is the integration of AI into the fifth-generation (5G) wireless networks. However, it is only intended to operate in specific areas under specific conditions (huge data and robust computing) [1]. This alliance is expected to be much tighter in future generations, starting with the upcoming sixth-generation (6G) wireless networks, as AI is expected to be a core component in it [2]. In addition, with Mobile Edge Computing (MEC) provides the possibility of processing large volumes of data by edge devices, and distributed learning paradigms such as Federated Learning (FL) which enable multiple parties to collaborate in building shared Machine Learning models without sharing their data. There is a lot of interest in making the edge intelligent, an emerging research area known as *distributed edge learning* [3].

The key driver for the advancement of wireless networks has been the requirement for higher data rates, which necessitated a continuous boost in network capacity. The ongoing rise of the IoE, which is defined by Cisco as the "*networked connection of people, process, data, and things*"[1], in which billions of devices are plugged in and exchanging large amounts of data continuously, has led to a fundamental upgrade from enhanced mobile broadband (eMBB) services to ultra-reliable, low-latency communications (URLLC) [4]. While currently commercialized 5G in the ground will comfortably handle core IoE and URLLC services, as well as potentially going to support fixed access to mmWave frequencies, early 5G deployments are expected to utilize sub-6 GHz frequencies to support mobility, making it questionable whether they can provide the IoE applications of future smart cities [4]. To address these issues, 6G networks are envisioned to be the solution with AI as an indispensable component [5].

Currently, not only have AI and IoT demonstrated their potential benefits in various spheres, but their synergistic effect is seen as a key factor in transforming the future, including Industry 4.0, Agriculture 4.0, and 6G communication networks. A good practical, real-world example of such cooperation is the development of research into autonomous vehicles. Furthermore, from an economic standpoint, both fields are emerging. In its 2022 AI Index report [2], the Stanford Institute for Human-Centered Artificial Intelligence (HAI) states that investment in AI surpassed $46 billion to reach $93.5 billion between 2020 and 2021, with the largest growth in investment coming from the global private sector. In addition, in the same context, the worldwide cellular IoT market size is projected to expand to $61 billion by 2026 (from $31 billion in 2022) [3]. The preceding statistics point out the success and breakthrough achieved through AI and IoT, which are virtually employed across all areas of daily lives, ranging from military, industry, healthcare, education, and entertainment, to name a few.

Although both AI and IoT are considered somewhat mature from a variety of perspectives, including efficiency and operability [7], as far as their security is concerned, they are viewed as being in the early developmental stages, and further progress is required to strengthen their robustness against cyber-attacks [8]–[10]. To illustrate, in 2021, a widely used neural networks framework was found to have more than 300 classical security vulnerabilities, including overflows, memory corruption, bypasses, information leaks, and code executions [4]. A riskier pathway is the adversarial example, where core-

[1]https://www.cisco.com/c/dam/en_us/about/businessinsights/docs/ioevalue-indexfaq.pdf

[2]https://aiindex.stanford.edu/wp-content/uploads/2022/03/2022-AI-Index-Report_Master.pdf

[3]https://www.juniperresearch.com/press/cellular-iot-market-value-to-exceed-61b-globally

[4]https://www.cvedetails.com/product/53738/Google-Tensorflow.html

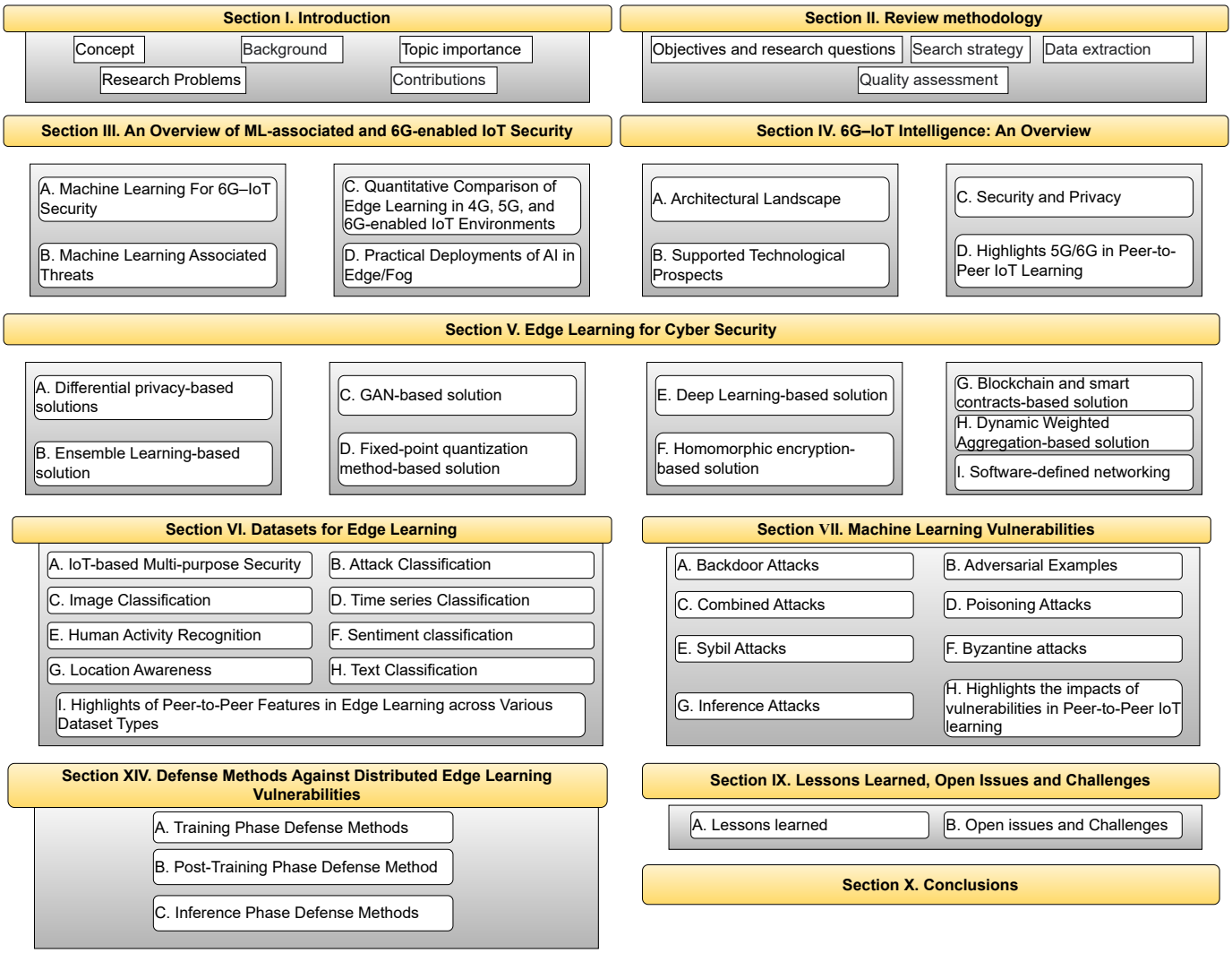


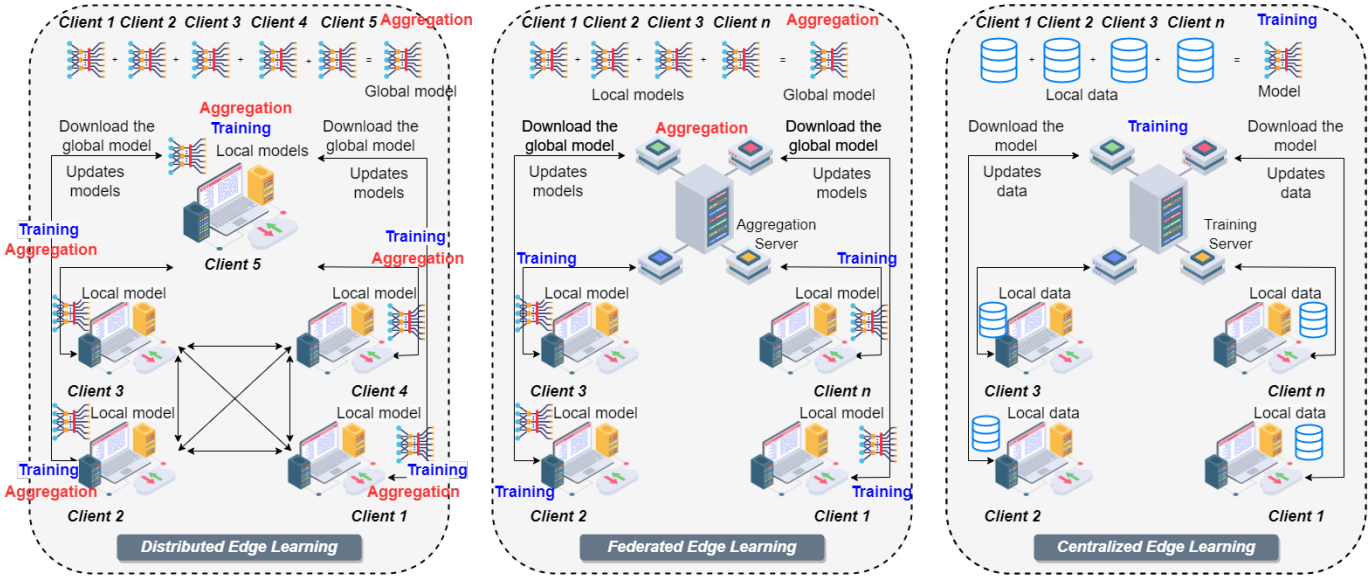
Fig. 1: Structure of this survey paper.

engineered manipulations or more refined implementation of AI methods are employed to produce synthetic input data to trigger malfunctions in the targeted AI systems. As an example, a group of Skylight researchers identified a specific bias toward a specific pattern in the Cylance AI-based antivirus product, which enabled the development of a workaround by adding a selected list of strings to a malware package, altering identification scores significantly, and avoiding malware detection, with a 100% success rate for the top 10 malware (2019), and near 90% for a broader sample of 380+ malware [5]. Not to mention the widespread attacks on Industrial IoT-based infrastructures in recent years, especially with sophisticated reconnaissance tools and search engines such as Shodan becoming publicly available on the Internet. We aim to shed light on these important issues within this paper (Figure 1).

As we delve into cybersecurity and ML, we must recognize the interconnectedness and interdependence of two primary concepts: "security for ML" and "ML for security". The former focuses on utilizing ML techniques to address traditional security attacks, such as denial of service (DoS), man-in-the-middle, malware, and intrusion detection. In contrast, the latter deals with the inherent vulnerabilities and potential attacks targeting ML systems, such as data poisoning. This paper explores both concepts concurrently rather than presenting them as separate entities. This approach may initially appear to create confusion; however, it is a deliberate decision driven by our belief that the two domains cannot be studied in isolation. By examining their relationship holistically, we aim to provide a comprehensive understanding of the challenges and opportunities that arise when using ML in security applications. Throughout the paper, we will demonstrate how advancements in 'security for ML' can contribute to more robust and resilient 'ML for security' solutions and vice versa. We believe that emphasizing the interplay between these areas is crucial for encouraging researchers and practitioners to develop innovative, integrated strategies that ensure the security and reliability of ML-based systems.

Figure 2 illustrates the differentiation between Distributed Edge Learning, Federated Edge Learning, and Centralized

[5]https://skylightcyber.com/2019/07/18/cylance-i-kill-you/



- Aggregation: Each client aggregate[s] the model updates.
- Training: Each selected client then performs training on the model based on its local dataset.
- Download the global model: Download the global model from selected clients.
- Updates models: Sends an update to each selected client.
- Clients: IoT devices are equally privileged to form a peer-to-peer network. Each client can perform train the model as well as aggregate the model updates.

- Aggregation: The edge server aggregate[s] the model updates.
- Training: Each selected client then performs training on the model based on its local dataset.
- Download the global model: Download the global model from the Edge server.
- Updates models: Each client sends the update to the Edge server.
- Clients: Each client has a local training dataset which is never uploaded to the Edge server.
- Aggregation server: Performs model averaging.

- Training: The edge server trains the model based on the aggregated dataset.
- Download the model: Download the model from the Edge server.
- Updates data: Each client has a local training dataset which is uploaded to the Edge server.
- Training server: Performs the aggregation of datasets.

Fig. 2: Distributed Edge Learning vs. Federated Edge Learning vs. Centralized Edge Learning.

Edge Learning. Centralized Edge Learning refers to the traditional approach to machine learning where data is collected and stored in a central location, such as an edge server or data center. In this approach, a single machine-learning model is trained using all the data available in the central location. The model is then used to make predictions on new data. On the other hand, Federated Edge Learning is a decentralized approach to machine learning where data is stored on multiple devices or servers. In this approach, the model is trained collaboratively using data from all the devices or servers without sharing the raw data. The devices or servers send model updates to a central edge server, aggregating them to form a new model. This process is repeated iteratively until the model converges to a desired level of accuracy. Distributed edge learning refers to a machine learning paradigm where a large-scale model is trained across a network of distributed devices, such as IoT devices. In this approach, each device or client is responsible for training the model using its local dataset, and then sending the updated model parameters to aggregators, which combines these updates to produce a new version of the model. Each device or client is equally privileged and forms a peer-to-peer network. Each device can communicate directly with its neighbors without a central coordinating authority. This decentralized approach is highly scalable and fault-tolerant, as it can continue to operate even if some devices fail or leave the network.

The increased reliability and the enhanced utility of AI and IoT are obvious and potentially useful but they also lead to a novel and unique attack surface with cyber vulnerabilities that resemble the traditional vulnerabilities through primitive tampering and probing or a fresh category of vulnerabilities like adversarial AI. Figure 3 illustrates the effectiveness of AI and IoT as enablers of future 6G systems while focusing on the three main roles of AI: 1) operative, 2) defender, and 3) target. The main objective of this paper is to provide a comprehensive and in-depth review of threats and challenges faced by AI-based IoT systems and infrastructures. While doing so, the article focuses on the Machine Learning (ML) subset and its different learning paradigms, namely centralized, federated, and decentralized approaches. In addition, we discuss possible effective countermeasures that can be employed to protect these systems. The contributions of this study are summarized below:

- An overview of enabling emerging technologies for 6G–IoT intelligence.
- A detailed report on the datasets used by the scientific community for experimenting and evaluating edge learning on cyber attacks.
- Presentation of the threat model of attacks against machine learning; and classification into eight categories: backdoor attacks, adversarial examples, combined attacks, poisoning attacks, Sybil attacks, byzantine attacks, inference attacks, and drop attacks.
- A comprehensive taxonomy and a side-by-side comparison of the state-of-the-art defense methods against federated machine learning vulnerabilities.
- Presentation of the security and privacy challenges and opportunities for federated machine learning in 6G-

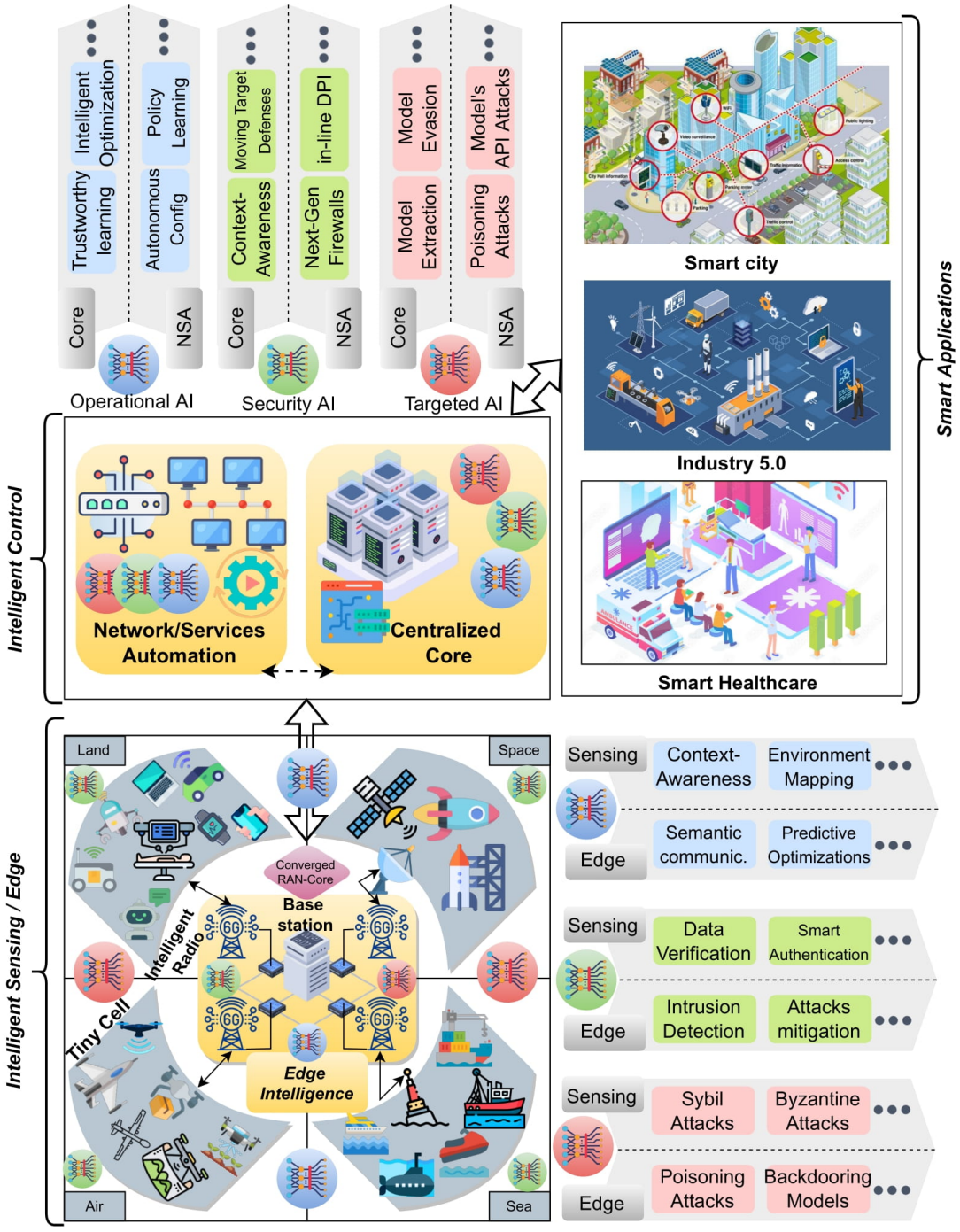


Fig. 3: AI as a native component of 6G concerning the operational, defensive, and targeted perspectives: 1) Intelligent Sensing/Edge: This comprises two primary components: the first is the data generation aspect, which includes devices, systems, and processes from which data originates; the second is the edge layer, where certain cloud processing tasks are executed at the network's periphery; 2) Intelligent Control: This pertains to smart network management, primarily at the network core (e.g., Core Network Data Analytics Function (CNDF) [6]); 3) Smart Applications: These encompass present and future intelligent applications that utilize the network; Additionally, we consider various AI perspectives within the network, such as operational, defensive, and targeted.

enabled IoTs.

The structure of this article is organized as shown in Figure 1. Section II provides the review methodology. Section III sheds light on state of the art in ML in 6G-enabled IoT security and ML-associated threats. Section IV presents an overview of enabling emerging technologies for 6G–IoT intelligence. Section V discusses the use of edge learning and its application in cybersecurity. Section VI provides a detailed report on the datasets used by the scientific community. Section VII reviews the threat models attacks against machine learning and provides a classification into eight categories. Section VIII provides a taxonomy and a side-by-side comparison of the state-of-the-art defense methods. Then, we discuss new research directions and future overall prospects for 6G-enabled IoTs in Section IX. Lastly, Section X presents concluding remarks.

II. Review methodology

In this section, we provide the review methodology that we employed to investigate the vulnerabilities, datasets, and defenses of edge learning for 6G-enabled IoT systems. We begin by outlining our objectives and research questions, followed by our search strategy, data extraction process, and quality assessment criteria.

A. Objectives and research questions

In this research study, we aim to investigate the vulnerabilities, datasets, and defenses of edge learning for 6G-enabled IoT systems. Our objective is to understand the security threats posed to these systems and the methods proposed by researchers to protect them. We will analyze the vulnerabilities of machine learning and advanced deep learning techniques used to differentiate between benign and malicious traffic patterns. Our research will also explore the growing interest in defense solutions and their potential effectiveness in securing IoT systems against machine learning large-scale attacks. Table II provides the particular identified research questions to address the goals outlined above.

B. Search strategy

To identify the literature for analysis in this paper, a keyword search was conducted using terms such as "Federated Learning for 6G-enabled IoT", "Edge Learning for 6G-enabled IoT", "Distributed Learning for 6G-enabled IoT", "Federated Learning for Cyber Security", and "ML Attacks" in academic databases such as SCOPUS, Web of Science, ACM Digital Library, IEEE, Wiley, and Springer. This search produced substantial results, but some relevant primary sources may have been missed during the process. Only proposed ML attacks and Defenses methods for 6G-enabled IoT were collected, and each source was evaluated based on criteria such as reputation, relevance, originality, publication date (between 2015 and 2023), and influence in the field. Reputation was evaluated based on the author's previous work and expertise in the field. The Reputation was evaluated based on the author's previous work and expertise in the field. The Originality was evaluated based on the novelty of the ideas presented in the source. The Influence was evaluated based on the impact of the source in the field. The Sources cited more frequently were considered to have a higher influence and were given a higher score. The evaluation criteria used in this study ensured that the sources selected for analysis were of high quality, relevant to the research topic, and contained innovative and influential ideas. After evaluation, the sources were ranked based on their overall score, and the top sources were selected for analysis.

C. Data extraction

The process of data extraction is a lengthy one that involves a thorough examination of selected research papers to identify and collect essential data. To ensure that the research papers align with our research questions, a quality assessment is conducted in the initial phase, resulting in fewer papers for further analysis. The remaining papers undergo data extraction to obtain pertinent information, and Table V outlines the extracted metadata.

D. Quality assessment

In our study, we conducted a quality assessment using a questionnaire that included ten questions related to the research questions and objectives of the study. These questions covered a range of topics, including 1) the clarity and appropriateness of the research questions, 2) the comprehensiveness of the literature review, 3) the appropriateness and relevance of the datasets used, 4) the effectiveness of the classification methods used, 5) the depth and comprehensiveness of the discussions on proposed solutions, 6) the degree to which the proposed solutions are supported by experimental analysis, and 7) the clarity of the presentation of the results. By answering these questions, we were able to evaluate the quality and identify the advantages and disadvantages of each work. The formulation of the quality assessment criteria in Table III is based on eight questions that assess the quality of our research in relation to the research questions.

III. An Overview of ML-Associated and 6G-enabled IoT Security

In the context of ML-Associated and 6G-enabled IoT Security, a threat refers to potential danger or harm to the security of the system, while a vulnerability refers to a weakness or flaw in the system that can be exploited by a threat actor [37]. Threats in ML-associated 6G-enabled IoT security inherit the threats associated with the previous generation communication networks including but not limited to data breaches, denial of service (DoS) attacks, unauthorized access to sensitive data, and tampering ML models [38]. On the other hand, vulnerabilities in ML-Associated and 6G-enabled IoT Security include outdated software, unsecured network connections, weak passwords, and insecure authentication mechanisms. These vulnerabilities can make it easier for a threat actor to carry out an attack on the system [39].

In this section, we present state-of-the-art reviews related to ML-related and 6G-enabled IoT security. We classify them

TABLE I: Related studies on edge learning for 6G-enabled IoT applications.

Reference	Year	ML for 6G–IoT security	Vision for 6G networks security	ML-associated threat (Centralized)	ML-associated threat (Federated)	ML-associated threat (Distributed)	Datasets	Defenses methods against ML vulnerabilities
Yang *et al.* [11]	2019	No	No	No	Partial	No	No	No
Hussain *et al.* [12]	2020	Yes	No	Yes	No	No	No	Partial
Mohanta *et al.* [13]	2020	Yes	No	Yes	No	No	No	Partial
Lyu *et al.* [14]	2020	No	No	Partial	Yes	Partial	No	Partial
Wahab *et al.* [15]	2021	No	No	No	Partial	No	No	Partial
Nguyen *et al.* [16]	2021	No	No	No	Partial	No	No	No
Alazab *et al.* [17]	2021	Partial	No	No	Partial	No	No	No
Zaman *et al.* [18]	2021	Yes	No	Yes	No	No	No	Partial
Mothukuri *et al.* [19]	2021	No	No	Partial	Yes	Partial	No	Partial
Ghimire and Rawat [20]	2022	No	No	No	Partial	No	No	No
Ma *et al.* [21]	2022	No	No	No	No	No	No	No
Liu *et al.* [22]	2022	No	No	No	No	No	No	No
Boobalan *et al.* [23]	2022	No	No	No	Partial	No	No	No
Yang *et al.* [24]	2022	Partial	No	No	No	No	No	No
Sarker *et al.* [25]	2022	Yes	No	Yes	No	No	No	Partial
Qian *et al.* [26]	2022	No	No	Partial	Partial	Yes	No	Partial
Ma *et al.* [27]	2022	No	No	Partial	Partial	Yes	No	Partial
Zhang *et al.* [28]	2022	No	No	No	No	No	No	No
Veith *et al.* [29]	2023	No	Partial	No	No	No	No	No
Mao *et al.* [30]	2023	Partial	Partial	No	Partial	No	No	Partial
Alotaibi *et al.* [31]	2023	Yes	Partial	Partial	Partial	No	No	Partial
Hua *et al.* [32]	2023	No	Partial	No	Partial	No	No	No
Al-Quraan *et al.* [33]	2023	No	No	No	No	No	No	No
Xia *et al.* [34]	2023	No	No	No	Partial	No	No	No
Issa *et al.* [35]	2023	No	No	No	No	No	No	No
Zhu *et al.* [36]	2023	No	No	No	No	No	No	No
Our study	2023	Yes	Yes	Yes	Yes	Yes	Yes	Yes

TABLE II: Research questions.

	Question	Objective	Section
RQ1	What are the current state-of-the-art reviews on ML-related and 6G-enabled IoT security, and how can they be classified into ML-based security for 6G-IoT systems and ML-associated threats?	To review and categorize the studies into two categories: ML-based security for 6G-IoT systems and ML-associated threats. The research can contribute to enhancing the security of 6G-IoT systems and identifying potential threats facing ML-associated paradigms.	Section III
RQ2	What are the potential benefits, challenges, and implications of using AI-based Network Intelligence (NI) as the backbone for network management in 6G and beyond, particularly in the context of the IoE ?	To provide insights into the architectural landscape, technological prospects, and security and privacy concerns associated with the use of NI in future sophisticated networks and applications.	Section IV
RQ3	What are the main characteristics and purposes of the datasets used in the scientific community to experiment and evaluate machine learning techniques for cyber attacks, and how can they be classified into different categories based on their content?	To provide a detailed report on the datasets used by the scientific community for experimenting and evaluating machine learning techniques on cyber attacks	Section VI
RQ4	What are the types of attacks and vulnerabilities that machine learning systems are susceptible to, particularly in 6G-IoT Networks, and how can they be classified based on the attacker's knowledge, the type of attack employed, and the final objective?	To identify and classify the most common types of attacks and vulnerabilities that machine learning systems face in 6G-IoT Networks, and to provide a comprehensive understanding of these threats based on the attacker's knowledge, the type of attack employed, and the final objective.	Section VII
RQ5	What are the current state-of-the-art methods for securing machine learning systems in 6G-IoT systems where AI is a key player?	To identify and analyze the current defense mechanisms against ML attacks and provide insights into the most effective ways to protect machine learning systems against potential security risks.	Section VIII
RQ6	What are the major challenges and open issues in enhancing cyber security in IoT and AI, and how can the scientific community ensure a fully secure cyber environment for future networks (i.e., 6G and beyond)?	To identify and explore the challenges associated with creating reliable and trustworthy learning environments for 6G-IoT intelligence, and to propose potential solutions to these challenges.	Section IX

TABLE III: Quality assessment questionnaire.

No.	Question	Description	Relevant to the research question
Q1	Were the research questions clearly stated and appropriate?	Evaluate the clarity and appropriateness of the research questions	RQ1, RQ2, RQ3, RQ4, RQ5, RQ6
Q2	Did the authors provide a comprehensive review of the state-of-the-art in the relevant areas?	Evaluate the comprehensiveness of the literature review	RQ1, RQ2, RQ4, RQ5, RQ6
Q3	Were the datasets used in the study appropriate and relevant?	Evaluate the suitability and significance of the datasets used in the study	RQ3
Q4	Were the methods used to classify the studies, datasets, attacks, and vulnerabilities appropriate and effective?	Review the appropriateness and effectiveness of the classification methods used	RQ1
Q5	Were the potential benefits, challenges, and implications of the proposed solutions discussed in detail?	Evaluate the depth and comprehensiveness of the proposed solutions	RQ1, RQ2, RQ4, RQ5, RQ6
Q6	Were the limitations of the proposed solutions discussed?	Evaluate the limits of the proposed solutions	RQ2, RQ6
Q7	Were the potential applications of the proposed solutions discussed in detail?	Review the thoroughness of the discussions on the potential applications of the proposed solutions	RQ2, RQ5
Q8	Were the conclusions and recommendations based on the results and analysis presented in the study?	Evaluate the degree to which the conclusions and recommendations are supported by the results and analysis	RQ1, RQ2, RQ3, RQ4, RQ5, RQ6

TABLE IV: Usage of Machine Learning Paradigms in 6G-enabled IoT Applications.

Paradigm	Characteristics	Usage in 6G-enabled IoT
Supervised Learning	- Models are trained on labeled data - The goal is to make predictions on new, unseen data	Build predictive models that can detect anomalies in large datasets, such as predicting device failures or detecting anomalies in IoT network traffic
Unsupervised Learning	- Models are trained on unlabeled data. - The goal is to identify patterns or structures within the data	Identifying patterns and insights from unstructured data in 6G-enabled IoT systems, such as clustering devices based on their behavior or detecting anomalies in sensor data
Reinforcement Learning	- Models take actions based on feedback from the environment. - The goal is to learn the optimal policy that maximizes reward over time	Optimizing the performance of devices and networks in 6G-enabled IoT, such as optimizing the energy consumption of IoT devices by learning from the device's environment and adjusting its behavior accordingly
Semi-Supervised Learning	- Models are trained on a combination of labeled and unlabeled data - The goal is to improve performance by leveraging the unlabeled data	Improving the performance of models in 6G-enabled IoT by leveraging both labeled and unlabeled data, such as classifying devices in the network by using both labeled and unlabeled data
Transfer Learning	- Models use knowledge learned from one task to improve performance on another task. - The goal is to reduce the amount of data needed to train the model	Improving the performance of models in 6G-enabled IoT by leveraging knowledge learned from one task to improve performance on another task, such as improving the accuracy of intrusion detection in IoT devices by using pre-trained models on similar datasets

into two categories: 1) ML-based security for 6G-IoT systems, which incorporate studies on IoT security and the vision of future 6G networks, and 2) ML-associated threats, which provide an overview of works on threats facing ML-associated paradigms, namely centralized, FL and distributed learning. In Table I, we provide a detailed comparison between our work and state-of-the-art studies.

A. Machine Learning For 6G–IoT security

ML is considered a key tool for robust security, especially for anomaly classification tasks. In this part, we focus on IoT and 6G from the perspective of ML-based security.

1) Usage of ML Paradigms in 6G-enabled IoT Applications: The table IV provides an overview of how the different machine learning paradigms can be used in 6G-enabled IoT applications. The different machine learning paradigms can be categorized into five paradigms, including, Supervised Learning, Unsupervised Learning, Reinforcement Learning, Semi-Supervised Learning, and Transfer Learning [39]–[41]. Supervised learning is the most commonly used paradigm, where the model is trained on labeled data and learns to map inputs to outputs based on example pairs of input-output data. On the other hand, unsupervised learning involves training the model on unlabeled data and finding patterns and relationships in the data without explicit guidance. Semi-supervised learning combines both supervised and unsupervised learning, and the model is trained on both labeled and unlabeled data to improve its performance. Reinforcement learning involves training the model to make decisions based on feedback from the environment and taking actions that maximize reward over time.

2) IoT-related security: Hussain *et al.* [12] explored the ways in which ML and Deep Learning (DL) have impacted the loT ecosystem from a security and privacy perspective. The survey first presents the background on the security and privacy concerns and challenges facing the IoT, including those obstacles related to the resource constraints associated with IoT devices, as well as the attack vectors and security expectations. This is followed by highlighting various ML and DL mechanisms and their applicability to IoT security in various scopes of applications such as forensic face recognition, cryptographic security character identification, and malicious code detection. In addition, the survey highlights the shortcomings that may be confronted by the adoption of ML techniques in IoT, including resource limitations, where IoT devices may not be suitable to support or execute sophisticated computational processing. Among other recommendations, the authors advocate that in order to overcome some of the limitations of ML approaches to IoT security, both DL and Deep Reinforcement Learning (DRL) theoretical frameworks should be further enhanced to allow adequate quantification of performances based on metrics such as computing complexity.

In the same context, Zaman *et al.* [18] provide a study regarding IoT security threats by layer, as well as security schemes to address them. The layers involved are perception, network, transport, processing, and application, with a focus on different IoT protocols related to each layer of the ecosystem. In response to the layer-based threats, rule-based (such as fuzzy logic) and AI-based layer response actions were presented, including ML (such as SVM) and DL (such as DNN and KNN) based systems, as well as performance evaluations of these AI-based layer-wise response actions.

In addition, Mohanta *et al.* [13] suggest the use of emerging technologies such as AI, ML, and blockchain technologies to address existing security and privacy issues in IoT applications, such as jamming, DoS, and malicious nodes identification. The authors begin by highlighting layer-wise security issues in the IoT system and then answer the question of "how can these technologies be used to mitigate security threats in IoT?" in an overview. A study by Sarker *et al.* [25] offered a holistic view of IoT security intelligence, which is driven by ML and DL techniques that mine information out of raw data to smartly safeguard IoT systems against a range of sophisticated cyberattacks, including booting, sinkhole, cloud malware, access control attacks. Based on their study, the authors outline the corresponding future research directions and associated problems within the scope of their study. Xiao *et al.* [42] examined various cyberattacks against IoT environments, such as eavesdropping, spoofing, and jamming. In addition, the authors identify several IoT-specific security techniques based

TABLE V: Data extraction relevant fields.

Field	Description
Reference	Provides the title of the research paper and its citation details
Year	Year of publication
Dataset	Provides the datasets used as a benchmark experiment in the evaluation of ML vulnerabilities
Attack category	The different types of attacks against ML techniques, such as Sybil attack, Backdoor attack, Poisoning attack, and Adversarial attack, are classified under attack categories
Attack type	This describes various types of attacks against ML techniques, such as miscellaneous attacks, RNN backdoor attacks, Sybil-based poisoning attacks, Federated poisoning attacks, imperceptible backdoor patterns, Clean-label poisoning attacks, Poisonous label attacks, and more
Machine learning	Provides the machine learning techniques used in the edge learning
Learning mode	This describes various types of learning modes, including centralized learning, distributed learning, federated learning, and transfer learning
Targeted ML	Provides the machine learning techniques that are affected by attacks and vulnerabilities
ML phases	This describes various types of ML model development lifecycles, including, Post-deployment and pre-deployment
Attack Goal	Provides the goal of an attack against a machine learning system, including Data Poisoning, Adversarial Examples, Model Stealing, Model Evasion, ...etc.
Attack Description	Describes an attack against a machine learning system
The attacker's knowledge	Provides the attacker's knowledge (e.g., the trained parameters, the learning algorithm, the feature values, and the training set) that can be used to launch sophisticated attacks, evade detection by traditional security systems, and create targeted attacks
Attack mode	This describes various types of attack modes, including, centralized learning, distributed learning, federated learning
The attacker's goal - Vulnerabilities	Provides the goal of an attacker targeting machine learning systems that can vary depending on the specific context and motivations of the attacker
Mitigation solution	Provides the mitigation solutions that can be implemented to address machine learning vulnerabilities
Attacks examples	Provides the threat models of adversarial examples generation designed to cause a machine learning model to make incorrect predictions or decisions
Attacks for the specific tasks	Provides the different types of machine learning tasks (e.g., Classification, Recognition, Image Segmentation, ... etc.)
White/black/Grey box	The techniques used to test and exploit the vulnerability of ML models in different levels of access and knowledge about the model's internal workings
Adversarial Example Generation	Describes the adversarial example generation
Defense framework	Provides the defense frameworks in mitigating machine learning vulnerabilities and safeguarding the integrity and security of AI-powered systems
Threat model	The threat model of machine learning vulnerabilities posed by adversarial attacks, data poisoning, model inversion, and other malicious activities that exploit weaknesses in the machine learning pipeline
Classifiers	
Pros (+) Open Issues (-)	Provides the Pros (+) and Open Issues (-) of the defense framework against ML vulnerabilities
Defense methods	This includes the classification of defense methods against edge learning vulnerabilities (e.g., Training phase defense methods, Post-training phase defense methods, Inference phase defense methods)
Defense mechanisms	This includes the classification of defense mechanisms used in defense Defense mechanisms against edge learning vulnerabilities (e.g., Privacy leakage defense mechanisms, Sybil attacks defense mechanisms, ... etc.)
Defense strategy	Provides the defense strategy adopted by defense mechanisms against edge learning vulnerabilities (e.g., Bio-inspired, Reputation-Awareness, Federated Filters,... etc.)

on learning, such as malware identification, access control and secure offloading. The study discusses the challenges related to state-of-the-art machine learning-based protection techniques, such as computing and communication overhead and partial state observation. Tahsien *et al.* [43] presented a discussion about the layered IoT architecture. The importance of IoT security in terms of possible attacks under different types, such as physical and cyber attacks, attack surfaces including device perception and cloud applications, and the effects of such attacks including accessibility, integrity, and authorization are discussed in detail. In addition, prospective ML-based contributions to IoT security based on different ML and DL algorithms are presented.

3) Vision For 6G networks: In a prospective overview that outlines the principles of a 6G system, Saad *et al.* [4] position 6G as a fundamental transformation of Self-Organizing Networks (SONs), whereby the network only adjusts its operations to particular environmental states, into a Self-Sustaining Network (SSN) capable of sustaining its Key Performance Indicators (KPIs) across the extremely complex and highly dynamic operating environments arising from the rich application landscape of 6G. The authors elaborate that AI and specifically Reinforcement Learning (RL) address the goal of building SSNs that can independently sustain high KPIs and handle network resources, functionality, and oversight. Furthermore, in the same study, it is anticipated that the AI-based 6G features to be joined by a collaborative network intelligence located at the edge, resulting in a 6G system that can accommodate future services such as Massive URLLC, and may even be capable of replacing classical network frame structures.

Although the main constituents of the 6G architecture remain undefined and yet to be standardized, some aspects can be foreseen and their associated threats are being discussed. For example, Siriwardhana *et al.* [5] provided a future-oriented view of the immense role of AI in 6G network security, pinpointing upcoming research directions through its discussion of the AI-based security and privacy challenges, along with some proposed possible solutions. The authors broadly categorize the 6G network threat landscape into two categories, namely, architectural and technological threats. The first entails attacks on the infrastructural level such as attacks on the User Plane Micro Services (UPMS), and the Control Plane Micro Services (CPMS), while the second entails attacks on embedded technologies such as attacks on ML, blockchain, cryptographic protocols, and Visible Light Communication (VLC).

Additionally, the authors suggest several AI-based solutions that can address such threats, including edge-based FL for securing networks under massive data and device conditions. The limitations and threats of AI, including data injection/manipulation and logical corruption, were also discussed. In the same context, Nguyen *et al.* [44] discuss potential security and privacy issues around multiple levels of 6G, namely the physical, connection, and service levels. In addition to those vulnerabilities by inheriting from earlier communication technologies, 6G introduces additional threat engines from emerging radio technologies and attacks against pervasive

intelligence. Predictions on protective measures against such threats are provided. These include AI-based security, Differential Privacy (DP), blockchain, real-time adaptive security, deep network slicing, and quantum-based cryptography.

B. Machine Learning Associated Threats

Although ML-based solutions can provide reasonable defense capabilities, they are prone to various adversarial attacks. Motivated by the facts above, we present different studies on ML threats with respect to three learning paradigms, as presented below:

1) Centralized Edge Learning: Centralized learning is considered the first learning paradigm for ML, where data is gathered at one location for training. Despite its advantages, This approach also involves disadvantages and associated threats. For instance, Liu *et al.* [45] review the security-related threats to ML and provide a methodical study on such threats in two dimensions: 1) training stage and 2) test/inference stage. Then, the authors classify existing protective techniques used for ML into four classes, including, assessment techniques, training phase precautions, inference-test stage defenses, and privacy and data protection precautions. In another study, Xue *et al.* [46] present threat models where ML classifiers are targeted by adversaries. The analysis of the reasons behind the possibility of being attacked is presented. Therefore, security concerns are categorized under five categories, including, poisoning of the training set, backdooring through the training set, adversarial attacks, model theft, and inference attacks. In addition, a number of suggestions on ML-related security evaluations are also provided.

Hu *et al.* [47] present the entire life-cycle of an AI-based operating environment as a roadmap to outline potential security-related threats occurring at each phase, and subsequently elaborate on the corresponding countermeasures that can be taken. Oseni *et al.* [8] and Liu *et al.* [48] focus on investigating adversarial attacks against AI-based systems, taking into account areas such as available methods for the generation of adversary samples. Oseni *et al.* also expands on the mathematical engines of AI, particularly the emerging variants of reinforcement and federated learning in order to illustrate how vulnerabilities in AI models are exploited. In addition, the study considers several cyber defenses to help prevent AI systems against these types of threats, including, data sanitization, robust statistics, defensive distillation, and gradient masking.

Kuzlu *et al.* [9] provides a study in which concepts around IoT security are presented, with a focus on attacks using and targeting AI. The authors provide a classification of AI attacks into three categories, namely 1) vulnerability scanning automation, which includes fuzzing and symbolic execution, 2) input attacks, and 3) data poisoning and fake data insertion, involving datasets and algorithm poisoning. Hao and Tao [49] review existing adversary evasion and poisoning attacks in smart grids. The types of adversarial examples are classified based on the aspect of the threat, including the attacker's influence, knowledge, specificity, computation and approach, and security breach. Multiple attack types per category are also provided, such as white box/black box attacks, targeted/non-targeted attacks, and specific approaches used such as gradient, decision, and transfer attacks. The authors propose six approaches for effective checks to mitigate adversarial examples on image classification, consisting of the following: gradient hiding, adjoint detection models, statistical methods, preprocessing methods, the ensemble of classifiers, and proximity metrics.

Chen *et al.* [50] discussed the critical infrastructure frameworks monitored by the IoT and the associated security vulnerabilities with the main focus on Advanced persistent threat (APT) attack patterns. Specifically, the authors examine a variety of cutting-edge AI-based approaches to discover and successfully mitigate attacks on such networks. Among them, 14 AI-based approaches are selected according to their application frequency. Al-Rubaie *et al.* [51] focused on different threats to ML privacy such as reconstruction, deanonymization, and membership inference attacks. For privacy preservation, the authors consider multiple cryptographic techniques as a defense barrier to mitigate these attacks and preserve ML privacy in its different stages (i.e., data preparation, learning, and inference), including DP, homomorphic encryption (HE), garbled circuits, and secure processors.

2) Federated Edge Learning: The introduction of FL provided a privacy-preserving paradigm for ML training. Although FL can ensure privacy preservation to some extent, there are FL-specific threats and the maintained privacy is only offered under certain circumstances. Mothukuri *et al.* [19] review specific privacy and security issues in federated learning that need to be addressed. The outcome of their investigation suggests that, overall, there are less significant specific privacy threats associated with federated learning than there are security threats. The authors state the following highest priority security threats: communication blockages, poisoning, and backdoor attacks, whereas inference attacks pose one of the greatest threats to FL privacy. Liu *et al.* [52] reviewed current threats and defense mechanisms in the FL domain across all stages of FL, namely data and behavioral auditing, training, and inference. For each stage, the authors discussed potential threats, related attacks, and available defenses. Lyu *et al.* [14] review existing privacy threats and attacks in all categories of FL, namely Horizontal Federated Learning (HFL), Vertical Federated Learning (VFL), and Transfer Federated Learning (TFL). The authors also describe the privacy leakage issues in FL by pointing out the fact that these issues can originate from the aggregator or from individual participants. It is important to note that threats can be associated with both insiders and outsiders, as well as with malicious or semi-honest nodes. The attacks on FL are classified into two broad categories: 1) poisoning attacks and 2) inference attacks. Yang *et al.* [11] present an overview of federated learning, a novel approach addressing challenges such as data privacy and security in AI development. The authors offer definitions, classifications, and potential applications of a secure federated learning framework, further discussing its successful application across different business domains. The authors advocate for a paradigm shift in AI, redirecting the focus from enhancing model performance (the predominant

current focus) to exploring compliant data integration methods, thereby aligning with data privacy and security laws. Wahab *et al.* [15] provide a tutorial on Federated Learning and its related concepts, technologies, and learning approaches. The authors then categorize the literature into high-level challenges addressed by Federated Learning and further divide these into low-level challenges. This three-tier classification allows a deeper understanding of the topic and the methods employed to tackle particular problems. The paper also offers a set of desirable criteria and future research directions for each category of high-level challenges, with the aim to assist the research community in designing innovative and efficient solutions.

Implementing blockchain as a ledger technology can help decentralize FL training without requiring a central server, improving security and scalability. This combination of FL and blockchain has led to the creation of a new paradigm called FLchain. FLchain potentially transforms Mobile-edge computing (MEC) networks into decentralized, secure, and privacy-enhancing systems. Zhu *et al.* [36] identify key issues in FL that blockchain can address and categorizes existing system models into decoupled, coupled, and overlapped classes based on federated learning and blockchain integration. It compares the benefits and downsides of these models and investigates potential solutions to their limitations. Issa *et al.* [35] proposes the use of blockchain technology and smart contracts to secure FL in IoT systems. It reviews blockchain-based FL methods and discusses current IoT security issues. The paper also covers IoT data analytics from a security perspective, as well as the challenges and risks of integrating blockchain and FL in IoT. Open research questions are addressed, and a thorough literature review of blockchain-based FL approaches for IoT applications is provided. Nguyen *et al.* [16] provides an overview of FLchain's fundamental concepts and explores its opportunities within MEC networks. Several challenges related to FLchain design are also identified, including communication cost, resource allocation, incentive mechanism, security, and privacy protection. Potential applications of FLchain in popular MEC domains such as edge data sharing, edge content caching, and edge crowdsensing are discussed. Therefore, Alazab *et al.* [17] present a review of various FL models developed to enhance authentication, privacy, trust management, and attack detection. The article also explores real-time use cases that have recently employed FL to preserve data privacy and enhance system performance. Ghimire and Rawat [20] survey the application of FL, a privacy-aware machine learning model, in enhancing the security of IoT systems. They compare centralized learning, on-site distributed learning, and FL, focusing primarily on the security aspects. The discussion also covers performance issues such as accuracy, latency, and resource constraints that might affect the overall functionality of IoT. In addition to evaluating current research efforts, challenges, and trends in the field, the authors consider future developments in this paradigm. The article provides readers with an in-depth understanding of FL's role in cybersecurity, outlining different security attacks and countermeasures.

Despite the potential of FL, non-IID (non-independent and identically distributed) data present on individual devices participating in the FL process pose significant issues. This statistical heterogeneity can hamper the model's performance and discourage user participation in FL. Ma *et al.* [21] discusses the challenges posed by non-IID data in the context of FL. The authors provide a review of the state-of-the-art solutions for non-IID problems, aiming to fill a gap in the literature and facilitate further implementation of FL. Boobalan *et al.* [23] propose an overview of the combination of FL and IIoT, addressing data privacy and on-device learning motivations, potential usage of machine learning, deep learning, and blockchain techniques for secure IIoT. It also explores the management of large and diverse data sets and discusses applications in industries like automotive, robotics, agriculture, energy, and healthcare. The upcoming deployment of billions of IoT devices, facilitated by faster Internet speeds from 5G/6G, will produce a massive amount of data, including potentially sensitive user information. This surge in data will escalate communication and storage costs and intensify privacy concerns within traditional, centralized cloud-learning systems for IoT platforms. By eliminating the need for data centralization, FL reduces costs and enhances user-level privacy. However, implementing FL in IoT networks is not without its challenges. Zhang *et al.* [28] delves into the opportunities and hurdles of integrating FL in IoT platforms and its potential to enable a variety of IoT applications. It specifically identifies and explores seven major challenges of using FL in IoT platforms and underlines some recent promising strategies for overcoming them. In a study by Jere *et al.* [53], the authors point out client-side attacks as the most significant threats to FL platforms since malicious FL clients are capable of tampering with and repositioning the model edges while developing it. In addition, the authors discuss different critical FL attacks, including Generative Adversarial Network (GAN) reconstruction, data poisoning, membership inference, and model inversion attacks. Some of the defense mechanisms reported in the paper include DP, robust aggregation, and outlier detection. The details of Federated Edge Learning are presented in Algorithm 1.

3) Edge learning: This learning paradigm incorporates learning from multiple sources. Chen *et al.* [54] discussed a range of possible avenues for deploying a variety of distributed learning approaches on real-world wireless edge networks. The outlined approaches include FL, multi-agent RL, Federated Distillation (FD), and distributed inference. The authors also highlight both the benefits as well as potential security and privacy issues these approaches may encounter, such as gradient leakage in FL. In an attempt to answer the question of whether distributed learning is appropriate for wireless communications, Qian *et al.* [26] survey the state-of-the-art research on distributed learning for wireless communication, as well as the application cases, framework, algorithms, and other suitable alternatives for distributed learning. The research focuses on three layers: 1) physical, 2) medium access control (MAC), and 3) network layer. Furthermore, other emerging areas such as tensor and blockchain technologies are explored by reporting that these systems are prone to security attacks.

Ma *et al.* [27] provide an information-exchange level security and privacy risk classification of distributed ML, which

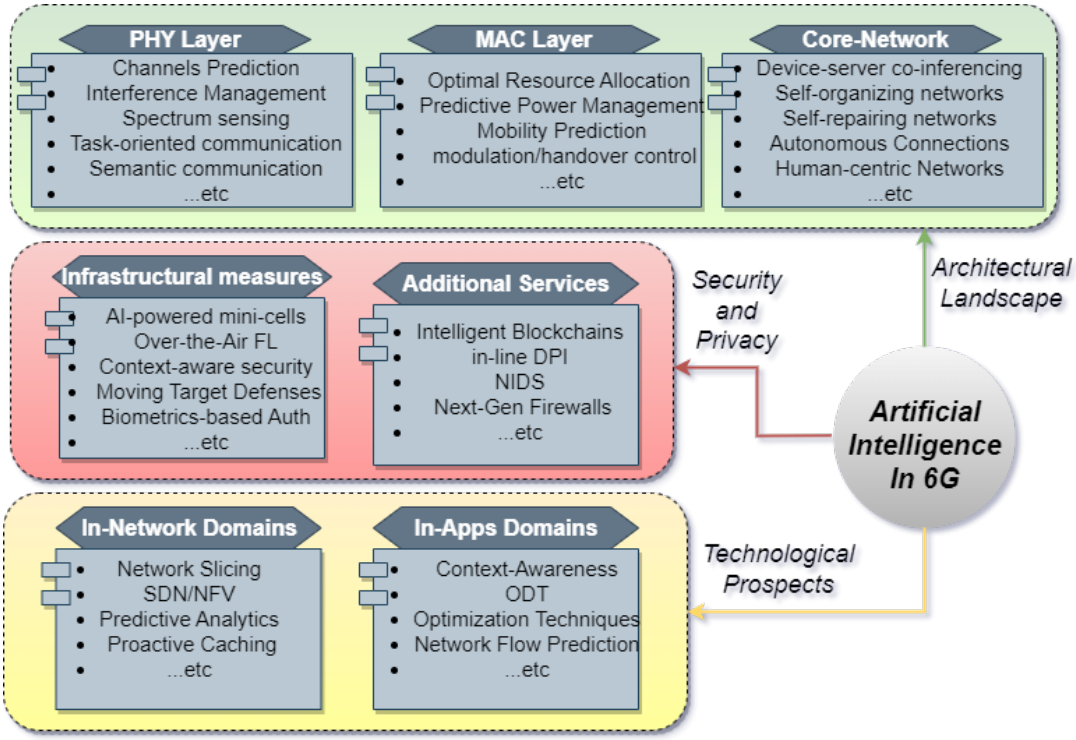


Fig. 4: Predicted AI involvement in future 6G networks: 1) Architectural landscape: AI as an enabler of native intelligent functionality. 2) Security and privacy: AI as an embedded/additive defender; 3) Technology Prospects: AI as an enabling intelligence service for high-level layers.

is organized according to the core phases of an ML workflow, namely preprocessing, learning, knowledge extraction, and result intermediation. The authors investigate and discuss possible risks involved in each level through an overview of current attack techniques, such as model poisoning/inversion attacks, inference attacks, label leakage, and data reconstruction attacks. The details of Distributed Edge Learning are presented in the Algorithm 2. In the case where a given client is involved in multiple FL sessions, it can communicate its real-time global status which is an array of every FL session status. The global status can be formalized as $GS_{C_{ID}} = [S_0 : Status, .., S_n : Status]$, where C_{ID} presents the client ID, and S_{ID} presents the FL session ID. The notations used in Algorithms 1 and 2 are presented in Table VI.

TABLE VI: Notations used in Algorithms 1 and 2

Notation	Description
η	Learning rate
Epo	Number of local epochs
$Batch$	Local minibatch size
K	Total clients number
$State$	State of the client
x	Single sample
C	Fraction
R	Global rounds
S_t	A subset of selected clients
n	The number of local samples
f	Model
$\mathcal{P}$	Pre-processed dataset
$f_c(.)$	Loss function
$InitializeModel(.)$	Random model initialization
$ConnectedClients(.)$	Number of active clients required
$RequestUpdate()$	A function used to request the model updates
$ReceiveUpdate()$	A function used to receive the model updates
$AggregateModels()$	A function used to aggregate the model updates
$UpdateModel()$	A function used to updates the model after the aggregation

C. Quantitative Comparison of Edge Learning in 4G, 5G, and 6G-enabled IoT Environments

Edge learning is a subset of edge computing, which aims to perform machine learning algorithms on data generated at the network's edge, i.e., closer to the source of data generation. Figure 5 illustrate the network architecture and communication processes for centralized edge learning-6G-enabled IoT and federated edge learning-6G-enabled IoT. Centralized edge learning and federated edge learning are two approaches to training machine learning models in 6G-enabled IoT, which differ in how they handle data. In centralized edge learning, all the data used to train the model is collected and stored in a central location, such as an edge server or cloud-based platform. The model is then trained based on this centralized dataset, which may be relatively large and diversified. On the other hand, federated edge learning refers to training machine learning models on distributed datasets across many devices. This approach can be particularly useful when data privacy is a concern, as the data remains on the device and is not transmitted to an edge server. However, both approaches have advantages and disadvantages, and the suitable approach will

TABLE VII: Difference between Edge Learning in 4G IoT, Edge Learning in 5G IoT, and Edge Learning in 6G IoT.

Feature	**Edge Learning in 4G IoT**	**Edge Learning in 5G IoT**	**Edge Learning in 6G IoT**
Bandwidth	10 Mbps	1 Gbps	10 Gbps
Latency	100 ms	1 ms	100 µs
Data Rate	100 MBps	10 GBps	100 GBps
Energy Efficiency	Medium	High	Very High
Network Density	1000-10,000 devices per square km	100,000-1,000,000 devices per square km	10 million+ devices per square km
Network Connectivity	Cellular network and Wi-Fi hotspots	Cellular network and Wi-Fi hotspots, with an increased focus on dense urban areas	Integrated satellite, airborne, and terrestrial networks
Spectrum Efficiency	2-3 bps/Hz	10-20 bps/Hz	100 bps/Hz
Network Architecture	Centralized	Decentralized	Autonomous
Applications	Streaming, Video Conferencing, Gaming	IoT, Autonomous Vehicles, Smart Cities	IoT, Holographic Communications, Teleportation, Brain-Machine Interface
Interoperability	Limited	Improved	Full Interoperability
Edge Computing Capability	Limited (Basic analytics, traditional ML)	Moderate (Advanced analytics, deep learning)	Advanced (Distributed computing, edge AI)
AI Integration	Limited	Advanced	Full AI Integration
Virtual Reality Integration	Limited	Advanced	Full Virtual Reality Integration
Autonomous Devices	Limited	Advanced	Full Autonomous
Coverage	Limited (Urban areas)	Wide (Urban and rural areas)	Very wide
Reliability	Medium (Packet loss, interference)	High (Low packet loss, low interference)	Ultra-high (No packet loss, no interference)
Semantic Communications	Limited, with communication primarily based on IP addresses and port numbers	Increased support for semantic communication through enhanced edge computing and network slicing	Advanced support for semantic communication, including the integration of augmented reality and virtual reality
Holographic MIMO Surfaces	Not supported	Limited support for holographic MIMO	Full integration of holographic MIMO for enhanced communication and data transmission
Millimeter-Wave and Terahertz Bands	Not supported or limited support	Increased support for millimeter-wave and terahertz bands	Full integration of millimeter-wave and terahertz bands for enhanced communication and data transmission
Network Slicing	Not supported	Enhanced support for network slicing to provide customized services to different types of users	Full integration of network slicing for customized communication and data transmission services
Physical Layer	Orthogonal frequency division multiplexing (OFDM)	Orthogonal frequency division multiplexing (OFDM) with additional support for millimeter-wave and terahertz bands	Next-generation modulation and multiplexing techniques for enhanced data transmission and communication
Massive MIMO	Not Available	Available	Available
uMUB	Not Available	Available	Available
uHSLLC	Not Available	Available	Available
Near Space Communications	Not supported	Not supported	Full integration for near space communication and data transmission
mMTC	Available	Available	Available
uHDD	Not Available	Not Available	Available
Key Technologies	LTE, WiMAX, VoLTE, MIMO	mmWave, Sub-6 GHz, Massive MIMO, Network Slicing, Edge Computing, IoT-optimized networks	Terahertz frequencies, quantum cryptography, AI-driven networks, holographic communication, 10x faster speeds
Use Cases	Smart homes, wearables, fleet management	Smart factories, autonomous vehicles, e-health	Smart cities, augmented reality, digital twins

Massive MIMO: Multiple Input Multiple Output, uMUB - Unlicensed Multiuser Beamforming, uHSLLC - Unlicensed High-Speed Low Latency Communication, mMTC - Massive Machine Type Communication, uHDD - Unlicensed High-Definition Driving Display.

depend on the specific needs and constraints of the current machine learning task for 6G-enabled IoTs [55], [56].

With the advent of IoT devices, there is an ever-increasing demand for real-time data processing and analysis, which is fueling the need for edge learning [57]. Table VII compares the features of Edge Learning in 4G IoT, 5G IoT, and 6G IoT. We can observe a significant improvement in various aspects of each generation of technology. Moving from 4G to 6G, we observe significant improvements in latency, bandwidth, the number of devices supported, computational power, energy consumption, and cost per device. These improvements are expected to profoundly impact the applications of edge learning [29], [30], [33]. For instance, the bandwidth, data rate, energy efficiency, and network density in 6G IoT are significantly higher compared to 4G IoT. Moreover, the latency in 6G IoT has improved to 100 microseconds compared to 100 milliseconds in 4G IoT. The level of AI and Virtual Reality integration in 6G IoT is also more advanced compared to 4G IoT. Furthermore, the level of autonomous devices has improved greatly in 6G IoT. Despite this, the availability of Massive MIMO remains constant throughout the three generations of the technology. However, other technologies

Algorithm 1 Federated Edge Learning

Data: η , Epo, $Batch$, K, k.
Edge Server EdgeFedLearn (K, C, R):
  $f_1 \leftarrow InitializeModel()$
  for $t = 1, .., R$ **do**
    $S_t \leftarrow$ Subset(max$(C \cdot K, 1)$, "$random$")
    Parallel.for $k \in S_t$ **do**
      $f_{t+1}^k \leftarrow ClientUpdate(f_t, k)$
    end
    $f_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} f_{t+1}^k$
  end
  Broadcast f_{t+1} the updated model to clients
IoT device ClientUpdate (f, k):
1 $\mathcal{B} \leftarrow$ Split$(\mathcal{P}, Batch)$
2 **for** i = 1,..,Epo **do**
    for $b \in \mathcal{B}$ **do**
      $f \leftarrow f - \eta \nabla f_c(x, b)$
    end
  end
  Return f to Edge Server

Algorithm 2 Distributed Edge Learning

Data: η, Epo, $Batch$, K , k, $State$
$f_1 \leftarrow InitializeModel()$
Broadcast f_1 the initial model to clients
while $K \neq$ length(ConnectedClients()) **do**
  continue
end
$RequestUpdate()$
$ReceiveUpdate()$
$AggregateModels()$ ▷ State =1
$UpdateModel()$ ▷ State =2
$State = 1$ EdgeFedLearn (K, C, R):
  for $t = 1, .., R$ **do**
    $S_t \leftarrow$ Subset(max$(C \cdot K, 1)$, "$random$")
    Parallel.for $k \in S_t$ **do**
      $f_{t+1}^k \leftarrow ClientUpdate(f_t, k)$
    end
    $f_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} f_{t+1}^k$
  end
  Broadcast f_{t+1} the updated model to clients
$State = 2$ ClientUpdate (f, k):
1 $\mathcal{B} \leftarrow$ Split$(\mathcal{P}, Batch)$
2 **for** i = 1,..,Epo **do**
    for $b \in \mathcal{B}$ **do**
      $f \leftarrow f - \eta \nabla f_c(x, b)$
    end
  end
  Return f to Client

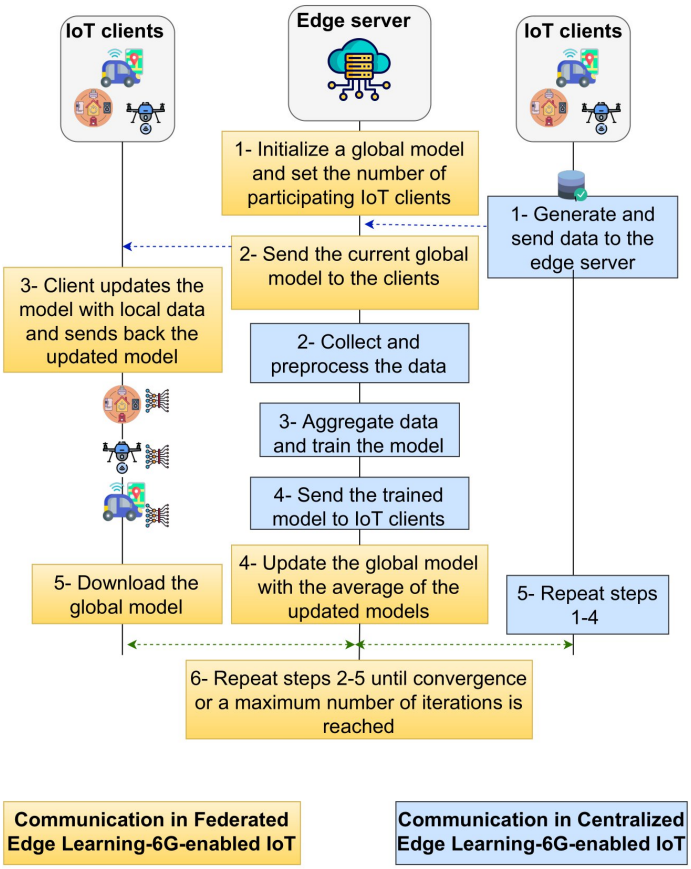


Fig. 5: Network communication processes for Centralized Edge Learning-6G-enabled IoT and Federated Edge Learning-6G-enabled IoT.

like uMUB and uHDD are only available in 5G IoT and 6G IoT. The level of interoperability has also improved greatly in 5G IoT and 6G IoT.

In 4G-enabled IoT, the latency ranges from 50-1000 ms, which may not be suitable for applications that require real-time data processing. The bandwidth is limited to 1-100 Mbps, which may not be sufficient for handling large datasets. The number of devices supported is relatively small, which limits the scalability of the system [31]. The computational power is also limited to less than 1 GFLOPS, which may not be enough for running complex machine learning algorithms.

In 5G-enabled IoT, the latency is significantly reduced to 1-10 ms, making it suitable for real-time applications. The bandwidth is also improved to 10-1000 Mbps, enabling the handling of larger datasets [58]. The number of devices supported is increased to 10,000-1M, improving the system's scalability. The computational power is also increased to 1-10 GFLOPS, making it possible to run complex machine-learning algorithms [59].

In 6G-enabled IoT, the latency is reduced even further to less than 1 ms, enabling ultra-low latency applications. The bandwidth is also significantly increased to more than 1000 Mbps, enabling the handling of massive datasets [60]. The number of devices supported is increased to more than 1M, making it possible to handle large-scale IoT deployments. The

computational power is also significantly increased to more than 10 GFLOPS, enabling the processing of complex machine learning algorithms. The energy consumption and cost per device also decrease significantly as we move from 4G to 6G, making it possible to deploy edge learning in resource-constrained environments [61].

D. Practical Deployments of AI in Edge/Fog

Although edge computing and fog computing are frequently used as synonyms, they are not identical concepts. Both emphasize the distribution of computing resources near the data's origin and usage; however, their methods differ. Edge computing involves processing data at or near the point of generation, like IoT devices, sensors, or other endpoints. In contrast, fog computing is a broader notion that expands the cloud computing paradigm to the network's periphery. This involves distributing computing, storage, and networking resources across multiple layers, from the edge devices to the cloud [62]. Effectively managing IT assets in cloud and fog/edge environments is a challenging task that necessitates a methodical decision-making process. The scarcity and heterogeneity of resources, in conjunction with the dynamic and diverse workloads, along with the unpredictable nature of advanced IT environments, have compounded the complexities of managing resources in such a landscape. It is crucial to observe the infrastructure's behavior since comprehending the workload's behavior can mitigate the intricacy of the challenge and improve the outcome of a particular implementation [63]. To overcome these challenges, adopting AI/ML-based solutions has gained traction, leveraging their ability to make sequential decisions and achieve optimal outcomes in this complex and ever-changing environment [64]. For example, in anticipating workload patterns or spatiotemporal impacts ahead of time for assisting in orchestrating resources [65].

AI-enabled fog and edge computing are being deployed in various industries, including healthcare, transportation, manufacturing, and retail, among others. Edge AI technology has been developed and commercialized by various companies, and there is a wide range of industrial products that are currently available or in development. Such implementations include edge AI chips, where companies such as NVIDIA [66], Intel, and Qualcomm have developed specialized chips designed to run AI algorithms on edge devices. These chips are used in a variety of applications, including autonomous vehicles, drones, and smart cameras. Also, Edge AI software platforms, where several companies, such as AWS IoT Greengrass, Google Cloud IoT Edge [67], and Microsoft Azure IoT Edge, have developed software platforms that enable developers to deploy AI models on edge devices. These platforms provide tools for building, training, and deploying machine learning models. In addition, edge AI cameras with built-in AI capabilities are becoming increasingly popular in industrial and commercial settings [68]. These cameras are used for applications such as facial recognition, object detection, and anomaly detection. Also, edge AI sensors, where IoT sensors with built-in AI capabilities are also becoming more common. These sensors are used in applications such as predictive maintenance, asset tracking, and environmental monitoring. Furthermore, Edge AI robots, where robots with onboard AI capabilities are being used in a variety of industrial applications, including warehouse automation, manufacturing, and agriculture. In terms of business models, AI-enabled fog and edge computing are being deployed through various models, including software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS). For instance, in the healthcare industry, companies such as Philips offer edge computing as a service for remote patient monitoring [69]. Similarly, in the manufacturing industry, companies such as GE are offering predictive maintenance as a service through their Predix platform [70].

Current research has shown that AI-enabled fog/edge computing can significantly reduce latency and energy consumption in various applications. In a recent paper by Hua *et al.* [32], the authors examined the mutually supportive feedback loop between AI and edge computing. On the one hand, the distributed nature of the edge/fog paradigm triggers fluctuating workloads for different edge devices depending on location and temporal conditions, making the implementation of edge computing very challenging due to unexpectedness and lack of certainty. In this case, AI optimization challenges are of significant value.

On the other hand, AI needs a fair amount of computing power and appropriate energy backing for learning. However, devices often fail to meet both of these requirements. One solution is to do all the heavy lifting in the cloud, which poses new challenges, including a shortage of bandwidth and elevated latency when dealing with a wide range of different AI models running on a wide range of endpoint devices. The emergence of edge/fog computing enables AI to be distributed close to end devices and users at the edge and on the endpoint with some processing and storage capabilities, addressing the demands for both high network steadiness and minimal latency. In another study conducted by Joshi *et al.* [71], an emerging concept of edge intelligence, highlighted by the authors, was the so-called "all-in-edge" tier, in which the formation and inference of AI models are carried out only by the edge servers. Research on AI-driven optimization of edge/fog systems can be roughly classified into two major categories: optimization-related challenges (e.g., offloading, resource allocation [72], energy consumption [73]) and privacy/security-related challenges [32]. For instance, DRL has been used to effectively learn the network dynamics in the work of. Cheng *et al.* [74]. The authors proposed a DRL-based offloading method based on an embedded space-air-ground network to minimize energy and latency overhead.

IV. 6G–IoT Intelligence: an overview

For future 6G and beyond, there will be a heavy reliance on Network Intelligence (NI) with AI models representing its backbone [86], which ensures that network management will be properly managed and fully automated. The AI models have shown considerable effectiveness in solving complex tasks, such as deriving complicated associations from massive, overlapping data, which will be even more required for future

TABLE VIII: A list of 5G/6G IoT testbeds.

Testbed	Description	Location	Features	Focus	Funding	Ref.
5G-TRANSFORMER	A European testbed for 5G-based IoT services and applications.	Europe	Network slicing, cloud computing, multi-access edge computing (MEC), and virtualization.	5G IoT	EU-funded	[75]
5G-MoNArch	A mobile network architecture testbed for 5G IoT applications.	Europe	Millimeter Wave (mmWave) Technology, MEC, cloud computing, and virtualization.	5G IoT	EU-funded	[76]
5G-PICTURE	A testbed for 5G-based IoT services and applications, including remote surgery and augmented reality.	Europe	Network slicing, MEC, cloud computing, and virtualization.	5G IoT	EU-funded	[77]
5G-CORAL	A testbed for 5G-based IoT services and applications, including smart transportation and energy management.	Europe	Virtualization, cloud computing, MEC (Mobile Edge Computing), and network slicing.	5G IoT	EU-funded	[78]
5G-EmPOWER	A testbed for 5G-based IoT services and applications, including smart cities and healthcare.	Europe	Network slicing, MEC, edge computing, and virtualization.	5G IoT	EU-funded	[79]
6G Flagship	A Finnish research program focused on developing the technology and applications for 6G-based IoT.	Finland	Terahertz and sub-terahertz communications, AI, and edge computing.	6G IoT	Industry and government-funded	[80]
5G Open Innovation Lab	An open innovation platform for developing 5G-based IoT solutions.	USA	Network slicing, MEC, cloud computing, and virtualization.	5G IoT	Industry-funded	[81]
ENCQOR 5G	A Canadian testbed for 5G-based IoT services and applications, including smart cities and healthcare.	Canada	Massive MIMO, MEC, cloud computing, and virtualization.	5G IoT	Government and industry-funded	[82]
5G City	A Danish testbed for 5G-based IoT services and applications, including smart transportation and energy management.	Denmark	Millimeter Wave (mmWave) Technology, MEC, cloud computing, and virtualization.	5G IoT	Industry-funded	[83]
5G Alliance for Connected Industries and Automation (5G-ACIA)	A German-led initiative focused on developing 5G-based IoT solutions for industrial automation and control.	Germany	Network slicing, MEC, cloud computing, and virtualization.	5G IoT	Industry-funded	[84]
6g-platform	The German platform for future communication technologies and 6G.	Germany	Terahertz and sub-terahertz communications, AI, and edge computing.	6G IoT	Government-funded	[85]

sophisticated networks and applications, such as IoE [2]. In this section, we present a vision of the expected effectiveness of intelligence in 6G-IoT systems from three angles as illustrated in Figure 4: 1) the architectural landscape, 2) technological prospects, and 3) security and privacy benefits.

A. Architectural Landscape

Using AI, and specifically, DL provides an innovative path to engineer and improve 6G networks throughout the physical and core layers. Advances in 6G wireless theory and communication will also influence the advancement and expansion of AI in the form of new learning theories and architectures, creating a positive feedback loop, and transforming the wireless landscape, from connecting objects to connecting intelligence [86].

1) Physical (PHY) Layer Enhancements: During the last few years, investigations and efforts have been made to deploy AI in the physical layer of wireless communication networks, including User Equipment (UE) and at the cell edge [86]–[88]. Therefore, several problems with current communication systems remain unaddressed due to inaccurate models or non-linearity [87], such as reciprocity in frequency division duplexing (FDD), predicting channels, detecting and reducing interference. According to Ali *et al.* [87], many of the physical layer's optimization problems, such as spectrum sensing, optimal beamforming formulation, and throughput maximization employing power control, are non-convex. Such problems may be addressed through dual decomposition techniques that require time-consuming iterative algorithms. On the other hand, DL techniques have significant capabilities in addressing such challenges in real-time without sacrificing performance. In addition, AI offers a new way to design the 6G radio interface by further improving the radio environment and communication algorithms. Letaief *et al.* [86] proposed innovative solutions such as joint source-channel coding (JSCC), task-oriented communication, and semantic communication.

2) Medium Access Control (MAC): ML supports an automated learning paradigm shift towards high-performance algorithms for addressing resource allocation challenges within wireless networks [2], [88]. MAC layer-related tasks such as selecting and matching users for multiple-input multiple-output (MIMO), modulation, and handover control can also be optimized using AI [2], [44]. For instance, given that RL can handle combinatorial action spaces with multi-agent settings, RL is considered suitable for such missions where the network can adapt to varying conditions while learning ideal strategies [87]. Therefore, AI plays an important role in 6G MAC layer optimization tasks for the following ML-based predictive tasks: 1) optimal resource allocation (e.g., in non-orthogonal multi-access (NOMA) and massive MIMO (mMIMO)), 2) ML-based predictive power management (e.g. traffic forecasting and prioritized packets separation), 3) FL-based mobility prediction (e.g. federated echo state network), 4) mobile data offloading, 5) link adaptation, and 6) caching [2], [87], [89], [90].

3) Core-Network and Services Intelligence: One of the key benefits of using AI at the core and edge of the network, specifically by integrating training capabilities into the network nodes, is to enable intelligent data collection, processing, delivery, and utilization at the network edge. These benefits will improve the network service quality. Therefore, the inference method can produce reliable and cost-effective services [89]. As one of several cases of this type, device-server co-inferencing can overcome existing traffic and processing constraints by spreading a sizeable DNN network across

different edge node types (I.e., devices and servers) [86]. In addition, future smart self-organizing and self-repairing characteristics, autonomous connections between devices, and the human-centric network with enhanced intelligence within the 6G network translate directly into a wealth of other communication services [2].

4) 5G/6G IoT testbeds: The table VIII provides a list of 5G and 6G IoT testbeds that are crucial for testing and developing advanced 5G and 6G IoT technologies and applications. The testbeds are designed to test and develop 5G/6G-based IoT services and applications, such as smart transportation, energy management, healthcare, and industrial automation. The European Union has funded several of the testbeds listed, including 5G-TRANSFORMER, 5G-MoNArch, 5G-PICTURE, 5G-CORAL, and 5G-EmPOWER. Other funded testbeds include the 5G Open Innovation Lab in the USA, ENCQOR 5G in Canada, 5G City in Denmark, the 5G Alliance for Connected Industries and Automation in Germany, and the 6G-platform in Germany. The testbeds feature network slicing, multi-access edge computing (MEC), cloud computing, virtualization, terahertz and sub-terahertz communications, and artificial intelligence (AI).

B. Supported Technological Prospects

AI-backed context awareness within the network provides satisfying networking experiences that future users and applications require. URLLC focuses on fulfilling the demanding latency and dependability expectations of mission and safety-critical applications. However, the emergence of new applications, such as extended reality (XR) applications, requires reliability and latency requirements that are significantly more challenging than those previously defined in the 5G URLLC. Hence, an upgrade was envisioned, called eXtreme URLLC (xURLLC). According to Park et al [91], xURLLC is based on three fundamental concepts, including 1) quicker and trustworthy data-driven ML-based predictability, 2) the utilization of both radio frequency and non-radio frequency modalities, and 3) collaborative communication and management co-design. The extensive KPIs expectations for future xURLLC and futuristic mission-critical applications, including IoE will require all aspects of innovative 6G-targeted AI-based methodologies and technologies, which can be roughly grouped into two classes: 1) In-Network Domains and 2) In-Application Domains.

1) In-Network Domains: In addition to introducing AI-based enhancements natively into already existing network features, 5G/6G and beyond are also being introduced into a range of other intelligent technologies for efficient management and optimization. Network infrastructure virtualization is an emergent concept for present and future generations of networks. The network slicing [86], [92] defines a networking architectural model permitting multiple isolated and virtualized logical networks on top of the pre-existing physical infrastructure. Each technology is designed to fit specific business conditions for varying applications. The concept is expected to be backed by other emerging technologies, including Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) [44], [86]. The massive volumes of data that will be processed worldwide by 2025 are expected to exceed 180 zettabytes [27]. Hence, AI-powered predictive analytics utilize data to forecast future trends such as future traffic profiles, customer placement, behavior, and preferences. In addition, the concept of proactive caching appears as a recent solution to dramatically reduce peak traffic on the wireless core networks [89].

2) In-Application Domains: Using ML to enhance networking and wireless communications will fundamentally influence software development practices. Considering the need for not only huge speeds and extremely low latency for future mission-critical applications, including advanced healthcare systems (e.g telemedicine), fully autonomous transportation systems, Industry 5.0, meta-universe, and augmented/extended/virtual reality applications, but also the network should be highly adaptive, dynamic, and context-aware which is enabled by AI models. Therefore, opportunistic data transfer (ODT), a context-aware network optimization technique, can be used to circumvent the data transfer challenge of autonomous transport systems. In addition, the ML-based network flow prediction module can both pick network interfaces and program data transfers based on expected resource availability, within a specified application-specific delay tolerance window [87].

C. 5G/6G peer-to-peer IoT communications

With the advent of next-generation networking technologies such as 5G and the forthcoming 6G, we're about to witness an unparalleled transformation in IoT communications. These new-generation networks promise exponential improvements in speed, capacity, and latency, which are vital for efficient and reliable IoT functionalities. This sub-section provides an exhaustive review of the emerging technologies in 5G and projected 6G networking that is poised to revolutionize peer-to-peer IoT communications. Table IX presents the new technologies in 5G/6G IoT communications.

D. 5G IoT Technologies

- Enhanced Mobile Broadband (eMBB): This technology aims to provide significant improvements in data bandwidth and latency compared to 4G networks. It allows IoT devices to transfer data at much higher rates, making it ideal for high-definition video streaming, VR/AR applications, and real-time monitoring [102].
- Ultra-Reliable Low-Latency Communications (URLLC): URLLC is a crucial part of 5G technology. It aims to provide ultra-reliable communication links with low latency, which is important for critical applications such as autonomous vehicles, remote surgeries, and industrial automation [103].
- Massive Machine Type Communications (mMTC): mMTC allows a large number of devices to be connected to the network simultaneously. This is crucial for large-scale IoT deployments, such as smart cities or large-scale agricultural monitoring, where thousands to millions of devices need to be interconnected [104].
- Network Slicing: Network slicing is a 5G feature that allows the creation of multiple virtual networks over a

TABLE IX: 5G/6G Technologies for IoT applications.

Technologies	Key Aspects	Description
Enhanced Mobile Broadband (eMBB) [93]	High Data Rates	- Supports high data rates for bandwidth-demanding applications such as 4K/8K video streaming, virtual reality (VR), and augmented reality (AR). Supports data rates of up to 20 Gbps for downloads and 10 Gbps for uploads. - Supports data rates of up to 20 Gbps for downloads and 10 Gbps for uploads.
	Broad Coverage	- Ensures consistent user experience across different environments — indoors, outdoors, in urban areas, and in more remote locations. - The minimum guaranteed data rate should be 100 Mbps for downloads and 50 Mbps for uploads.
	Network Capacity	- Designed to handle high traffic density, accommodating many simultaneous high-demand users in a small area. - Designed to handle traffic density of up to 10 Mbps/square meter.
	Mobility	- Offers high data rates and quality of service even when the user is moving rapidly. - Aims to provide latency of less than 4 ms for time-critical communications, and less than 20 ms for regular communications.
Ultra-Reliable Low-Latency Communications (URLLC) [94]	Low Latency	- The delay between the sender initiating a data transfer and the receiver getting it. It's crucial for applications like remote surgery or autonomous driving where delays can have severe consequences. - The URLLC Target is 1 millisecond round-trip latency
	High Reliability	- The ability of a system to function correctly, without interruption, over a certain period or amount of data transfer.
	Availability	- The ability of a system to remain in a functional state, even in the presence of faults or disruptions. In URLLC, high availability ensures that the system is functional and accessible when needed.
	Security	- Given the mission-critical nature of URLLC use cases, robust data security measures are essential. Security in URLLC includes data integrity, and confidentiality.
Massive Machine Type Communications (mMTC) [95]	High-density network	- Supports large number of devices (up to 1 million per square kilometer), low data rates (tens of kbps), and low-cost. - Dealing with potential interference issues with large scale devices.
Network Slicing [96]	eMBB Slice, URLLC Slice, and mMTC Slice	- Managing multiple slices over a shared physical infrastructure. - Providing Quality of Service (QoS) according to slice requirements.
Edge Computing [97]	Data Offload	- Edge computing allows data to be processed closer to the source, reducing the amount of data that must be sent back to a central server.
	Bandwidth and Capacity	- By processing data closer to the source, edge computing can reduce the amount of bandwidth required and increase network capacity.
Terahertz (THz) Communications [98]	High-speed wireless access and backhaul	- Using THz frequencies for ultra-high-speed wireless communication. - Beamforming, MIMO techniques, and short-range communication can be used.
	Ultra-high-resolution sensing	- Exploiting THz frequencies for high-resolution sensing applications. - Frequency selection, short-range communication.
	Wireless chip-to-chip communication	- Using THz communication for inter-chip data exchange. - Advanced semiconductor materials and nano-antenna arrays.
Advanced AI and Machine Learning [99]	Network Optimization	- Utilizing AI/ML for predicting and optimizing network conditions.
	Predictive Maintenance	- Predicting failures and maintaining the network efficiently using AI/ML.
Integration of Satellite and Terrestrial Networks [100]	Satellite-backed eMBB (Enhanced Mobile Broadband)	- Satellite communications supporting high data-rate services with peak data rates up to 20 Gbps.
	Satellite-backed URLLC (Ultra-Reliable Low-Latency Communications)	- Satellite networks providing ultra-fast and highly reliable communication with a latency goal of around 1ms.
	Satellite-backed mMTC (Massive Machine Type Communication)	- Satellite networks supporting high-density communication between machines that typically generate small amounts of data.
Advanced Positioning and Sensing [101]	Advanced Positioning in 6G	- Improved accuracy in device positioning, with expected sub-meter or even centimeter-level precision.
	Advanced Sensing in 6G	- The ability for the network to perceive and understand the surrounding environment by using integrated sensors and AI algorithms.

common physical infrastructure. This allows for better resource allocation and quality of service management, especially important in IoT applications where different devices may have vastly different requirements [105].

- Edge Computing: While not a communication technology per se, edge computing is expected to play a significant role in 5G IoT. By moving computation and storage closer to the devices, edge computing reduces latency and network congestion [106].

E. 6G IoT Technologies

- Terahertz (THz) Communications: 6G is expected to utilize the terahertz frequency bands, enabling much higher data rates, potentially up to 100 Gbps or more [107].
- Advanced AI and Machine Learning: AI and ML technologies are expected to play a major role in 6G networks, both in managing the network infrastructure and in processing the vast amounts of data generated by IoT devices [108].
- Integration of Satellite and Terrestrial Networks: 6G is expected to fully integrate satellite networks with terrestrial networks, providing truly global coverage and seamless handoff between different types of networks [109].
- Advanced Positioning and Sensing: 6G networks may include advanced positioning technologies, with accuracy down to the centimeter level, and may also incorporate data from various types of sensors into the network infrastructure [110].
- Advanced Security and Privacy: With the growing concerns about security and privacy, particularly in IoT applications, 6G is expected to incorporate advanced security and privacy features, both at the device level and at the network level [30].

F. Security and Privacy

For 6G networks to successfully address the varying threats within the space-to-air-to-ground embedded network environment and novel technologies, AI is envisioned as the core driver for overcoming both security and privacy challenges [44], [88], [92]. We classify intelligent security in 6G under two categories, namely, 1) Infrastructural Measures and 2) Additional Services.

1) Infrastructural Measures: Integrating training and inference functionalities across multiple nodes in the network helps to maintain privacy and confidentiality, and ensure a high level of security. Considering that the 6G infrastructure is designed for connected intelligence and will deploy AI at different network hierarchy levels, the AI-powered mini-cell can be used to block certain threats such as DoS, man-in-the-middle, and information theft at the lowest levels before it reaches the targeted applications [5].

To preserve privacy, Over-the-Air FL is getting attention as a prominent approach for fast wireless data aggregation through exploiting the property of overlay in multiple access channels, without the need for accessing the network nodes' private data [86]. The context-aware security is a promising area of research where the goal is to obtain a network that can extract the threat level from the existing situation, use the context to determine the necessary security level, and match security levels to security parameters [88].

Another intelligent security concept that is predicted to play an important role in future networks is the Moving Target Defense (MTD). MTD technologies can provide enhanced security flexibility by continuously and dynamically reshaping the underlying system on different layers during the execution cycle. This makes it difficult for adversaries to successfully exploit a continuously evolving, instantaneous, and unpredictable targeted system. For example, when an attack is noticed, the MTD module(s) can tell the SDN controller which way to re-program its data plane to reduce the extent of the attack [44]. Furthermore, various security and privacy architectural mechanisms are proposed [92], including the protection of sensitive customer information such as location, for which privacy-aware ML-based offloading schemes can be used. In addition, biometrics-based authentication or a password-less utility can be adopted within the network's access control arrangements.

2) Additional Services: Apart from the native security intelligence expected for 6G, including the techniques mentioned above, Nguyen *et al.* [44] envisioned that the big improvement in 6G security is the broad adoption of security-targeted AI-driven techniques and in-line deep packet inspection (DPI) capabilities within firewalls and network intrusion detection systems (NIDS). As an example, the authors suggested that the 5G security gateway situated on the Access and Mobility Function (AMF) will require significant enhancements to its existing capabilities which they predict, among other things, in 6G will involve an integrated AI-based intrusion prevention engine. Therefore, other advanced technologies and services predicted to protect the security and privacy of future 6G mobile networks have virtually secured their positions. For example, Blockchain and Distributed Ledger Technologies (DLT) are expected to support next-generation networks, including 6G, in terms of privacy, transparency, and immutability in various critical tasks. These tasks include access control, authentication, mutual trust, anonymization, and single point of failure perturbations avoidance at the infrastructure and application levels [2], [111].

Current Blockchain designs and operational smartness security implementations are mainly realized in two forms, including, 1) programmable software and 2) consensus mechanism. The programmable software is called a smart contract, which automatically performs operations according to predefined conditions. The consensus mechanism, which is basically a distributed algorithm that verifies the validity of submitted candidate blocks, with the verification process being different depending on the type of Blockchain (i.e., public, private, or consortium) as well as its implementation logic (e.g. puzzle-based, vote-based, etc.).

The 6G is expected to introduce ultra-high speeds (over 1Tbps [112]), large data volumes (5 zettabytes per month by 2030 [113]), and a huge number of globally connected mobile devices (13.1 billion by 2023 [114]). AI-based security should be implemented in these operations to keep up with the 6G, and to enable the identification of anomalous patterns in blockchain transactions and potential vulnerabilities in smart contracts at higher speeds and low costs [44].

G. Highlights 5G/6G in Peer-to-Peer IoT Learning

The integration of 5G/6G technology into IoT ecosystems will be transformative, providing a powerful impetus for peer-to-peer (P2P) learning among IoT devices [115]. We explore some of the unique aspects of 5G/6G that enhance P2P IoT learning.

1) Network Slicing: 5G/6G's network slicing capability allows for the creation of bespoke networks tailored to provide different levels of service for diverse types of devices. Such customization will permit high-priority IoT devices to learn from each other unimpeded by lower-priority devices, thereby ensuring the efficacy of the learning process. Wu et al. [116] discuss the use of AI-based solutions across the network slicing lifecycle, thereby allowing for intelligent management of network slices (termed as "AI for slicing"). Then, they explore network slicing solutions constructed to support emerging AI services, which involves creating AI instances and executing effective resource management (termed as "slicing for AI").

2) Increased Speed and Capacity: At the forefront of 5G/6G's unique attributes is its significantly heightened bandwidth and data transfer rate, surpassing its predecessor (4G) by up to tenfold. 4G offers download speeds up to 1 Gbps, 5G offers download speeds up to 20 Gbps, and the still-conceptual 6G is expected to offer speeds up to 100 Gbps. This remarkable speed boost facilitates swift data exchange and processing, allowing IoT devices to communicate and learn from each other in real-time [117].

3) Ultra-Low Latency: Another distinct feature of 5G/6G is its ultra-low latency, which can be reduced to as low as 1 millisecond. This latency minimization is essential for IoT devices, permitting them to have rapid peer-to-peer learning [99].

4) Reliability and Availability: 5G/6G will make strides in improving reliability and availability over its predecessors (i.e., 5G). IoT devices can thus rely on the network to maintain their connections and continue their learning processes without interruptions, assuring transparency in peer-to-peer learning [118].

5) Edge Computing Support: 5G has significantly enhanced edge computing, optimizing cloud computing systems by processing data close to the network's edge, where data generation occurs. This local data processing capability means that IoT devices can learn from data on-site, reducing the need for long-distance data transmission and hastening peer-to-peer learning. Edge Intelligence uses AI methods, which are anticipated to be a crucial factor in the development of future 6G networks, bolstering their performance, enabling new services, and introducing new functions [119].

6) Energy minimization: Le *et al.* [120] apply reconfigurable intelligent surface (RIS)-aided wireless power transfer to improve battery life in federated learning (FL)-based wireless networks. Therefore, optimizing transmission time, power control, and the RIS's phase shifts, shows that the total transmit power can be minimized while meeting both minimum harvested energy and transmission data rate requirements. Zarandi and Tabassum [121] present a federated deep reinforcement learning (DRL) framework using double deep Q-network (DDQN) to optimize multi-objective problems in IoT devices, minimizing task completion delay and energy consumption. The framework enhances scalability and privacy, and the results show faster learning speeds than federated DQN and non-federated DDQN. Zheng *et al.* [122] proposes an optimization strategy for wireless-powered multi-access edge computing networks to minimize total computation delay. Using a mixed integer programming model, the proposed strategy breaks down the problem into sub-parts for offloading decisions and power transfer duration optimization. In addition, the proposed strategy leverages deep reinforcement learning to handle time-varying channel conditions, which leads to near-minimal computation delays and low computational complexity.

V. Edge Learning for Cyber Security

Edge learning, a variant of machine learning, leverages the power of edge computing—processing data close to its source, thereby reducing latency and bandwidth usage. This technology, when applied to cybersecurity, provides swift, real-time detection and mitigation of cyber threats, thereby boosting the overall resilience of the systems. This section aims to comprehensively discuss the use of edge learning and its application in cybersecurity.

A. Differential privacy-based solutions

Differential privacy-based solutions [137] can protect sensitive data in the context of IoT security by adding random noise to the data, making it difficult for attackers to identify individual users or extract sensitive information. This technique can effectively prevent privacy breaches and unauthorized access to data, which are major concerns in IoT security. Moreover, differential privacy can help build trust between users and IoT systems by providing transparency and assurance that their data is safeguarded.

Truex *et al.* [123] presents a new approach to address the privacy and accuracy trade-offs associated with existing federated learning systems. The authors highlight the vulnerabilities of existing federated learning systems and propose a solution that combines differential privacy and secure multiparty computation. The authors' approach combines these two techniques that enable the reduction of noise injection as the number of parties increases without sacrificing privacy while maintaining a pre-defined rate of trust. The system is designed to withstand three potential adversaries, including an honest-but-curious aggregator, colluding parties, and outsiders. The system assumes secure communication channels between each party and the aggregator, and the use of the threshold variant of the Paillier encryption scheme. This scheme ensures the privacy of individual messages sent to the aggregator, preventing any set of parties from decrypting ciphertexts. The proposed FL system remains resilient to an inference against potential adversaries who may be users of the service. The experimental results validate the system's effectiveness and superiority over state-of-the-art solutions, making it a scalable and reliable approach to federated learning. Zhou *et al.* [124] proposes a privacy-preserving federated learning scheme in fog computing that enables fog nodes to collect Internet-of-Things (IoT) device data and complete the learning task. The proposed scheme addresses the issues of the uneven distribution of data and the large gap of computing power, which affects the efficiency of training and model accuracy in federated learning. The scheme leverages differential privacy to resist data attacks and uses a combination of blinding and Paillier homomorphic encryption to secure model parameters. The proposed scheme is formally verified to guarantee both data security and model security and to resist collusion attacks.

Friha *et al.* [131] proposed a decentralized, secure, and Differentially Private (DP) Federated Learning (FL)-based IDS (2DF-IDS) system for securing smart industrial facilities. The proposed system offers high performance in identifying different types of cyber attacks in an Industrial IoT system while mitigating the risks associated with conventional FL approaches. Additionally, the system offers improved overall performance compared to other competing FL-based IDS solutions, particularly under strict privacy settings. You *et al.* [133] proposed a Federated Adaptive Accuracy Controlling (FedAAC) framework, which dynamically controls the model accuracy to match the contribution of existing participating clients. To achieve this, an Accuracy Degrading algorithm is designed to obtain a decaying version of the accuracy of the global model by executing the accuracy degrading task specified by the server on the client side. To address the unbalance between the client and the central server for the model reward, assurance is introduced to ensure clients' contributions always match the model reward they receive. The differential privacy mechanism is also introduced into the FedAAC implementation, and the client-level differential privacy approach is improved for the scenarios targeted by FedAAC.

TABLE X: Edge Learning-based Security and Privacy Solutions.

Framework	Year	Threat model	Mitigation solution	Learning mode	ML model	Datasets	Pros (+)	Open Issues (-)
Truex *et al.* [123]	2019	An honest-but-curious aggregator	Threshold Homomorphic Encryption	Federated Edge Learning	DT, CNN, SVM	Nursery Data Set	+ The proposed FL system remains resilient to an inference against potential adversaries	- High computational and communication costs
Zhou *et al.* [124]	2020	Collusion attacks	Paillier homomorphic encryption	Federated Edge Learning	FNN	Fashion-MNIST dataset	+ The proposed scheme is formally verified to guarantee both data security and model security and resist collusion attacks	- The experiments are based only on a single dataset (Fashion-MNIST), which may not provide sufficient evidence of the scalability in other scenarios
Mothukuri *et al.* [125]	2021	IoT network attacks	Ensemble Learning	Decentralized Edge Learning	LSTM and GRU	Modbus network data set	+ The use of long short-term memory (LSTM) and gated recurrent units (GRUs) neural network models helps in achieving higher accuracy rates	- The experimental results presented in the article are based on a specific data set and may not necessarily generalize to other IoT networks or data sets
Cui *et al.* [126]	2021	DoS, U2R, R2L, and Prob attacks	Generative Adversarial Network	Decentralized Edge Learning	CNN	KDD CUP 1999	+ Preserves the privacy of local model parameters while improving the utility of the anomaly detection model	- The proposed idea is evaluated with old dataset
Xu *et al.* [127]	2022	N/A	Fixed-point quantization method	Federated Edge Learning	MLP	MNIST dataset and CIFAR10 dataset	+ Achieves higher accuracy and lower quantization error than other quantization methods	- The proposed idea is not evaluated with IoT dataset
Friha *et al.* [128]	2022	IoT network attacks	Deep learning solution	Federated Edge Learning	CNN, RNN, DNN	CSE-CIC-IDS2018, MQTTset, and InSDN	+ FELIDS can overcome the privacy issues associated with centralized machine learning models by using a federated learning approach	- Considers network traffic and does not take into account other factors such as device security or physical attacks
Gao *et al.* [129]	2023	Malicious server may conduct dishonest data aggregation	Homomorphic encryption	Federated Edge Learning	CNN	Fashion-MNIST dataset	+ It ensures the integrity of participant-uploaded parameters and the correctness of aggregated results from the server	- The paper does not discuss the limitations or scalability of SVeriFL
Ouyang *et al.* [130]	2023	The privacy risks in the existing FL blockchain design patterns	Blockchain and Smart Contracts	N/A	N/A	N/A	+ The framework is secure, private, and decentralized	- Future research is needed to explore the scalability and performance of the framework in more complex scenarios
Friha *et al.* [131]	2023	Quantum-based Cryptoanalysis Attacks	A differentially private gradient exchange scheme	Decentralized Edge Learning	DNN	Edge-IIoT dataset	+ The proposed system achieves a high level of performance, with comparable accuracy to the centralized learning approach	The proposed system is vulnerable to adversarial attacks
Li *et al.* [132]	2023	Malicious local models	Dynamic Weighted Aggregation	Federated Edge Learning	CNN	CSE-CIC-IDS2018	+ The client filtering and local model weighting strategy improves the global model's performance and reduces communication overhead	- Considers network traffic and does not take into account other factors such as device security or physical attacks
You *et al.* [133]	2023	The vulnerability in the fairness of model reward	Differential privacy	Federated Edge Learning	Pre-trained VGG	CIFAR10, Rice, Fashion-MNIST, and CIFAR10 datasets	+ The proposed FedAAC framework provides a scalable solution that can be extended to existing federated learning approaches	The proposed system is vulnerable to adversarial attacks
Baucas *et al.* [134]	2023	Attacks against data privacy in wearable IoT devices	Blockchain Technology	Federated Edge Learning	CNN	HAR dataset	+ The model accuracy shows the platform's ability to preserve the integrity of a predictive service	- The paper does not discuss the scalability of the proposed platform as well as the potential latency issues
Abou El Houda *et al.* [135]	2023	Network attacks	SDN and Blockchain	Federated Edge Learning	CNN	NSL-KDD dataset	+ The paper shows that MiTFed achieves high accuracy and efficiency in detecting new and emerging security threats in both binary and multi-class classification	- The dataset used in the performance evaluation is outdated
Chen *et al.* [136]	2023	Attacks in P2P networks	An enhanced Eschenauer-Gligor (E-G) scheme	Decentralized Edge Learning	CNN	SMS Spam Collection	+ The proposed system can resist various security threats, preserve user privacy, and achieve better computation efficiency and prediction performance	- The proposed system is vulnerable to adversarial attacks

DT: Decision trees, CNN: Convolutional neural networks, SVM: Support Vector Machines, FCNN: Fully Connected Neural Network, LSTM: Long Short-Term Memory, GRUs: gated recurrent units, MLP: Multi-Layer Perceptron.

Chen *et al.* [136] proposes a decentralized global model training protocol, named PPT, that addresses security, privacy preservation, and robustness requirements in the context of Federated Learning in P2P networks. The paper proposes solutions to various security threats and privacy preservation using the symmetric cryptosystem, secure key distribution, and random noise generation. PPT also adopts game theory to resist collusion attacks and has elaborate designs for communication efficiency and dropout-robustness. Extensive experiments show that PPT outperforms Google's Secure Aggregation and LDP-based FL methods in computation efficiency and prediction performance, but the proposed system is vulnerable to adversarial attacks.

B. Ensemble Learning-based solution

Ensemble Learning [138] is a machine learning technique that combines multiple individual models to create a more robust and accurate prediction model. When it comes to IoT security, Ensemble Learning can be a valuable tool to enhance the security of IoT devices and networks. Mothukuri *et al.* [125] proposed a decentralized federated learning approach to enable anomaly detection on Internet of Things (IoT)

networks. This approach enables on-device training, thereby eliminating the need to transfer network data to a centralized server, ensuring optimal results in predicting intrusion in IoT networks. The proposed approach utilizes long short-term memory (LSTM) and gated recurrent units (GRUs) neural network models to train the machine learning model on the Modbus network data set. The article also provides an overview of LSTM and GRU networks, their architecture, and real-time implementations. The proposed approach is beneficial in securing data privacy at end devices, achieving optimal results, and being computationally inexpensive.

C. GAN-based solution

Generative Adversarial Networks (GANs) [139] have been applied successfully in various fields, including computer vision and natural language processing. In recent years, researchers have started exploring their use in IoT security. One potential application of GAN-based solutions is to generate realistic attack scenarios and create diverse datasets for training security models. GANs can also be used to generate synthetic data that mimics real-world sensor data. This can be used to detect anomalies and prevent security breaches. Furthermore, GANs can generate adversarial examples to test the robustness of IoT security systems. Cui *et al.* [126] presented an approach for decentralized and asynchronous Federated Learning (FL) for Internet of Things (IoT) anomaly detection. The proposed approach uses a modified Generative Adversarial Network (GAN) model called DP-GAN, which has an additional perceptron, the DP identifier (DPI), to generate differential noise, while approximating the raw data to a better degree. The proposed method also integrates blockchain to achieve global aggregation and improve system reliability. The paper demonstrates the effectiveness of the proposed approach in terms of accuracy, privacy protection, and efficient convergence, as shown by the experiments on benchmark datasets.

D. Fixed-point quantization method-based solution

Fixed-point quantization is a method used in digital signal processing to reduce the storage requirements and computational complexity of numerical data. In IoT security, this technique can be applied to reduce the amount of data transmitted over the network, minimizing the risk of data leakage and unauthorized access. Xu *et al.* [127] proposes Federated Learning with Minimum Square Quantization Error (FedMSQE), an approach to address the challenges of deploying popular neural network models on Internet of Medical Things (IoMT) devices. The paper focuses on the security of neural network model transmission and reducing the network scale. The proposed method uses fixed-point quantization with stochastic rounding to achieve the smallest quantization error for each individual client in the FL setting. The authors demonstrate through numerical experiments that their proposed algorithm achieves higher accuracy and lower quantization error than other quantization methods, and can be applied to any deep neural network on any single device. FedMSQE improves the security of FL and reduces the transmission cost while maintaining high accuracy.

E. Deep Learning-based solution

By analyzing large datasets of IoT device behavior and network traffic, deep learning algorithms can detect patterns and anomalies that may indicate malicious activity [140]. Friha *et al.* [128] proposed FELIDS, a federated IDS based on deep learning for mitigating cyberattacks on agricultural IoT infrastructures. The authors identify gaps in the literature, including outdated or contextually inappropriate datasets, privacy issues for centralized models, and limited threat models that only address a subset of the attack vectors. FELIDS uses three deep learning classifiers - DNNs, CNNs, and RNNs - and evaluates their performance using three recent real-world traffic datasets. The paper provides a detailed performance evaluation and comparative analysis between FELIDS, the centralized machine learning model, and state-of-the-art works. The paper uses three recent real-world traffic datasets, namely: CSE-CIC-IDS2018, MQTTset, and InSDN, to evaluate the performance of each classifier.

F. Homomorphic encryption-based solution

Homomorphic encryption [141] can protect the algorithms and models used in IoT applications, preventing attackers from reverse engineering or tampering with these crucial components. Gao *et al.* [129] proposed a Homomorphic encryption-based solution, named SVeriFL, that provides a solution to the privacy and security challenges in federated learning, ensuring data integrity and correctness of uploaded parameters and aggregated results. It also allows for consistency verification among multiple participants. The scheme employs a dynamic successive verification mechanism, running throughout the FL training process without impacting the original FL performance. The security analysis and experiments demonstrate the effectiveness of the proposed system, with moderate computational cost.

G. Blockchain and smart contracts-based solution

The integration of blockchain and smart contracts can offer a robust and decentralized solution for IoT security [142]. By leveraging blockchain technology, data transactions can be recorded and verified by a distributed network of nodes, ensuring that data integrity is maintained and tampering is prevented. Ouyang *et al.* [130] introduced a privacy framework for FL based on blockchain and smart contracts. The framework offers complete privacy services for off-chain federations that want to use on-chain FL. The proposed framework is implemented and analyzed based on two prevalent blockchain projects, Ethereum and inter-plenary file systems (IPFS). The experiments show that the framework has acceptable collaboration costs and has advantages in privacy, security, and decentralization. Additionally, the framework can enable automatic on-chain identification and autonomous FL of machine clusters made up of IoT devices or distributed participants. Baucas *et al.* [134] proposes a fog-based IoT platform that uses federated learning and blockchain technology to address the challenges of preserving patient data privacy and improving the security of wearable IoT devices. The platform enforces decentralized

servers and resource reallocation to enhance adaptability and sustainability. Federated learning is used to preserve patient data privacy, while blockchain technology provides access control and a cryptographic structure to improve security. The platform's testbed effectively simulates and evaluates the proposed implementation, and the model accuracy shows the platform's ability to preserve the integrity of a predictive service.

H. Dynamic Weighted Aggregation-based solution

The dynamic Weighted Aggregation-based solution is a promising approach for addressing the security challenges associated with IoT. This solution involves integrating multiple security measures and assigning weights to each mechanism based on their importance and effectiveness in a given context. The weights are then dynamically adjusted based on the changing security environment and the behavior of IoT devices. Li *et al.* [132] introduces DAFL, a dynamic weighted aggregation federated learning system for intrusion detection. The proposed approach reduces the effect of poorly performing local models on the global model during training, which improves the global model's performance in intrusion detection and reduces communication overhead. The paper presents the system design and key implementation details of DAFL and evaluates its performance against existing approaches. The experimental results demonstrate that DAFL achieves better detection performance with lower communication overhead, but the proposed system does not take into account other factors such as device security or physical attacks.

I. Software-defined networking

Software-defined networking (SDN) can offer a robust solution for addressing the security risks associated with IoT [143]. SDN-based security solutions allow for fine-grained access control and enforcement policies, ensuring the protection of IoT devices and networks from unauthorized access. Abou El Houda *et al.* [135] proposes a framework called MiTFed that enables multiple SDN domains to collaboratively build a global intrusion detection model using FL, blockchain, and SDN technologies. The framework achieves high accuracy and efficiency in detecting new and emerging security threats while preserving the privacy of collaborators, making it a promising solution to address security threats in large-scale distributed networks. Further research is needed to assess the scalability of the proposed framework and address potential performance issues due to the use of blockchain technology as well as the dataset used in the performance evaluation is outdated.

VI. Datasets for Edge Learning

Following upon a thorough analysis of recent contributions in the literature, we provide a detailed report on the datasets used by the scientific community for experimenting and evaluating ML techniques on cyber attacks, which is provided in Table XI. Based on the content of each dataset, we classify them into the following seven main categories: 1) IoT-based multi-purpose security, 2) Attack classification, 3) Image classification, 4) Time series classification, 5) Human Activity Recognition (HAR), 6) Sentiment classification, 7) Location awareness, and 8) Text classification, as illustrated in Figure 6.

A. IoT-based multi-purpose security

In recent years, addressing IoT security has received a great deal of focus from academia, industry, and governments, mainly motivated by the expanding attack surface from existing threats across the IoT networks. The availability of sophisticated, easy-to-run, and easy-to-find exploit scripts and specialized utility [140]. IoT-based security defenses using AI (specifically anomaly-based approaches) have provided a good line of defense against these threats, and the efforts provided by the security community deserve mention [7]. However, given that one of the most important factors in providing an effective final model is the data. This means that it is safe to say that the final model is well-founded if the data is well-grounded. Although different datasets have been proposed to design, train, and evaluate AI-based security mechanisms for IoT-related environments, the majority of them are either context-specific (e.g., a specific protocol), constrained threat model (limited types and quantities of attacks), or collected within a shallow generation framework (either real or virtual) [56]. Others, however, have been designed as IoT-based multi-purpose security-related datasets such as the Edge-IIoTset dataset.

- *Edge-IIoTset dataset [56]:* is a comprehensive and realistic dataset collected within multiple IoT/IIoT environments, intended to be used by ML-based cybersecurity models for training and assessment under both training approaches (centralized and FL). Data are collected from a variety of sources, including notifications, logs, and network traffic. A total of 61 features were selected from 1176 encountered characteristics. In particular, the dataset is produced with a designed IoT/IIoT test bed featuring a broad spectrum of interconnected devices, sensors/actuators, protocols, and multi-layer implementations. The generation framework is designed in a multi-layer approach comprising seven layers, including 1) the IoT/IIoT perception layer where there exist more than 10 types of devices including flame sensors, ultrasonic sensors, and water level detection sensors, 2) the edge layer where various brokers and controllers are implemented 3) the fog layer where various platforms are installed, such as IoT hubs and digital twins 4) the SDN layer 5) the NFV layer 6) the blockchain layer, and 7) the application layer. In addition to general-purpose protocols such as HTTP, the dataset includes a range of IoT and IIoT-specific protocols such as MQTT, CoAP, and Modbus. The dataset includes 14 attack classes, including DoS/DDoS, the man-in-the-middle, malware, and injection attacks. ProvNet-IoT proposed in [144], is a forensic provenance-based scheme for examining IoT-targeted network-level assaults. The system applies progressive provenance within the network to illustrate various events within different attack strategies and gen-

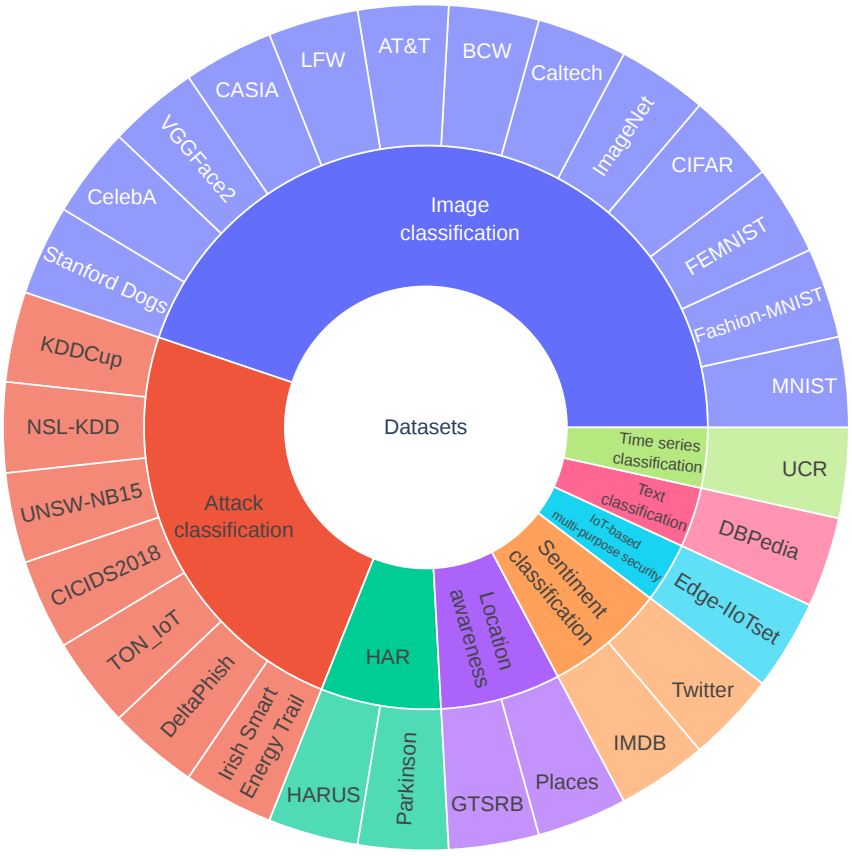


Fig. 6: Datasets used for machine learning security implementations and evaluations

erate forensic proofs. The authors employed various community detection algorithms including Label Propagation (LPA). Experimental results on the Edge-IIoTset dataset assessed and demonstrated the potential of ProvNet-IoT to recognize specific selective artifacts in order to generate credible proof in forensic examinations. In the same context (i.e., IoT security), but in a different space, namely, smart health system security, Ghourabi *et al.* [145] proposed a two-tiered security system. The first one is an IDS for smart medical devices, including the Internet of Medical Things (IoMT), and the second one is a malware detection system for general-purpose devices. The proposed system is built on the basis of an optimized LightGBM and a transformer-based model. Evaluations under the Edge-IIoTset reached accuracies as high as 99%.

B. Attack classification

ML techniques provide valuable tools for cybersecurity solutions to detect intrusions, identify malware and implement mitigation schemes. Various datasets have been proposed to assist researchers in designing, developing, and evaluating their cybersecurity defense methods, including:

- *KDDCup dataset [182]:* since 1999, it has been the most widely used dataset for assessing anomaly detection systems during that decade. It was built on the foundation of roughly 4 GB of tcpdump data from seven weeks of actual network traffic (collected for the DARPA'98 IDS evaluation program). Within the training dataset, there are about 4.9 million individual connection vectors, with each holding 41 features with labels of either normal or attack. The attack types include DoS, User to Root (U2R), Remote to Local (R2L), and probing attack. Although this dataset is considered somehow outdated, various modern studies continue to use it. For example, the work of Fung *et al.* [147] proposed a defense mechanism against Sybil-based poisoning attacks, called FoolsGold, which is used for securing FL-based learning. The system proposed places no restriction on the number of expected attackers, nor does it necessitate additional information beyond the learning process, while making minimal assumptions regarding the clients and their data. For the KDDCup dataset, the authors trained a single layer fully-connected softmax for the classification task, using 5 clients performing 5 attacks on the same target class. In the same context, Li *et al.* [148] proposed LoMar, a Local Malicious Factor defense algorithm for addressing poisoning attacks on FL. The proposed algorithm is based two main phases. The first phase rates each remote client's model updates by scaling relative distribution across their neighbors using a kernel density estimation technique. In the second phase, a statistically optimal trigger threshold is derived to differentiate between malicious and safe

TABLE XI: Datasets used as a benchmark experiment in the evaluation of Machine learning vulnerabilities

Datasets	Attack category	Attack type	Machine learning	Learning mode	Targeted ML	Systems
Edge-IIoTset dataset	Sybil attacks	Miscellaneous	LPA	Centralized	Attack classification	[144]
	Sybil attacks	Miscellaneous	LightGBM, BiLSTM	Centralized	Attack classification	[145]
Reuters-21578 dataset	Backdoor attack	RNN backdoor attacks	RNN	Centralized	Text classification	[146]
KDDCup dataset	Sybil attacks	Sybil-based poisoning attack	SofT	Distributed	Attack classification	[147]
	Poisoning attacks	Federated poisoning attack	SqueezeNet	Federated	Attack classification	[148]
CIFAR-10 dataset	Backdoor attack	Imperceptible backdoor pattern	DNN	Centralized	Image classification	[149]
	Poisoning attacks	Clean-label poisoning attack	ResNet and VGG-16	Centralized	Image classification	[150]
	Poisoning attacks	Clean-label poisoning attack	Deep k-NN	Transfer	Image classification	[151]
	Poisoning attacks	Federated poisoning attack	ResNet	Federated	Image classification	[152]
	Poisoning attacks	Federated poisoning attack	ResNet-18	Federated	Image classification	[153]
MNIST dataset	Backdoor attack	Distributed backdoor attack	LeNet5	Centralized	Image classification	[154]
	Poisoning attacks	Poisonous label attack	CNN	Centralized	Image classification	[155]
	Poisoning attacks	Federated poisoning attack	MLP	Federated	Image classification	[156]
	Poisoning attacks	Federated poisoning attack	CNN	Distributed	Image classification	[157]
	Sybil attacks	Sybil-based poisoning attack	SofT	Distributed	Image classification	[147]
	Poisoning attacks	Federated poisoning attack	LetNet	Federated	Image classification	[152]
Caltech-101 dataset	Synergetic attack	Synergetic attack	ResNet	Centralized	Image classification	[158]
UCR archive	Adversarial attack	Black-box adversarial attack	FCN and ResNet	Centralized	Time series classification	[159]
HARUS Dataset	Poisoning attacks	Federated poisoning attack	FML	Federated	HAR	[160]
CASIA dataset	Poisoning attacks	Generative poisoning attack	FaceNet	Centralized	Face recognition	[161]
TON_IoT dataset	Poisoning attacks	Federated poisoning attack	KNN, LR, RF	Federated	Attack classification	[162]
VGGFace2 dataset	Sybil attacks	Sybil-based poisoning attack	SqueezeNet	Distributed	Face recognition	[147]
LibriSpeech dataset	Adversarial attack	Black-box adversarial attack	DNN	Centralized	Speech Recognition	[163]
Fashion-MNIST	Byzantine attacks	Local model poisoning attack	ResNet20	Federated	Image classification	[164]
	Poisoning attacks	Federated poisoning attack	CNN	Federated	Image classification	[165]
	Sybil attacks	Sybil-based collusion attack	CNN	Federated	Image classification	[166]
	Dropping Attack	Targeted dropping attack	CNN	Federated	Image classification	[167]
UCF-101 dataset	Adversarial attack	Black-box adversarial attack	I3D, CNN+LSTM	Centralized	Video Recognition	[168]
AG's news dataset	Adversarial attack	White-box adversarial attack	CharCNN-LSTM	Centralized	Text classification	[169]
NSL-KDD dataset	Adversarial attack	White-box adversarial attack	DNN	Centralized	Attack classification	[170]
Parkinson Dataset	Poisoning attacks	Federated poisoning attack	FML	Federated	HAR	[160]
FEMNIST dataset	Poisoning attacks	Federated poisoning attack	DNN	Federated	Image classification	[171]
	Poisoning attacks	Federated poisoning attack	N/A	Federated	Image classification	[10]
CICIDS2018 dataset	Poisoning attacks	Clean label attack	CNN-FL	Federated	Attack classification	[172]
IMDB dataset	Poisoning attacks	Federated poisoning attack	SVM	Federated	Sentiment classification	[173]
	Poisoning attacks	Label flipping attack	BiLSTM	Federated	Sentiment classification	[174]
AT&T dataset	Poisoning attacks	Federated poisoning attack	CNN	Federated	Face recognition	[175]
GPS trajectory dataset	Poisoning attacks	Data aggregation attack	N/A	Federated	Travel mode detection	[176]
Osteoporotic Fracture	Poisoning attacks	Data poisoning attack	DNN	Centralized	Multivariate numerical	[177]
LFW dataset	Poisoning attacks	Generative poisoning attack	FaceNet	Centralized	Face recognition	[161]
ImageNet10 dataset	Backdoor attacks	Clean label backdoor attack	DNN	Centralized	Image classification	[178]
UNSW-NB15 dataset	Poisoning attacks	Label flipping attack	CNN-FL	Federated	Attack classification	[172]
Stanford Dogs dataset	Poisoning attacks	Data poisoning attack	ResNet-13	Centralized	Image classification	[177]
Irish Smart Energy Trail	Poisoning attacks	Data poisoning attack	FFNN, GRUs, AEA	Centralized	Electricity theft detection	[179]
Breast Cancer Wisconsin	Byzantine attacks	Local model poisoning attack	ResNet20	Federated	Image classification	[164]
GTSRB dataset	Combined attacks	Trojaning attack	DNN	Centralized	Sign recognition	[180]
DBPedia-14 dataset	Dropping attacks	Targeted dropping attack	CNN	Federated	Text classification	[167]
DeltaPhish dataset	Adversarial attacks	Grey-Box adversarial attack	13 classifiers	Centralized	Phishing website detection	[181]

LPA: Label Propagation, LightGBM: light gradient-boosting machine, biLSTM: Bidirectional LSTM, RNN: Recurrent Neural Network, DNN: Deep Neural Network, LeNet5: DNN architecture for recognizing the handwritten and machine-printed characters, ResNet: Residual neural network, FCN: Fully Convolutional Network, MLP: Multilayer Perceptron, CNN: Convolutional neural network, LSTM: Long short-term memory, I3D: inflated 3D convolutional network, CharCNN-LSTM: Consists of a 2-layer stacked LSTM width 6 for temporal convolutions, SVM: Support Vector Machine, KNN: k-nearest neighbor, LR: linear regression, RF: random forest, FML: Federated multitask learning, SofT: A single layer fully-connected softmax for classification, BiLSTM: Bidirectional Long/Short-Term Memory, VGG-16: Very deep convolutional networks for large-scale image recognition, FFNN: feed-forward neural networks, GRUs: gated recurrent units, AEA: deep auto-encoder with attention.

updates. For the KDD Cup dataset, the authors use 2 classes with over two hundred thousand data samples as their source and target label, while setting up three malicious clients with flipped samples. The target label accuracy reported is 99.1%.

- *NSL-KDD dataset [183]*: is a proposed dataset to overcome certain issues associated with the KDDCup dataset, including failure to provide a clear definition of the attacks. It consists of two main parts: 1) KDDTrain+ with 125,973 records and 2) KDDTest+ with 22,544 records, which are created from the KDDCup dataset, with four main categories of attacks, namely R2L, Prob, DoS, and U2R. A recent work by Wang *et al.* [170] proposed IFPA and IUA, two integration-based adversarial generation techniques. These techniques have the potential to conduct DNN-targeted white-box attacks. The IFPA is designed for cases where a specific number of points are to be disturbed, while the IUA is designed for cases where no perturbation point number is requested. The performance of the NSL-KDD dataset achieves an overall acceptable accuracy of 78.96% on the test set.
- *UNSW-NB15 dataset [184]:* developed in the Cyber

Range Lab at the Australian Cyber Security Centre (ACCS) as a hybrid of real-world and synthetic attack models. It holds over 2.5 million data instances with 49 features mined out of 100 GB of raw traffic. It includes nine types of attack including DoS, shellcode, reconnaissance, generic, exploits, and worms. The work by Zhang *et al.* [172] proposed *SecFedNIDS*, a poisoning attacks-robust FL-based NIDS. To effectively shadow poisonous traffic data and avoid its participation in further local training, SecFedNIDS employs a novel detection method for poisonous data based on the similarity of class paths. The authors followed a layer-wise suitability propagation technique to retrieve the classpath from the clean traffic data and transmit it to poisoned clients to assist in discriminating the poisoned data. The results showed that SecFedNIDS increases accuracy in case of poisoning attacks on the UNSW-NB15 dataset by up to 48%.

- *CICIDS2018 dataset [185]:* generated by the Canadian Institute for cyber security [185] using the profiles notion containing comprehensive descriptions of attacks and detailed abstract distribution models for applications, protocols, or lower-level network entities. In addition to the Benin profile, the dataset includes different attack types such as DoS/DDoS, botnet, infiltration, web, and brute-force attacks. This dataset has been widely used for evaluating Host and network IDSs in the last few years.
- *TON_IoT dataset [186]:* generated by collecting and analyzing heterogeneous data collections from IoT and IIoT domains. The experiment setup to generate this dataset uses several VMs with different operating systems to support interconnectivity between IIoT, cloud, and edge/fog layers. Different attack types can be found in this dataset including DoS/DDoS, ransomware, and web apps attacks, targeting IoT gateways and systems across the IoT/IIoT domain. Khan *et al.* [162] proposed *DepoisoningFSL*, a data poisoning defense federated split learning for edge computing. When the proposed system is compared to the KNN-based semi-supervised defense (KSSD) mechanism with different data poisoning rates (up to 25%), it presents an overall increased accuracy.
- *DeltaPhish dataset [187]:* was gathered from active phishing URLs retrieved online from the PhishTank feed during the period of Oct 2015 to Jan 2016. The authors gathered and verified more than 1000 phishing pages manually. Within each phishing page, they subsequently harvested the related homepage of the hosting domain. Using hyperlink analysis in the HTML code, three to five legitimate pages are collected and manually approved. In total, there are 5,511 distinct web pages included, out of which 1,012 are phishing pages. The authors in [181] proposed a collection of Grey-box attacks on phishing detectors that a phishing adversary can use. The attacks vary according to the knowledge that the adversary has of the specific phishing detector. In addition, the authors introduced an associated defense algorithm, named Protective Operation Chain (POC), which is based on a mix of random feature picking. The feature linkages are exploited to minimize the attacker's guess of the target's phishing detector. Experiment evaluations on different public datasets including the DeltaPhish dataset showed that the proposed algorithm is robust to attacks on 13 different classifiers.

C. Electricity theft detection

Electricity theft is a significant concern for utility systems as it causes high economic losses. Furthermore, the shift from controlled to smart devices may introduce highly sophisticated attacks that are harder to defend against. This requires AI-based defense mechanisms to mitigate the risks, which in turn requires Benin profile data to differentiate the pattern of attacks. Examples of such datasets include:

- *Irish Smart Energy Trail:* collected by the Sustainable Energy Authority of Ireland, where the electricity records originate from 3,000 residential units' smart meters that performed 30-minute readings for 18 months. As a result, there are 25,000 reports per customer. This dataset is used by Takiddin *et al.* [179] as a benign profile dataset to quantify the effect of data poisoning attacks on smart grids. The authors evaluated the capabilities of customer-specific and generalized baseline detectors for detecting data poisoning and power theft attacks.

D. Image classification

Image classification represents a powerful task for assessing the architectures and modern methodologies within the area of computer vision. Image classification is a core discipline concerned with understanding what an image looks like in its entirety, and the objective is to classify the given image with a specific category. Various datasets have been proposed to evaluate such security architectures and systems, including:

- *MNIST dataset [188]:* built from the NIST Special Databases (1 and 3), and containing binary images of numbers and handwritten characters. This dataset is widely used for training various AI-based image processing systems and can be considered the default dataset for image classification tasks around the globe. A vast amount of hand-written data is embedded in the dataset with over 70,000 images of 0 to 9 hand-written digits, which have been standardized in size and focused in a pixel-square grid. Within each image is an array ($28 \times 28 \times 1$) of varying floating-point numbers that depict grayscale shades of intensity. This dataset has been used widely for evaluating different edge-located security schemes in recent works [147], [152], [154]–[157].
- *Fashion-MNIST dataset [189]:* Based on the broad success that the MNIST dataset has achieved, various works have attempted to take a similar approach. For example, Fashion-MNIST is designed to become a full-scale substitute for the original MNIST dataset when evaluating ML-based algorithms. The dataset shares the identical image size (70,000), the same data format (28x28 Grey-scale imagery), and the identical training (60,000 images) and testing (10,000 images) setup. However, instead of

handwritten digits, Fashion-MNIST is made up of fashion items that fall under ten categories, each with 7,000 images per category. This dataset is also used for evaluating different edge-located security schemes such as in [164]–[167].

- *FEMNIST dataset [190]:* is a federated version of the MNIST dataset containing 341873 samples coming from more than 3300 writers as FL clients. With distinguishable writing styles, individual authors' data is Non-Independent and Identically Distributed (Non-IID). Each client in the FEMNIST dataset has approximately 100 samples with 10 classes. This dataset has been used to evaluate edge-enabled FL-based security systems recently [10], [171].
- *CIFAR-10 and CIFAR-100 datasets [191]:* represent labeled subsets from the 80 million tiny images dataset. The CIFAR-10 dataset is made up of 60,000 color images of 32x32 split into 10 classes, each with 6,000 images per class. It includes 50,000 training and 10,000 test images. The dataset is partitioned into 5 training batches and 1 test batch, each one containing 10000 images. The testing batch contains 1000 images randomly picked from every class. Each training batch contains the remaining images in a completely random order, although some of them may have more images within 1 class than any other. In addition, the training batches hold 5000 images from every class. The work by Xiang *et al.* [149] used this dataset to benchmark their proposed backdoor attacks defense mechanism.
- *ImageNet10 dataset [192]:* is a hand-picked 10-class dataset derived from the ImageNet dataset, which provides high-resolution images divided into 1000 classes and formed only from the 320x320 core regions of the original images. The ImageNet10 classes were chosen to be optically distinct and to reflect natural objects. This dataset is used in different works for benchmarking along with other datasets such as MNIST and CIFAR. [178]
- *Caltech-101 dataset:* is regarded to be a challenging dataset for different works in the experimental parts [158]. The dataset includes 101 classes and 1 background scene class. Every class contains between 40 and 800 images and there are a total of 9146 images in this dataset.
- *Stanford Dogs dataset [193]:* It was constructed from imagery and annotations from ImageNet for categorizing fine-grained images. The dataset features a collection of images representing 120 breeds of dogs worldwide. This dataset includes 20,580 images with 120 categories. This dataset can be used in conjunction with other datasets such as MNIST to benchmark edge learning systems [177].
- *Breast Cancer Wisconsin dataset [194]:* was developed to identify whether a person has breast cancer. It contains 569 examples (Benign: 357 and Malignant: 212), each having 30 features for the person's cell nuclei characteristics. This dataset was used by [164] to evaluate ML-targeted data poisoning attacks, which can pose a great danger to human lives in smart healthcare applications.

1) Facial recognition: Facial recognition refers to the identification or verification of a person's identity based on their face. It catches, processes, and matches different features based on the details of the person's face. Different datasets are used for such tasks, including:

- *AT&T dataset [195]:* also known as the Database of Faces. The dataset consists of 400 grayscale (92×112) images of human faces taken from 40 different individuals, with a total of 10 images for each person. For certain individuals, the images were shot at varying times, differing in lighting and facial expressions. The work of Zhang *et al.* [175] evaluated a GAN-based poisoning attack on this dataset.
- *LFW dataset [196]:* The Labeled Faces in the Wild is a collection of face photographs conceived to investigate the unconstrained face recognition task. It has over 13000 face images gathered from the web. Individual faces are labeled with their respective names. A total of 1680 of the pictured persons possess two or more separate photos. This dataset was also used for generating poisoned training samples in [161].
- *CASIA dataset [197]:* collected in the period 2007-2010 by the Institute of Automation, Chinese Academy of Sciences (CASIA), including 725 face images. The dataset was designed for Near-infrared vs. Visible light (NIR-VIS) face recognition, with 640×480 resolutions, and with 1 to 22 VIS and 5 to 50 NIR face images per person. This dataset was also used in [161].
- *VGGFace2 dataset [198]:* is composed of more than 3M images featuring 9131 famous people across a wide range of ethnic groups. All images are downloaded from Google and exhibit considerable variation regarding background, age, and lighting. Overall, the entire dataset is relatively balanced (40.7% for females and 59.3% for males). It includes 80-843 images for each individual. Fung *et al.* [147] used the VGGFace2 dataset for evaluating a poisoning Sybil's defense mechanism.
- *CelebA dataset [199]:* constructed by labeling selected images from the CelebFaces dataset, including 10000 celebrities, with 20 images for each one. And all photos are annotated with 40 face attributes and 5 key points. EdgeConnect is a two-stage adversarial scheme composed of an edge generator tracked by an image completion system, which can be used for scene generation. The scheme proposed in [200] is evaluated using the CelebA dataset.

E. Time series classification

Time series classification involves ML techniques designed to analyze several classes of labeled time series data in order to forecast or categorize which class a new dataset falls into.

- *UCR time series classification archive [201]:* was launched in 2002 and has since grown to become an extensive resource for the time series data mining field. While the original version of the archive contained 16 data sets, the archive has seen periodic updates over the years. A major expansion took place in 2015 when the

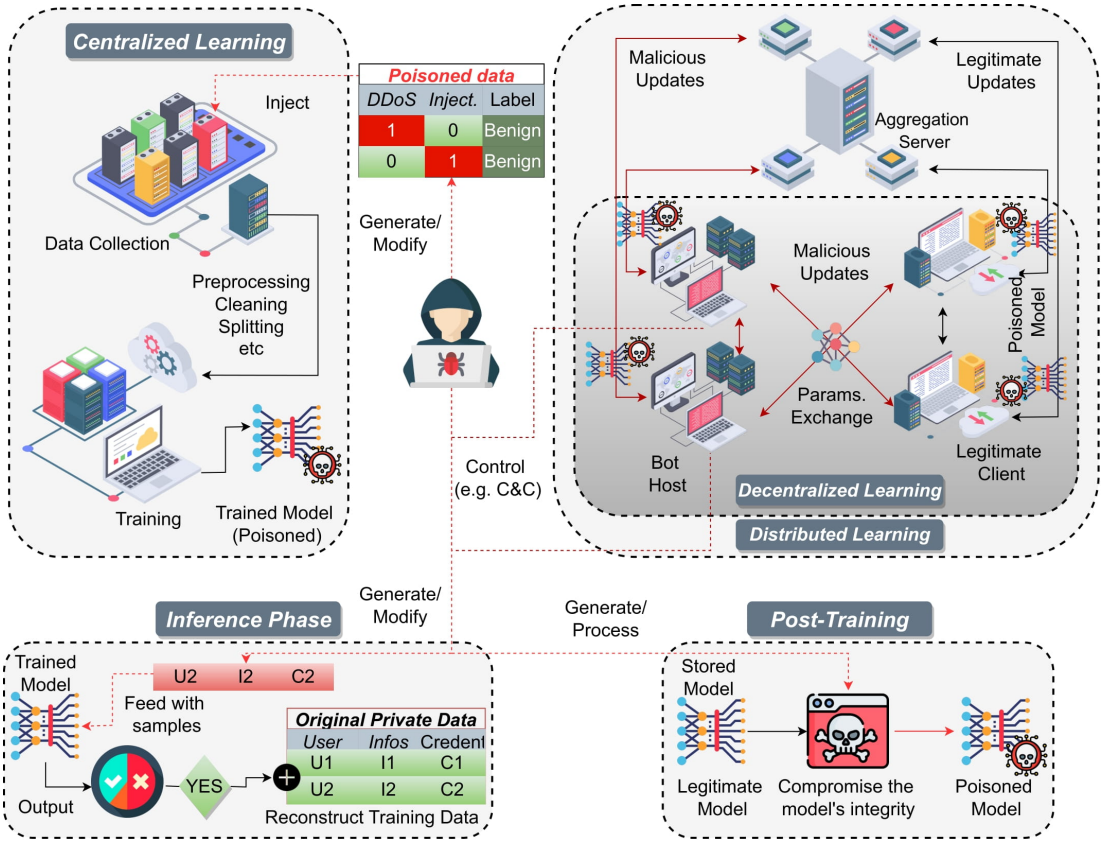


Fig. 7: Various attacks on ML classifiers under different contexts

archive expanded to 85 datasets. Various recent works on adversarial attacks and defenses against time series classification (TSC) problems have been proposed as discussed in [159].

F. Human Activity Recognition

Human Activity Recognition (HAR) has broad applicability in light of the worldwide use of interconnected sensing devices such as IoT devices, smartphones, and cameras, as well as its capability to collect data on human behavior. In addition, progress in AI has dramatically improved the extraction of profoundly hidden intelligence for precise identification and interpretation. Different datasets were proposed for HAR tasks including:

- *HARUS dataset [202]:* gathered from a group of 30 volunteers with ages ranging between 19-48 years wearing a smartphone. The data considered about their activities are split into 2 classes (dynamic and static) with each having 3 activities. It includes a total of 10299 samples with 561 attributes. The work in [160] evaluated their proposed bilevel optimization framework, which is intended to compute optimal poisoning attacks on FL.
- *Parkinson dataset [203]:* created to identify people with Parkinson's disease. It consists of 5,875 observations from a series of biomedical readings of 31 people's voices, 23 of whom have Parkinson's disease. This dataset is used for evaluating the framework in [160].

TABLE XII: Attacks on federated ML algorithms.

Attack	ML phases	Attack Goal	Attack Description
FL model extraction attack	Post-Deployment	Obtaining the model weights or AI architecture	Malicious clients creating local models that are functionally equivalent to target models of honest clients
FL evasion attack	Post-Deployment	Bypassing classifiers	Malicious clients generally will not modify the target model of honest clients but will cheat the model to generate false predictions
FL inversion attack	Post-Deployment	Private data search	Malicious clients require knowledge of labels to retrieve training data from honest clients
FL inference attack	Post-Deployment	Private data search	Identify whether a data point was used in a training set of honest clients
FL poisoning attack	Pre-Deployment	Disrupt the functioning the aggregate model	Malicious clients inject malicious data in a training set of honest clients
FL drooping attack	Pre-Deployment	Disrupt the functioning of the aggregate model	Malicious clients drop the network traffic from selected honest clients

G. Sentiment classification

Sentiment analytics is a self-learning approach that analyzes the sense of meaning in text, ranging from positive to negative. Using ML tools that are trained with instances of sentiment in text, they automatically learn to recognize sentiment with no human intervention.

- *IMDB dataset [204]:* consists of a large dataset of different reviews of movies for sentiment binary classification.

Specifically, the dataset is a collection of 50000 movie reviews and their associated binary labels (positive or negative). This dataset is also used to benchmark edge learning targeted attacks such as poisoning and backdoor, and their associated defenses [173], [174].

- *Twitter dataset [205]:* consists of a dataset for target-dependent Twitter sentiment analysis which is manually annotated. A total of 6940 tweets is collected, with 6,248 tweets for the training set and 692 tweets for the test set. The sentiment labels include negative, neutral, and positive, with class distribution being 25%, 50%, and 25%, respectively.

H. Location awareness

Location awareness designates the responsiveness of a particular system in ascertaining its position. This research field is significant for autonomous mobile operations. Available datasets for such tasks include:

- *GTSRB dataset [206]:* for (German Traffic Sign Recognition Benchmark), which is a dataset that contains more than 50000 traffic sign images with 43 classes. It was constructed in 2010 from 10 hours of footage recorded using a Prosilica GC 1380CH camera (1360×1024 pixels), during daytime drives on various types of roads in Germany. The dataset is populated with over 1700 instances of road signs, with available image resolutions varying between 15 × 15 and 222 × 193 pixels. A framework for defending against Trojans in DNNs is proposed in [180]. The proposed framework uses a moving target defense strategy given multi-dimensional training with random dimensions selection and was evaluated on this dataset.
- *Places database [207]:* is designed to provide AI systems with high-level visual perception training, including background, recognition, and prediction tasks. It includes over 10M images covering over 400 specific scenes, with anywhere from 5000 to 30000 training images per class. In addition, this dataset is used for the evaluation of the EdgeConnect system discussed above [200].

I. Text classification

Manual text classification involves analyzing the text material and making classifications accordingly, generally conducted by an annotator. Considering the fact that text is among the widest kinds of unstructured data, it is difficult and costly to parse, comprehend, arrange, and classify textual data. The purpose of automated text classification is to categorize documents into pre-specified classes, usually using Natural Language Processing (NLP) and ML algorithms. Diverse parties have provided datasets for facilitating such tasks, like:

- *DBPedia datasets [208]:* was conceived as a way to mine organized content from Wikipedia's entries. It consists of an archive of information that provides hierarchical and classification data for more than 4.23M instances covering 685 classes. Variants of this data collection are a widely used reference for evaluating NLP/text classification projects, giving a valuable baseline for multi-class/multi-label hierarchical text classification. For instance, the DBPedia-14 version which consists of 560000 instances from 14 ontological categories (including film and animal classes), was employed in [167] where the authors studied the impact of network-level adversaries on FL training and dropping attacks amplification.

J. Highlights of Peer-to-Peer Features in Edge Learning across Various Dataset Types

In all various dataset types, the peer-to-peer feature of edge learning is captured by ensuring that each device learns from its unique local data and shares the model's insights with the rest of the network. This maintains data privacy and security, utilizes the computational resources of edge devices, and enables learning from diverse and heterogeneous datasets.

- Multi-purpose security: IoT devices gather data on potential security threats, learning from it and sharing insights or models with other devices, while maintaining the privacy of raw data.
- Attack classification: In network infrastructures, devices collect and process data on different types of network attacks, updating learning models based on individual experiences, and sharing these models with other devices.
- Image classification: Devices like cameras or smartphones classify images locally, share the learned models, and keep the original images private to maintain privacy.
- Time series classification: Sensors that generate time-series data classify it locally and share their models, adhering to data privacy and decentralized learning principles.
- Human Activity Recognition (HAR): Wearable devices like smartwatches learn to recognize human activity from user data, and share their learning models, maintaining the privacy of the raw data.
- Sentiment classification: Devices (e.g., smartphones) classify sentiment on local data (such as social media posts, and messages) and share the learned models while keeping the actual data private.
- Location awareness: GPS-enabled devices learn from location data and share their models with other devices in the network, improving location awareness without revealing individual location data.
- Text classification: Each device performs text classification tasks on its unique text data, sharing the learned models to collectively improve text classification tasks, keeping the original text data private.

VII. Machine learning vulnerabilities

Machine learning systems are vulnerable to attack, which places limitations on the application of machine learning, especially in 6G–IoT Networks [215]. There are six attacks on federated ML algorithms, including, FL model extraction attack, FL evasion attack, FL inversion attack, FL inference attack, FL poisoning attack, and FL drooping attack, as presented in Table XII. For the purpose of visualizing some of these threats, Figure 7 illustrates various attacks against different ML paradigms including centralized, distributed, and decentralized learning during different model phases, including training, post-training, and inference phases. And while there

Fig. 8: Classification of machine learning vulnerabilities.

are different strategies for targeting AI models, for instance in centralized learning by poisoning the data on which the model trains directly, while in decentralized learning with poisoned updates, the result remains the same: a compromised model. In this section, we provide a comprehensive classification of threats against ML, as presented in Figure 8. In addition, a thoughtful and comprehensive review of ML vulnerabilities based on the provided classification, and related to three main aspects, namely, the knowledge already acquired by the attacker, the type of attack employed, and the final objective, which is provided in Table XIII.

A. Pre/Post-deployment attacks

Table XII lists the attacks that can be launched on federated machine learning (FL) algorithms along with their goals and descriptions. These attacks can be categorized based on the phase at which they occur - pre-deployment or post-deployment. Pre-deployment attacks target the aggregate model itself before it is deployed to honest clients. Post-deployment attacks occur after the aggregate model has been deployed to honest clients. The following six attacks compromise the privacy of the private data of honest clients [53], [160], [216], [217]:

- FL model extraction attack: This attack occurs in the post-deployment phase, where malicious clients aim to extract private data from honest clients. The attackers create local models that are functionally equivalent to the target models of honest clients to extract data.
- FL evasion attack: This attack occurs in the post-deployment phase, where malicious clients aim to bypass the classifiers of honest clients. The attackers do not modify the target model of honest clients but instead cheat the model to generate false predictions.
- FL inversion attack: This attack occurs in the post-deployment phase, where malicious clients aim to search for the private data of honest clients. The attackers require knowledge of labels to retrieve training data from honest clients.
- FL inference attack: This attack occurs in the post-deployment phase, where malicious clients aim to identify whether a data point was used in the training set of honest clients. The attackers try to extract private data from honest clients.
- FL poisoning attack: This attack occurs in the pre-deployment phase, where malicious clients aim to disrupt the functioning of the aggregate model. The attackers

TABLE XIII: Machine learning vulnerabilities.

Attack category	The attacker's knowledge	Attack type	Attack mode	The attacker's goal - Vulnerabilities	Targeted ML	Mitigation solution
Backdoor attacks	- Knowledge of the trigger (a small patch)	RNN backdoor attacks regarding text classification	CL, FL, DL	The adversary takes control of the global model training operation	Text classification	[146]
	- Knowledge of some poisoned data based on the trigger	Imperceptible backdoor pattern	CL, FL, DL	The adversary poisons the training set with a small ensemble of pictures sourced from a source class (or classes)	Image classification	[149]
	-Inject maliciously the poisoned data to the victim to train a deep model with	Distributed backdoor attack	CL, FL, DL	Backdoor attack performed by several different collaborative attackers	Image classification	[154]
	- The trigger is stored as a secret by the attacker and only then exposed at the time of the test	Pixel-space backdoor attack	CL, FL, DL	Manipulate the pixel data values of the image to poison the images in the training dataset	Image classification	[209]
	- The attacker poison training examples and change their labels	Clean label backdoor attack	CL, FL, DL	Requiring zero-knowledge of the target model	Image classification	[178]
Combined attacks	Attack through jointly training neural network classifiers	Synergetic attack	CL, FL, DL	Combines adversarial examples and Trojan backdoor	Image classification	[210], [211]
Adversarial examples	Black-box attack: the configuration of the target ML models is unavailable to adversaries	Black-box adversarial attack	CL, FL, DL	The adversary has restricted information about the model but certainly does not have knowledge of the model's parameters	- Time series classification - Speech Recognition - Video Recognition	- [159] - [163] - [168]
	- White-box attack: the adversary has access to all the information of the target ML model.	White-box adversarial attack	CL, FL, DL	White-box attacks require full and complete knowledge of the model being targeted, including its training method, architecture and parameter values	- Text classification - Attack classification	[169] - [170]
	- Grey-box attack: the adversary has some knowledge about the target ML model	Grey-box adversarial attack	CL, FL, DL	Requires partial knowledge instead of full knowledge	Face Recognition	[212]
Poisoning attacks	- There are two distinct attack scenarios black-box and white-box attacks	Federated poisoning attack	FL, DL	Falsifying machine learning training data to generate unwanted results in a decentralized mode. The adversary's goal is to modify the parameters learned and manipulate the training data on their own device	- Image classification - Sentiment classification - Attack classification - Human Activity Recognition	- [157], [165], [171] - [173] - [162] - [157] - [160]
		Intrusion poisoning attack	CL, FL, DL	The application of poisoning attacks in the intrusion detection datasets	Attack classification	[213]
	- White-box attacks: the attacker is assumed to know the trained parameters, the learning algorithm, the feature values, and the training set	Label flipping attack	CL, FL, DL	Attackers can flip the labels of some samples from one class to another, for example from attack to benign	- Sentiment classification - Attack classification	- [174] - [172]
		Poisonous label attack	CL, FL, DL	Injects fake images with the poisonous label in the training dataset than modifying directly the label of the images	- Image classification	[155]
	- Black-box attacks, the attacker is assumed to know the learning algorithm and feature set, and collect a substitute data set but has no knowledge of the training set as well as the trained parameters	Clean-label poisoning attack	CL, FL, DL	The poison patterns are generated by inserting undetectable changes that will cause the model to misbehave in reaction to particular target inputs	- Transfer learning - Attack classification	- [151] - [172]
		Data poisoning attack	CL, FL, DL	The attacker attempts to infect the training data by inserting well-formed samples to impose a harmful model on the learner	- Image classification - Electricity theft detection	- [177] - [179]
	- In FL settings, the attacker's ultimate goal is to modify the source distribution of the raw data based on the corruption of the aggregate model by uploading the parameters of the poisoned local model	Generative poisoning attack	CL, FL, DL	Deploying a GAN model from the attacker's side to impersonate other participants' training dataset patterns	- Image classification - Face recognition	- [153] - [161]
		Over-the-air spectrum poisoning attack	CL, FL, DL	Confuse the sender into making poor transmission decisions (i.e., an evasion attack) or to tamper with the sender's recycling process (i.e., a causal attack)	- Communication systems	[214]
Sybil attacks	- It assumed that the attackers exploit the Sybils to initiate poisoning attacks against federated learning	Sybil-based poisoning attack	FL, DL	Sybils conduct federated learning poisoning attacks by delivering status updates that lead the distributed model to a poisoned common target.	- Face recognition - Image classification	- [147] - [10]
		Sybil-based collusion attack	FL, DL	To complete local poisoning training, the attacker employs the label flipping scheme	- Image classification	[166]
Byzantine attacks	The attacker's knowledge can be classified into three levels: (i) no knowledge, (ii) partial knowledge, and (iii) complete knowledge	Local model poisoning attack	FL, DL	Affect the integrity of the learning phase in the training process	- Image classification	[164]
Dropping Attack	The adversary drops contributions from the clients in every round	Targeted dropping attack	FL, DL	By monitoring more rounds of the FL system, the adversary can drop the network traffic from selected clients	- Text classification	[167]

CL: Centralized Learning, FL: Federated Learning, DL: Distributed Learning

inject malicious data into the training set of honest clients to manipulate the model.

- FL drooping attack: This attack occurs in the pre-deployment phase, where malicious clients aim to disrupt the functioning of the aggregate model. The attackers

drop the network traffic from selected honest clients to manipulate the model.

B. Backdoor attacks

Backdoor attacks involve the manipulation of training data or models to trigger specific behaviors in the model during the testing phase. Different types of backdoor attacks are presented, including RNN backdoor attacks, imperceptible backdoor pattern attacks, distributed backdoor attacks, pixel-space backdoor attacks, and clean-label backdoor attacks.

1) RNN backdoor attacks: Backdoor attacks in deep neural networks are based on two assumptions, namely, the adversary takes control of the global model training operation and the second assumption is that the adversary only has control over particular training data. Chen *et al.* [146] proposed a mitigating system against backdoor attacks in the text classification based on Backdoor Keyword Identification. The proposed system can identify the poisoning patterns in the training dataset with no trust data and no knowledge of the backdoor release. The following text classification datasets are used in the performance evaluation with the LSTM models, Reuters-21578 dataset, 20 newsgroups, DBpedia ontology, and IMDB. The results show good results in terms of identification precision and recall of poisoning samples. To remove poisoning data, the authors proposed the following formula to filter the keywords in a dictionary and detect the suspicious keyword, which is most susceptible to be a backdoor keyword:

$$Ident_{backdoor} = \overline{f(k)} \cdot log_{10} gen\_sam \cdot log_{10} \frac{x}{gen\_sam} \quad (1)$$

$\overline{f(k)}$ is the average score for the keyword k in the dictionary. The score could be based on various metrics such as the frequency of the keyword, the similarity between the keyword and other words in the dictionary, or other measures. $log_{10} gen\_sam$ uses a logarithmic function of the number of samples that produce the keyword gen_sam. The purpose is to adjust the weight of the keyword based on the number of samples in which it appears. $log_{10} \frac{x}{gen\_sam}$ represents the logarithm of the reciprocal of frequencies, which is used to normalize the weight of the keyword. The x value is a fixed parameter that is higher than 0. By combining these components, the formula calculates an identifier $Ident_{backdoor}$ for each keyword, which can be used to detect suspicious keywords that are more likely to be backdoor keywords, i.e., the higher the value of $Ident_{backdoor}$, the more suspicious the keyword.

2) Imperceptible backdoor pattern: In image classification, the attacker poisons the training set with a small ensemble of pictures sourced from a source class (or classes), incorporated with a backdoor feature, and then tagged to a target class. Xiang *et al.* [149] proposed a mitigation technique based on imperceptible backdoor patterns. The proposed technique can identify if the training set is poisoned and precisely recognizes the target class and training images in which the backdoor pattern is integrated. At the end of the process, the proposed technique performs reverse engineering to provide an estimate of this backdoor pattern used by the attacker. The CIFAR-10 dataset is used as a benchmark experiment, and the results show a reduced rate compared to state-of-the-art to no more than 4.9%.

The poisoned dataset used for training the victim classifier consists of a combination of the $Dataset_{backdoor}$ and the benign $Dataset_{train}$. The $Dataset_{backdoor}$ is a set of examples that have been intentionally modified with a backdoor pattern to trick the classifier into predicting a specific target class when presented with an image from a particular source class. The $Dataset_{backdoor}$ is defined as follows:

$$Dataset_{backdoor} = \{(f(x;pattern^*), y) | x \sim P_{Source^*}, \; y = Class^*\} \quad (2)$$

$f(x;;pattern^*)$ is the embedding function that maps an image x to a feature vector, and $pattern^*$ is the backdoor pattern that is added to the image. The resulting image $f(x;;pattern^*)$ is then labeled as $Class^*$, which is the target class that the backdoor is designed to trigger. The source class(es) P_{Source^*} determines which images the backdoor is applied to. If P_{Source^*} contains multiple classes, the backdoor can be triggered by images from any of those classes. The poisoned dataset is used to train the victim classifier, which learns to classify images from both the benign dataset and the poisoned dataset. However, since the poisoned dataset contains examples with backdoors, the victim classifier can be manipulated to produce incorrect predictions on images that contain the backdoor pattern.

3) Distributed backdoor attack: Backdoor attacks performed by several different collaborative attackers, i.e. distributed backdoor attacks, can reach very successful rates and are very difficult to be detected. Based on a dynamic norm clipping approach, Guo *et al.* [154] proposed a security system to defend against distributed backdoor attacks in federated learning. The proposed system is evaluated with the following four public datasets: MNIST, FMNIST, CIFAR-10, and Tiny Imagenet. The results show that the proposed system can reduce the attack access rate by 84.23% compared to the state of the art. With the distributed backdoor attack, there are attackers who constitute a group with identical malicious purposes. More precisely, all attackers conduct the backdoor attack on their local models independently using their specific local backdoor trigger $Trig_i$ and the same target label $Label_{target}$. The attacker's corresponding goal i is defined as follows :

$$G_i(w) = \sum_{j \in Data^i_{infected}} \left[f(w;;x^i_j + Trig_i, Label_{target}) \right] + \sum_{j \in Data^i_{uninfected}} \left[\left[f(w;;x^i_j + y^i_j) \right] \right] \quad (3)$$

Where w represents the global model parameters, $Data^i_{infected}$ is the infected image dataset of attacker i, $Data^i_{uninfected}$ is the uninfected image dataset of attacker i, x^i_j represents input from the local dataset, and y^i_j represent the corresponding label sampled from the local dataset. The

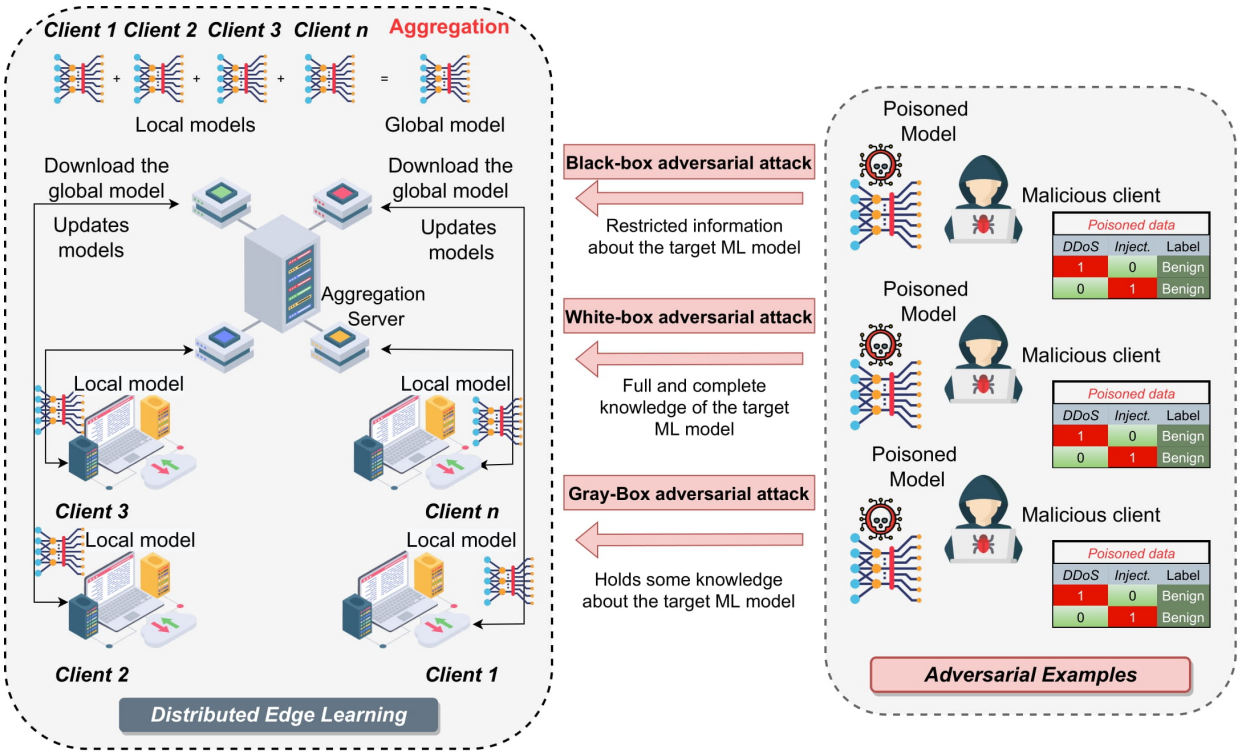


Fig. 9: Adversarial examples in edge learning.

objective of each attacker is to maximize the misclassification rate of the global model on a specific target label $Label_{target}$ when presented with inputs that contain a specific backdoor trigger $Trig_i$. The goal $G_i(w)$ of attacker i is defined as the sum of two terms: the first term represents the misclassification rate of the infected images from attacker i on the target label $Label_{target}$, while the second term represents the misclassification rate of the uninfected images from attacker i on their true labels. The function $f(w;x,y)$ represents the output of the global model with parameters w on input x and label y. The double semicolon notation $w;;x_j^i + Trig_i$ indicates that the backdoor trigger $Trig_i$ is added to the input x_j^i before it is fed into the global model.

4) Clean label backdoor attack: Ning *et al.* [178] proposed a black box clean label backdoor attack, which requires zero knowledge of the target model. In order to detect and eliminate, the authors propose the following two methods: 1) Supervised poison sample detection and 2) Unsupervised poison sample detection. For Backdoor injection, the adversary summarized another dataset by combining the clean database with a small part of the poisoned dataset as follows:

$$Dataset_{new} = Dataset_{clean} \cup Dataset_{poisoned} \quad (4)$$

The adversary uses this dataset during the training process and verifies that other properties are not influenced by a combined dataset. The backdoor injection attack involves adding malicious data to the training dataset, which can be used to exploit the model during inference. In this case, the adversary has combined a clean dataset with a small part of the poisoned dataset to create a new dataset for training. The poisoned dataset contains inputs and labels that have been specifically chosen to trigger a backdoor in the model. To train the model using this dataset, the adversary minimizes a loss function that includes both clean and poisoned data. The goal is to find the parameters of the model that minimize the total loss on both datasets. The loss function is typically based on cross-entropy, which measures the difference between the predicted probabilities and the true labels.

The availability of a poisoned dataset gives the following loss function:

$$F_l = min_{model} \sum_{i=0}^{n} Loss(model, (x^i, y^j)) + \sum_{j=0}^{m} Loss(model, (x^{i'}, y^{j'})) \quad (5)$$

$model$ represents the parameters of the model. $Loss(model, (x^i, y^j))$ represents the loss incurred by the model when making a prediction for input x^i with label y^j from the poisoned dataset. $Loss(model, (x^{i'}, y^{j'}))$ represents the loss incurred by the model when making a prediction for input $x^{i'}$ with label $y^{j'}$ from the clean dataset. n is the size of the poisoned dataset. m is the size of the clean dataset.

5) Pixel-space backdoor attack: Pixel-space backdoor attacks are a type of adversarial attack where an attacker modifies the pixel values of an image in a way that is imperceptible to the human eye but causes misclassification by the target model. Arshad et al.'s pixel-space backdoor attack [209] is a type of data poisoning attack, where the attacker

injects malicious samples into the training dataset to cause the model to learn a backdoor trigger. In this attack, the attacker flips certain bits of the pixel values of the training images, which can be done without significantly changing the appearance of the image. These modified images are then inserted into the training dataset, along with their original labels. During training, the model learns to associate the backdoor trigger with the target label, allowing the attacker to trigger the attack by adding the trigger to any input image. Arshad *et al.* evaluated their backdoor attack strategy on the MNIST digit dataset, which consists of 28x28 Grey-scale images of handwritten digits. They found that their attack was successful in 90% of the cases, meaning that the target model misclassified the poisoned images with the intended target label. Therefore, it's important to note that the success of a backdoor attack depends on several factors, including the size and quality of the training dataset, the choice of the backdoor trigger, and the strength of the attacker's evasion techniques. Therefore, while Arshad et al.'s attack achieved a high success rate on the MNIST dataset, it may not be as effective on other datasets or against more robust models with stronger defenses against adversarial attacks.

C. Adversarial examples

Adversarial examples refer to inputs specifically crafted to deceive machine learning models. Different types of adversarial attacks are presented, including black-box attacks, white-box attacks, and Grey-box attacks. Akhtar and Mian [232] proposed a classification of adversarial examples on deep learning into two categories: 1) Attacks for Classification and 2) Attacks Beyond Classification/Recognition. The Attacks for Classification include miscellaneous attacks, adversarial transformation networks (ATNs), Houdini, Upset and ANGRI, Universal Adversarial Perturbations, Deepfool, Carlini and Wagner Attacks (C&W), One Pixel Attack, Jacobian-Based Saliency Map Attack (JSMA), Basic & Least-Likely-Class Iterative Methods, Fast Gradient Sign Method (FGSM), and Box-Constrained L-BFGS. Attacks Beyond Classification/Recognition include Attacks on Face Attributes, Attacks on Semantic Segmentation & Object Detection, Attacks on Deep Reinforcement Learning, Attacks on Recurrent Neural Networks, and Attacks on Autoencoders and Generative Models. Table XIV presents the threat models of adversarial examples generation. Figure 9 presents the adversarial examples in edge learning. Therefore, in our work, we propose a classification of adversarial examples into three categories: 1) White-box adversarial attacks, 2) Black-box adversarial attacks, and 3) Grey-Box adversarial attacks.

1) White-box adversarial attacks: White-box attacks require full and complete knowledge of the model being targeted, including its training method, architecture, parameter values, and, in most cases, its data training. To manipulate discrete text structure at its one-hot representation, Ebrahimi *et al.* [169] propose white-box adversarial examples, named HotFlip, which is applied for text classification. The HotFlip system uses the gradient with respect to a one-hot input representation and can produce conflicting patterns with character substitutions ("flips"). The HotFlip also provides access to add and remove transactions by displaying them as character substitution sequences. The performance evaluation with AG's news dataset and CharCNN-LSTM architecture shows that the adversary selects the flip transaction about 80% of the time and prioritizes suppression over insertion by two-to-one. For fooling deep neural networks, Wang *et al.* [170] propose white-box attacks based on the integrated gradient. The proposed attacks are evaluated with the following three datasets: CIFAR-10, MNIST, and NSL-KDD. One of the most popular algorithms for generating white-box adversarial examples is the Fast Gradient Sign Method (FGSM) introduced by Goodfellow *et al.* in 2014 [226]. The FGSM generates adversarial examples by taking the gradient of the loss function with respect to the input and adding a perturbation to the input in the direction of the gradient. The formula for generating adversarial examples using FGSM is as follows:

$$x_{adv} = x + \epsilon \cdot sign(\nabla_x J(\theta, x, y)) \tag{6}$$

Where x is the input, y is the true label, J is the loss function, θ is the model parameters, ϵ is a small constant that controls the magnitude of the perturbation, and $sign$ is the sign function.

2) Black-box adversarial attacks: Black-box attacks provide a trained model with conflicting examples (during testing) which are produced with no knowledge of the trained model. In particular cases, it is supposed that the adversary has restricted information about the model but certainly does not have knowledge of the model's parameters. Yang *et al.* [159] propose a black-box method called TSadv, which attacks deep neural networks on time series. For solving the constrained optimization problem, the TSadv method uses a gradient-free attack strategy. The performance evaluation with the University of California Riverside (UCR) time series dataset shows good results in terms of success rate, the average number of iterations, and mean squared error. Therefore, Biolková and Nguyen [163] propose a black-box adversarial attack in speech recognition. The proposed attack uses a neural predictor that approximates the length of the decision boundary which results in a wrong audio signal transcription. The experiments on the LibriSpeech dataset show that the proposed attack can achieve a better success rate compared to BayesOpt [233] and SignOpt [234].One of the most popular algorithms for generating black-box adversarial examples is the Boundary Attack introduced by Brendel and Bethge in 2019 [235]. The Boundary Attack generates adversarial examples by iteratively exploring the decision boundary of the model. The formula for generating adversarial examples using Boundary Attack is as follows:

$$x_{adv} = x + \alpha \cdot \frac{g}{|g|_p} \tag{7}$$

Where x is the input, α is a small constant that controls the step size, g is the gradient of the loss function with respect to the input, and $|g|_p$ is the l_p norm of the gradient.

3) Grey-box adversarial attacks: Grey-box adversarial attacks involve generating adversarial examples with limited knowledge of the targeted model. The attacker has access

TABLE XIV: Threat models of adversarial examples generation.

Attacks examples	Attacks for the specific tasks	White/black/Grey box	Adversarial Example Generation	Methodes
Miscellaneous attacks	Classification	Feature-space white-box attacks	The XGBoost can be used as the adversarial feature extraction model	Miscellaneous malware detectors based on OpCode n-gram features [218]
Adversarial transformation networks	Classification	Trained in a white-box or black-box manner	Adversarial Autoencoding (AAE) and Perturbation ATN (P-ATN)	Deterministic/stochastic defense models [219]
Fooling gradient-based attacks	Classification	Black/Grey-box attack	Generating adversarial audio files	Fooling Deep structured prediction models [220]
Universal perturbations and antagonistic network	Classification	Black/Grey-box attack	Generating rogue images	Breaking high-performance image classifiers [221]
Carlini and Wagner Attacks (CW)	Classification	Black/Grey-box attack	Generating adversarial examples based on distance metrics	Defensive distillation [222]
One Pixel Attack	Classification	Semiblack-Box Attack	Generating adversarial examples using differential evolution	Covariance matrix adaptation evolution strategy [223]
Jacobian-Based Saliency Map Attack (JSMA)	Classification	White-box attack	Generating adversarial samples using the mapping between inputs and outputs of deep learning	Predictive measure of distance [224]
Basic Least-Likely-Class Iterative Methods	Classification	White-Box Attack	Generating adversarial images based on the linearization of the cost function	Iterative least likely method [225]
Fast Gradient Sign Method (FGSM)	Classification	White-Box Attack	Generating adversarial examples using a family of fast methods	Explaining and harnessing adversarial examples [226]
L-BFGS attack	Classification	White-Box Attack	Generating adversarial examples using a box-constrained L-BFGS algorithm	Intriguing properties of neural networks [227]
Adversarial vulnerability of facial attributes	Classification/Recognition	White-Box Attack	Generating natural adversarial images	Fast flipping attribute technique [228]
Attacks on Semantic Segmentation object Detection	Image Segmentation	Black/Grey/White-box attack	Adversarial target generation	Universal adversarial perturbations [229]
Strategically-timed attack	Classification/Recognition	Black/Grey/White-box attack	Combining a planning algorithm and a generative model	Adversarial attack on deep reinforcement learning systems [230]
Attack on Recurrent Neural Networks	Classification/Recognition	Black/Grey/White-box attack	Creating adversarial sequences	Fast gradient sign method and Forward derivative method [231]

to some, but not all, of the model's internal information. For example, the attacker may know the architecture of the model, but not the weights or biases. The goal of a grey-box attack is to find an adversarial example that can deceive the targeted model into producing incorrect output, even though the attacker does not have full knowledge of the model's internal workings. This is often done by leveraging the gradients of the model with respect to the input, which can be computed even if the attacker does not know the exact values of the model's parameters [236]. Grey-box attacks are particularly relevant in real-world scenarios, where attackers may not have access to the full details of a deployed model, but may still be able to mount effective attacks. Defenses against grey-box attacks typically involve techniques such as input sanitization [237] or model hardening, which aim to make it harder for attackers to find effective adversarial examples. One of the most popular algorithms for generating Grey-box adversarial examples is the Transferability Attack introduced by Papernot *et al.* in 2016 [238]. The Transferability Attack generates adversarial examples by exploiting the transferability property of adversarial examples, which means that adversarial examples generated for one model can also fool another model. The formula for generating adversarial examples using Transferability Attack is as follows:

$$x_{adv} = x + \epsilon \cdot sign(\nabla_x J(\theta_{target}, x, y)) \tag{8}$$

This equation represents the process of generating an adversarial example for a given input image x, with respect to a target class θ_{target} and a true label y, by adding a small perturbation ϵ to the original image x in the direction of the sign of the gradient of the target class probability $J(\theta_{target}, x, y)$ with respect to the input image x.

In most cases, black-box attacks are viewed as more difficult since the attacker has less information to operate with. Nevertheless, Grey box attacks can still be successful if the offensive party has sufficient knowledge of the pattern to produce successful adversarial examples.

D. Combined attacks

Combined attacks are a combination of adversarial examples and Trojan backdoors, which can be used to bypass defenses in the model.

1) Synergetic attack: A synergetic attack consists of a combination of backdoor and adversarial examples against neural network classifiers. Liu *et al.* [158] proposed a synergetic attack, named AdvTrojan. The AdvTrojan is enabled based only on the model being infected through a backdoor during training, and the entries are thoughtfully disrupted. The AdvTrojan uses two stages that operate in parallel to progressively relocate the targeted input through the decision-making boundary to the adversary's objective class. In the initial stage, the model is injected with a Trojan backdoor at the time of training. During inference, the Trojan backdoor is enabled by incrementing the relevant target inputs using the pre-specified Trojan trigger. In the second stage, the targeted

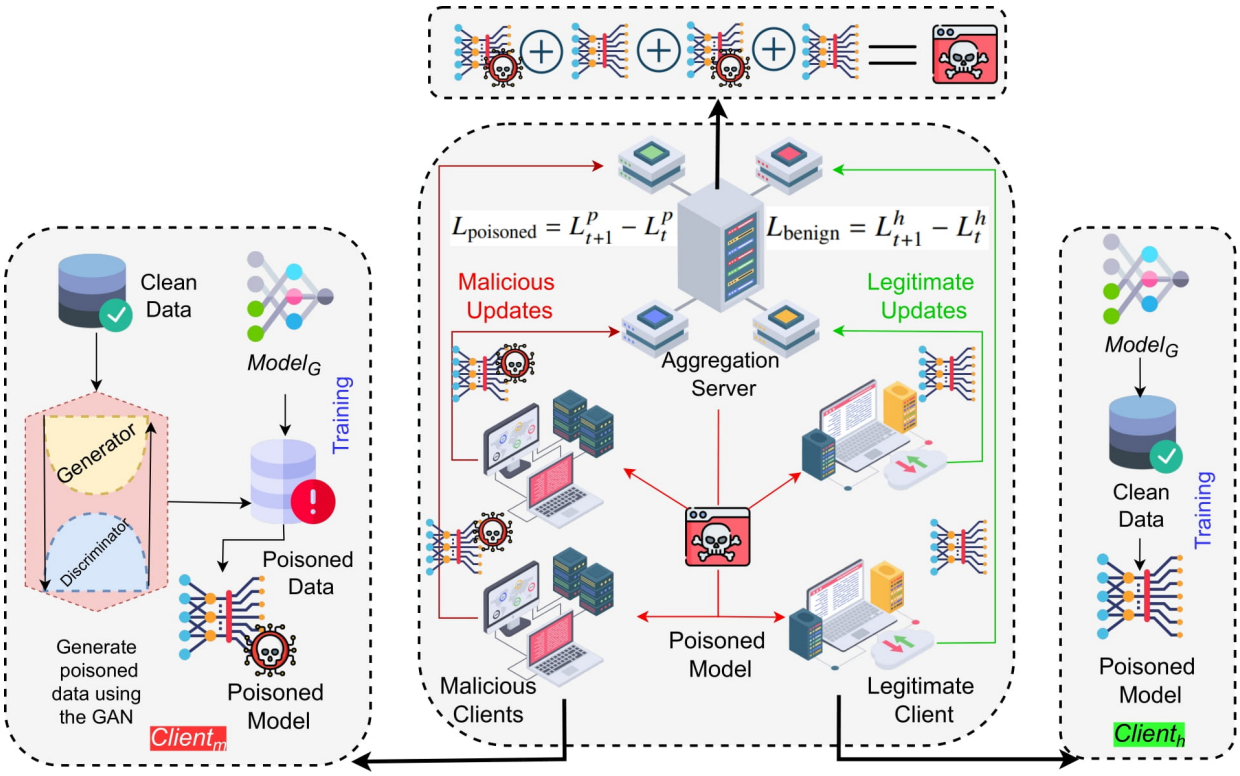


Fig. 10: FL model poisoning attack using generated poisoned samples

entry is completed by a prudent mixture of some adversarial disruption and the Trojan horse trigger. In the second stage, the targeted entry is completed by a prudent mixture of the Trojan horse trigger and some adversarial disruption. The adversary disruption enhances the Trojan horse trigger to exchange the entry label in the class targeted by the adversary. In practical terms, the Trojan horse trigger forwards the entry to an artificial position in the space of the entry closest to the boundary of the model's decision. Afterward, the adversarial disruption makes the terminal push by shifting the forwarded example through the boundary of the decision, initiating the pre-established backdoor. The combined attacks can be defined as follows:

$$Classifier_{\theta\uparrow}(x) = \begin{cases} y_{target}, & \text{if } x \text{ contains Trojan trigger } t \\ Classifier_{\theta\downarrow}(x), & \text{otherwise} \end{cases} \tag{9}$$

Where y_{target} is the attacker's target, $Classifier_{\theta\uparrow}(x)$ is the classifier with normal behavior, $Classifier_{\theta\downarrow}(x)$ is the Trojan-infected classifier, and x refers to the general input, that can be benign or malicious.

E. Poisoning attacks

Poisoning attacks involve the manipulation of the training data to produce a biased or inaccurate model. Different types of poisoning attacks are presented, including federated poisoning attacks, intrusion poisoning attacks, label flipping attacks, poisonous label attacks, clean-label poisoning attacks, data poisoning attacks, and generative poisoning attacks [160], [217]. An attacker will penetrate a machine learning system and inject false or incorrect information into the database. Over time, as the algorithm learns from this falsified data, it will produce unwanted and potentially damaging outcomes [239], [240]. However, data poisoning attacks can be divided into two major classes: those attacking availability and others attacking integrity. Availability attacks are frequently not particularly advanced but extensive, inserting at least as many bad data items as they can into a database. Once an attack is successful, the machine learning algorithm will be totally incorrect. Attacks against the integrity of machine learning are more complicated and could be more dangerous.

1) Federated poisoning attack: Federated learning uses machine learning in a decentralized mode that does not directly access the private data of clients. However, Federated learning suffers from many issues, such as adversarial machine learning-related security attacks, high communications costs between clients and the server, and high accuracy [10], [147], [148], [152], [157], [160], [161], [165]. To address these issues, Tabatabai *et al.* [171] present a federated learning algorithm based on evolutionary methods. The proposed clustering algorithm collects clients in many clusters, each with a selected model at random to exploit the capabilities of individual models. Then, the clusters are built in a repeated procedure with the worst cluster being dropped at every repetition process until there is only one cluster available. During all iterations, certain clients are kicked out of the clusters for corrupted data usage or poor overall performance. The winning clients are operated in the upcoming iteration. The cluster that remains with the surviving clients is then utilized for the best FL model training. The Fast Gradient Sign Method (FGSM) is

adopted for producing adversarial examples in order to create a federated poisoning attack.

To mitigate data poisoning attacks, Doku and Rawat [173] proposed a security system in a federated learning mode. To generate the total error produced by a training dataset, the proposed system uses a Support Vector Machine. The performance evaluation with the IMDB review sentiment dataset shows that the proposed system achieved an accuracy score of 0.72. Uprety and Rawat [156] proposed a technique for mitigating poisoning attacks that focuses on the reputation of the nodes involved in the process of training. The average reputation score of every client is computed with the method of probability distribution beta. The benchmark MNIST dataset is used in the performance evaluation, and the security analysis demonstrates that the classification model accuracy improved from 88% to 93% over 100 rounds of communication after eliminating the attacker nodes by the aggregation servers. The generation of poison attacks in FL, as illustrated in Figure 10, can be defined by the following steps [175]:

- Step 1: Send the global model $Model_G$ to honest clients $Client_h$ and malicious clients $Client_m$.
- Step 2: The malicious clients $Client_m$ generate samples of targeted class.
- Step 3: The malicious clients $Client_m$ assign wrong label to generated samples.
- Step 4: The malicious clients $Client_m$insert poison data to the local dataset $Dataset_{Local}$.
- Step 5: The malicious clients $Client_m$ calculate the poisoned update $L_{\text{poisoned}} = L^p_{t+1} - L^p_t$.
- Step 6: The honest clients $Client_h$ calculate the benign update $L_{\text{benign}} = L^h_{t+1} - L^h_t$.
- Step 7: Update the local update L_{poisoned} and L_{benign} to the server.

2) Intrusion poisoning attack: This type of attack consists of the application of poisoning attacks in the intrusion detection datasets Venkatesan *et al.* [213]. An intrusion detection system (IDS) is a network monitoring system that detects anomalous network behavior and issues alerts when anomalous activity is detected. An IDS is a computer program that analyzes a system or network for malicious activity or policy violations. However, in order to disturb the IDS systems, the attackers use intrusion poisoning attacks by implementing the poisoning attacks in the IDS dataset. The following datasets can be used for evaluating the performance of ML-based-IDS: Edge-IIoT dataset [56] and X-IIoTID dataset [241].

3) Label flipping attack: This attack is conducted by flipping the labels of specific data samples from one class (the source class) to another (the target class), as shown in Figure 11. Jebreel *et al.* [174] introduced a new security system against the label-flipping attack in federated learning that extracts dynamically the target and source class potential gradients of local peer updates, performs an application of a clustering method on these gradients, and then evaluates the resulting groups to extract potential bad updates prior to the aggregation of the model. The following three data sets are used in the experiments: MNIST, CIFAR10, and IMDB. The results of three models CNN, ResNet18, and BiLSTM show that the proposed defense system can offer a lower attack success rate, higher source class accuracy, and lower test error. The following steps define the generation of label-flipping attacks in FL:

- Step 1: The malicious clients poison their local training data by flipping the labels of training examples from a source class $Class_{source}$ to a target class $Class_{target}$ without changing the features of the input data.
- Step 2: The honest and malicious clients train their local models using the same hyper-parameters, model architecture, optimization algorithm, and loss function sent by the server.
- Step 3: The honest and malicious clients train their local models (i.e., bad updates from malicious clients and good updates from honest clients) to the server.

4) Poisonous label attack: Compared to the poisonous attack under the white box requirement, the poisonous label attack is the black box, which is able to enhance the misclassification error under a more constrained and workable condition of the poisonous label attack. The poisonous label attack involves three parts: its capability, the attacker's knowledge, and the attacker's goal. The attackers' goals are identified by three components: error specificity, attack specificity, and security violation. Liu *et al.* [155] propose the poisonous label attack, which injects fake images with the poisonous label in the training dataset than modifying directly the label of the images. The knowledge budget for the poisonous label attack is defined as:

$$Budget_{know} = \frac{N(w_{know})}{N(w)} + \frac{N(Dataset_{know})}{N(Dataset_{train})} \tag{10}$$

Where $N(w)$ is the number of parameters of the victim model, $N(Dataset_{train})$ is the number of samples in the training dataset, $N(Dataset_{know})$ is the number of samples known by the malicious clients, $N(w_{know})$ is the number of parameters known by the malicious clients.

5) Clean-label poisoning attack: This is a class of poisoning attacks where the attacker has no control over the labeling procedure. In this particular threat model, the poison patterns are generated by inserting undetectable changes that will cause the model to misbehave in reaction to particular target inputs. Aghakhani *et al.* [151] propose a scalable clean-label poisoning attack, which enhances the current state-of-the-art attack success rate by 26.75% in transfer learning while improving the speed of attack by a factor of 12. This attack is based on injecting poisoned data into the training dataset in such a way that the model learns to classify the target input incorrectly. Based on the method that is based only on first-order information, Zheng *et al.* [150] proposed a clean-label data poisoning attack based on first-order information. The proposed attack is evaluated with the CIFAR-10 dataset on multiple network architectures, namely, ResNet, VGG, and ConvNet. The attack uses a Convex Polytope (CP) to solve an optimization problem, which involves minimizing the distance between the target feature vector and the poisoned samples set. The coefficients of the poisoned samples set are specified by V^i_j. The goal of the attack is to find a set of poisoned samples

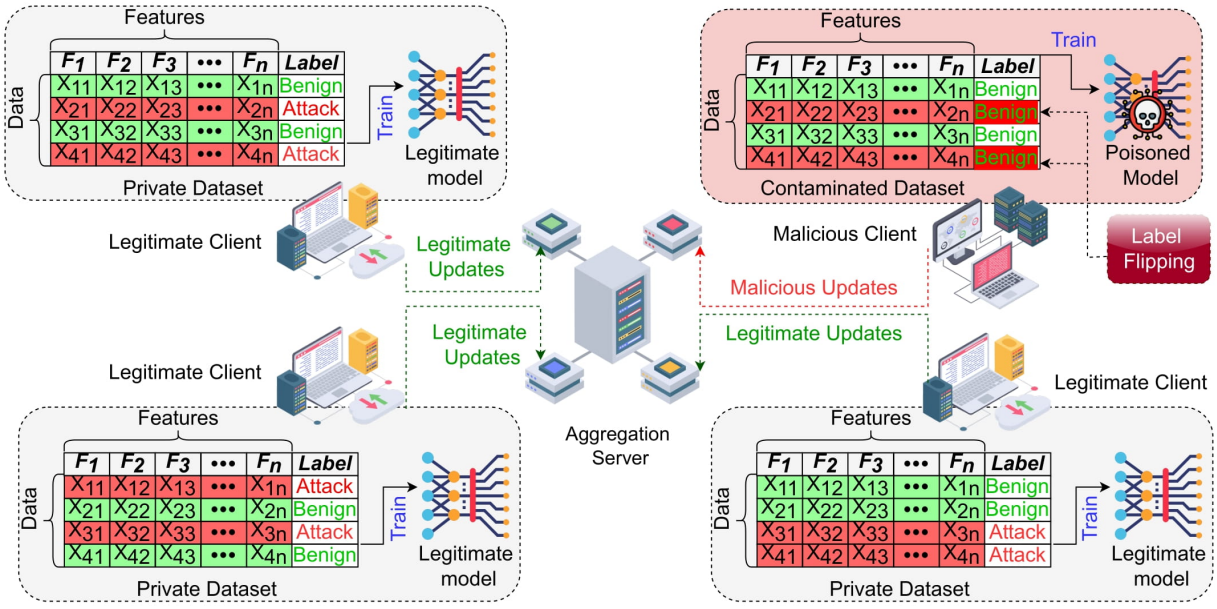


Fig. 11: FL-targeted label flipping attack

that can mislead the model when it encounters specific inputs. The CP adopted solves the following optimization problem:

$$\frac{1}{2n}\sum_{i=1}^{n}\frac{\left\|\alpha^i\left(x_t\right)-\sum_{j=1}^{m}V_j^i\alpha^i\left(x_p^j\right)\right\|^2}{\left\|\alpha^i\left(x_t\right)\right\|^2} \tag{11}$$

where x_p^j is a poison samples set, α^i is the target feature vector, and V_j^i specifies the j-th poison's coefficient. The objective function is a convex combination of squared Euclidean distances, where each distance is weighted by a positive coefficient. The denominator of each term ensures that the objective function is normalized by the length of the target feature vector $\alpha^i(x_t)$, which makes it invariant to the magnitude of the feature vector. The minimization problem seeks to find the value of x_t that minimizes the objective function. Because the objective function is convex, any local minimum is also a global minimum. Therefore, any optimization algorithm that converges to a local minimum will also converge to the global minimum.

6) Data poisoning attack: Data poisoning attacks against machine learning models have emerged as an influential research area of adversarial machine learning. This form of attack occurs during the training phase of machine learning models. The attacker attempts to infect the training data by inserting well-formed samples to impose a harmful model on the learner. Zhang *et al.* [242] proposed a data poisoning attack scheme, called IMF. The IMF scheme adopts the bi-level optimization problem to formulate the poisoning attack. Specifically, the upper-level problem estimates the strength of the actual fabricated poisoning sample, while the lower-level problem implements the model parameters based on these poisoning samples. The iterative learning algorithm is implemented by the IMF scheme to solve the bi-level optimization problem. The performance evaluation on two datasets, namely, the Stanford Dogs dataset and the Osteoporotic Fracture dataset, shows that the MFI scheme has the potential to reshape interpretations of the target samples successfully. However, Takiddin *et al.* [179] present a sequential ensemble detector against data poisoning attacks, which is based on feed-forward neural networks, gated recurrent units (GRUs), and a deep auto-encoder with attention (AEA).

7) Generative poisoning attack: This type consists of deploying a generative adversarial network (GAN) model from the attacker's side to impersonate other participants' training dataset patterns, which can initiate the poisoning attack successfully under an assumption of a more feasible threat. Zhang *et al.* [153] propose a novel poisoning attack, named PoisonGAN, which can be applied in federated learning settings. Specifically, the PoisonGAN attack is based on generative adversarial networks as presented in Figure 10. Chen *et al.* [161] propose a novel adversarial network, named DeepPoison, which is based on one generator and two discriminators. More precisely, the generator extracts hidden features from the target class automatically and integrates them into benign training patterns. The one discriminator monitors the poisoning perturbation ratio. The second discriminator operates as a target model to witness the poisoning impacts. The idea of GAN is to build an adversarial game between a generator Gen and a discriminator $Disc$, where $Disc$ is trained on the true samples and the generated samples simultaneously, causing the generator coupled Gen to produce samples that are close to the true ones. The objective functions used by the GAN model are shown as follows:

$$Learn\_algo_G\left(Model_g\right)=E_{z\sim p(z_{noise})}\left[log\left(Disc\left(Gen\left(z\right)\right)\right)\right] \tag{12}$$

$$Learn\_algo_D\left(Model_d, Model_g\right) = E_{z\sim p(x_{real})} [log\left(Disc\left(x\right)\right)] + E_{z\sim p(z_{noise})}\left[log\left(1-Disc\left(Gen\left(z\right)\right)\right)\right] \quad (13)$$

8) Over-the-air spectrum poisoning attack: This type of poisoning attack focuses on the spectrum sensing time period and handling the transmitter's input data in the test and training stage. An adversary performs these attacks primarily to confuse the sender into making poor transmission decisions (i.e., an evasion attack) or to tamper with the sender's recycling process (i.e., a causal attack). To carry out an over-the-air spectrum poisoning attack, an attacker needs to have knowledge of the transmitter's sensing parameters and the communication protocol being used. Once the attacker has this information, he can craft adversarial examples that exploit the vulnerabilities of the transmitter's decision-making process. Sagduyu *et al.* [214] present over-the-air spectrum sensing poisoning attacks based on the application of adversarial machine learning. Benchmarking shows that over-the-air spectrum sensing poisoning attacks are superior to conventional jamming attacks and substantially decrease the transmitter throughput. Therefore, over-the-air spectrum poisoning attacks can be more effective than traditional jamming attacks, which simply flood the communication channel with noise to disrupt communication. Additionally, these attacks can significantly reduce the throughput of the transmitter, which can have significant consequences in applications such as wireless networks and IoT devices.

F. Sybil attacks

Sybil attacks involve the creation of multiple fake identities to manipulate the training process of a federated learning system. Different types of Sybil attacks are presented, including Sybil-based poisoning attacks and Sybil-based collusion attacks [147]. A Sybil attack involves the use of a single node inside the targeted network (or system) to simultaneously generate and operate multiple active false entities. The primary objective of these attacks is to gain the majority of influence within the targeted system in order to facilitate its manipulation.

1) Sybil-based poisoning attack: Sybils conduct federated learning poisoning attacks by delivering status updates that lead the distributed model to a poisoned common target. Based on contribution similarity, Fung *et al.* [147] propose a new defense system against Sybil-based poisoning attacks. The proposed system adapts the learning rate of clients based on contribution similarity.

2) Sybil-based collusion attack: Xiao *et al.* [166] propose the Sybil-based collusion attack, in which the attacker employs the label flipping scheme to train the poison data locally and collide with other poisoned patterns. The malicious adversary, meanwhile, will virtually implement several Sybil nodes in the network, so that the server chooses the collusion pattern to aggregate with a higher probability, constructing a poisoning pattern globally. Both CIFAR-10 and Fashion-MNIST are used as benchmark datasets to evaluate the attack performance in federated learning settings with convolutional Neural Networks (CNN). The experiment analysis shows that the proposed attack can achieve a more substantial attack effect.

G. Byzantine attacks

These attacks consist of Byzantine failures in federated learning, which can exploit the parameters of the local model on the affected devices during the learning phase. Specifically, Byzantine attacks in federated learning involve malicious participants that deliberately manipulate the local model to corrupt the training process. These attacks can severely compromise the integrity and accuracy of the final global model. The participants that perform Byzantine attacks are known as Byzantine faulty or malicious participants. These participants can exhibit arbitrary behavior and can send incorrect or intentionally corrupted updates to the central server, leading to the poisoning of the global model. The attacker's knowledge in Byzantine attacks can be classified into three levels: No knowledge, Partial knowledge, and Complete knowledge. In the case of no knowledge, the attacker has no specific knowledge of the distributed machine learning model, its architecture, or the data it's working with. An attacker with partial knowledge has some information about the distributed machine learning model or its architecture, but not complete details. In the scenario of complete knowledge, the attacker has access to all information related to the distributed machine learning system, including its architecture, model parameters, training data, and even updates from other nodes.

1) Federated learning-Byzantine attack: Different from the existing data poisoning attacks that affect the data collection integrity of the training datasets, Fang *et al.* [164] propose Byzantine attacks against federated learning which affect the integrity of the learning phase in the training process. The Byzantine attacks are formulated as optimization problems. The performance evaluation on four real-world datasets (i.e., Breast Cancer Wisconsin (Diagnostic), CHMNIST, Fashion-MNIST, and MNIST datasets) shows that the attacks against Byzantine-robust federated learning methods can substantially increase the error rates.

H. Inference attacks

In this type of attack, malicious users tend to exploit the already-trained models.

1) Model Inversion attack: Model inversion attacks are a type of inference attack, where an adversary attempts to infer sensitive information about the training data, such as inputs or parameters, by analyzing the output of the trained model. Fredrikson *et al.* [260] demonstrated the feasibility of such attacks on machine learning models trained on sensitive data, such as medical records, by using optimization techniques to reconstruct input data that results in a given model output.

2) Membership inference attack: A membership inference attack (MIA) is this kind of security vulnerability exposure where an adversary seeks to investigate whether an intended sample has been utilized for training the target ML model or not, on the basis of the model's behavior and output. Shokri *et al.* [261] conducted a black-box MIA attack against ML with a binary NN classification task-based formalization

Fig. 12: Defense methods against machine learning vulnerabilities

with a shadow learning technique to identify members of the training set from non-members. Tang *et al.* [262] classified these attacks into two categories, namely, 1) direct attacks, which often make a single query to interrogate the targeted sample in a direct manner, and 2) indirect attacks, which often makes numerous queries to interrogate vicinity of the targeted sample to deduce membership.

I. Highlights the impacts of vulnerabilities in Peer-to-Peer IoT learning

In this section, we have reviewed a variety of attacks that can significantly disrupt operations and compromise data in peer-to-peer IoT learning. Backdoor and poisoning attacks introduce malicious behavior into models or influence learning processes, while adversarial examples and evasion attacks can cause widespread model malfunctions or incorrect classifications. Combined attacks utilize multiple methods, making detection difficult. Sybil and byzantine attacks can disproportionately impact model learning or introduce unpredictability into the system. Dropping attacks disrupt the efficiency of the learning process. Inference, model extraction, and inversion attacks pose significant threats to data privacy by inferring sensitive information, copying proprietary models, or reconstructing original inputs from outputs. These vulnerabilities underscore the importance of strong security measures to protect data integrity, user privacy, and model robustness and accuracy.

VIII. Defense Methods Against Edge Learning Vulnerabilities

The previous section clearly highlighted the potential risks facing future 6G-IoT systems, as AI will be a key player in ensuring the proper functioning, optimizing, and safeguarding of these systems. However, in order to take full advantage of AI's benefits while avoiding its associated security risks, the cybersecurity research community has made significant efforts to make this possible. In this section, we will attempt to shed light on the importance of such efforts in a comprehensive manner and provide a classification of defensive mechanisms against ML attacks, as presented in Figure 12. In Table XV, we provide a list of state-of-the-art methods proposed for securing machine learning systems. Table XVI presents the summary of defense methods against federated machine learning vulnerabilities.

A. Training Phase Defense Methods

We start with the most vulnerable stage of ML, namely the training phase. We classify the deference mechanisms in the training phase against five threat categories, namely poisoning, backdoor, byzantine, Sybil, and combined attacks, as shown in Figure 13.

1) Poisoning Attacks Defense Mechanisms: Based on a selection of relevant recent literature, we classify the works that belong to this class into eight subclasses, namely

TABLE XV: Defense methods against machine learning vulnerabilities.

Defense framework	Year	Threat model	Mitigation solution	Learning mode	Targeted ML	Classifiers	Datasets	Pros (+) Open Issues (-)
Blanchard *et al.* [243]	2017	Byzantine attacks	Aggregation rules	Centralized learning	Spam filtering and image classification	A multilayer perceptron	MNIST dataset	+ The proposed approach works against attacks involving up to 33% of adversely affected parties - Federated poisoning attack is not considered
Fung *et al.* [147]	2018	Sybil attacks	Contribution similarity	Federated learning	Attack classification, Image classification, and Face recognition	SqueezeNet model	MNIST, VGGFace2, KDDCup, and Amazon datasets	+ Uses diversity of client updates to identify Sybil's poisoning. - Distributed backdoor attack is not considered
Qu *et al.* [244]	2020	Poisoning attacks	Bio-inspired	Federated Learning	Image classification	- CNN models	CIFAR-10 dataset	+ Fast convergence + Defense against poisoning attacks - Distributed backdoor attack is not considered
Wang *et al.* [245]	2021	Data poisoning attacks	Experience-based learning	Centralized learning	Image classification	- Deep Q-Network - LSTM model - DT model	Beijing PM 2.5 dataset	+ Robustness under data poisoning attacks + Training speed enhancement - Generative poisoning attack is not considered
Jangseung *et al.* [246]	2021	Poisoning attacks	Preprocessor-based	Centralized learning	Image classification	SVM model	MNIST and VCI Wisconsin breast cancer datasets	+ The protected model outperformed the unprotected model in all best-case scenarios - Federated poisoning attack is not considered
Li *et al.* [247]	2021	Label-flipping attacks	Dimensionality-reduction and clustering	Federated learning	Image classification	DNN model	CIFAR-10 and Fashion-MNIST datasets	+ Detect and mitigate data-poisoning attacks in Federated learning - The authentication using cryptography is not applied
Chan *et al.* [248]	2021	Label-flipping attacks	Knowledge Transfer	Centralized learning	Security-related applications	SVM model	Phishing Website Detection, Spam Assassin, and Letter Recognition datasets	+ Can significantly reduce the attack success rate - Distributed backdoor attack is not considered
Stokes *et al.* [249]	2021	Data poisoning attacks	Cryptography-based authentication	Centralized learning	Machine learning software	N/A	Text-based datasets	+ Authenticate both the trained model and the evaluation set - Federated poisoning attack is not considered
Shi *et al.* [250]	2021	Untargeted attack and targeted attack	Historical distance detection	Federated learning	Image Classification	CNN model	MNIST dataset	+ It can standardize the scenario of federated learning - Generative poisoning attack is not considered
Andreina *et al.* [251]	2021	Backdoor attacks	Feedback-based	Federated learning	Image classification	ResNet18 CNN model	CIFAR-10 and FEMNIST datasets	+ Achieve a false-positive rate below 5% with a detection accuracy of 100% - The authentication using cryptography is not applied
Gu *et al.* [252]	2021	Byzantine attacks	Architectural Style	Federated learning	Image classification	Conditional Variational Autoencoder	MNIST and FEMINIST datasets	+ Can withstand Byzantine attacks and targeted model poisoning attacks - Generative poisoning attack is not considered
Liu *et al.* [253]	2021	Combined attacks	Asynchronous convergence	Federated learning	Image classification	CNN model	MNIST and CIFAR-10 datasets	+ Achieve accuracy rates of 98.96% with horizontal mode and 95.84% vertical FL mode - The authentication using cryptography is not applied
Qiu *et al.* [180]	2021	Combined attacks	Moving target defense	Centralized learning	Sign recognition and Image classification	DNN model	GTSRB and Imagenet datasets	+ Resistance against Trojaning attacks - Federated poisoning attack is not considered
Shayan *et al.* [254]	2021	Sybil attacks	Verifiable random functions	Federated learning	Image classification	A logistic regression and softmax classifiers	Credit Card fraud and MNIST datasets	+ Ability to withstand poisoning attacks and node churn - The authentication using cryptography is not applied
Xie *et al.* [255]	2022	Local poisoning attacks	Bio-inspired	Federated learning	Image and text classification	- CNN models - LSTM models	IMDB, CIFAR-10, MNIST, and SST-5 datasets	+ Lower success attack rate + Prevents attacks by extracting the benign client's features - Distributed backdoor attack is not considered
Chen *et al.* [256]	2022	Poisoning attacks	Reputation-awareness	Federated learning	Image classification	CNN model	MNIST, Fashion-MNIST, and CIFAR-10 datasets	+ Achieve a 30% reduction in learning time and is strong in resistance to representative poisoning attacks - The authentication using cryptography is not applied
Hou *et al.* [257]	2022	Backdoor attacks	Federated Filters	Federated learning	Image classification	CNN model	MNIST and CIFAR10 datasets	+ Backdoor recognition with the accuracy up to 99% - The authentication using cryptography is not applied
Wang *et al.* [258]	2023	Backdoor attacks	Adaptive clustering	Federated learning	Image classification	CNN model	CIFAR-10, MNIST, and FEMNIST datasets	+ Defend against backdoor attacks - Increased computational overhead and longer training times
Jiang *et al.* [259]	2023	Label Flipping Attacks	Data quality detection mechanism	Federated learning	Image classification	CNN model	CIFAR-10 and Fashion-MNIST datasets	+ Defend against label flipping attacks with the lightweight generator - Introduce an additional layer of complexity and potential vulnerability

Fig. 13: Training phase defense methods against machine learning vulnerabilities

bio-inspired, experience-based learning, preprocessor-based, reputation-awareness, dimensionality-reduction and clustering, knowledge transfer, cryptography-based authentication, and historical distance detection, as presented below:

- ***Bio-inspired:*** these defensive mechanisms are motivated by biological functions or structures. For example, Xie *et al.* [255] propose an AI-targeted local poisoning attacks defense mechanism, which is inspired by the *biological immune system* and implemented in a multiple-party training environment. The proposed system supports antigen identification, immune reaction, and immunological recall with an adversarial pipeline, with no restrictions concerning malicious clients' numbers or the period of time they participate in the process. Also, it can make adaptive judgments regarding the aggregation weights of different local models. Detailed experimental outcomes across both image and text datasets, including MNIST, CIFAR-10, IMDB, and SST-5, with different neural networks, including CNN and LSTM, and under different types of poisoning attacks, such as local label flipping attacks. The results proved the merit of the proposed system with respect to model efficiency and resilience against poisoning attacks. Specifically, the accuracy reaches 95% even with 75% of the participating clients being compromised. Another Bio-inspired technique used for defending against poisoning attacks is the cognitive computing paradigm, which attempts to reproduce human cognitive processes within a computerized model. Qu *et al.* [244] propose a *decentralized cognitive computing* model (D2C) using a Blockchained FL framework for IIoT, specifically, for Industry 4.0 systems. The authors introduced blockchain FL into cognitive computing learning to enhance its privacy-preserving abilities in a fully decentralized manner. In addition, the authors introduced a modified Markov decision process (MDP) for performance optimization purposes. Evaluation of the CIFAR-10 dataset regarding three different aspects, global model accuracy, model convergence, and resistance to poisoning attacks, indicated the feasibility of the proposed system.
- ***Experience-based Learning:*** such defensive mechanisms incorporate practical, real-life experiences into the systems learning workflow. Wang *et al.* [245] propose a Deep Q-network (DQN)-based feature selection approach for cleaning multi-source data and mitigating poisoning attacks, which is influenced by RL in terms of an experience-based learning paradigm. Specifically, the problem is formulated as a contest in dynamic states among agents and the environment. To circumvent the problem of high computational complexity, the authors provided SS, a spatial search algorithm for quicker learning of the DQN agent. Performed simulations on Beijing PM 2.5, Colon Data, and MNIST datasets proved that the proposed approaches can successfully mitigate data

TABLE XVI: Summary of defense methods against edge learning vulnerabilities.

Defense methods	Defense mechanisms	Defense strategy	Methodology	Reference
Training Phase Defense Methods	Poisoning Attacks Defense Mechanisms	Bio-inspired	The idea is inspired by the *biological immune system* and deployed in the multi-party learning scenario	Xie *et al.* [255]
			The authors propose a decentralized cognitive computing model (D2C)	Qu *et al.* [244]
		Experience-based Learning	A Deep Q-network (DQN)-based feature selection approach is proposed to mitigate poisoning attacks	Wang *et al.* [245]
		Preprocessor-based	A preprocessor for minimizing the effects of poisoning attacks without affecting model performance	Jangseung *et al.* [246]
		Reputation-Awareness	Provides a dynamic reputation measurement system among clients participating in learning	Chen *et al.* [256]
		Dimensionality-reduction and Clustering	A Kernel Principal Component Analysis (KPCA) and K-mean-based defense strategy against FL-targeted data-poisoning attacks	Li *et al.* [247]
		Knowledge Transfer	A transfer learning-based defense against label-flipping attacks	Chan *et al.* [248]
		Cryptography-based Authentication	A cryptography-based authentication technique, called VAMP, to protect the integrity of training data	Stokes *et al.* [249]
		Historical Distance Detection	A defense strategy based on the analysis of the statistical relationship of the Euclidean distance between clients' models	Shi *et al.* [250]
	Backdoor Attacks Defense Mechanisms	Federated Filters	The federated backdoor filter defense is proposed in order to identify backdoor inputs	Hou *et al.* [257]
		Feedback-based	A feedback-based FL backdoor detection technique called BaFFLE is proposed, which is based on several clients' data for both training and model tampering discovery	Andreina *et al.* [251]
	Byzantine Attacks Defense Mechanisms	Architectural Style	An unsupervised Conditional Variational Autoencoder (CVAE)-based training anomaly detection system	Gu *et al.* [252]
		Aggregation Rules	Chooses and eliminates the most distant vectors with respect to the calculated average from the aggregated gradient updates	Blanchard *et al.* [243]
	Sybil Attacks Defense Mechanisms	Verifiable Random Functions	A secure blockchain-based framework for fully decentralized multi-party privacy-preserving ML	Shayan *et al.* [254]
		Contribution Similarity	A defensive discriminant technique to distinguish sybils from legitimate users	Fung *et al.* [147]
	Combined Attacks Defense Mechanisms	Moving Target Defense	an MTD-based defense strategy against Trojan attacks on DNN networks	Qiu *et al.* [180]
		Asynchronous Convergence	A security architecture dedicated to a secure FL-based learning workflow	Liu *et al.* [253]
Post-Training Phase Defense Methods	Trained Model Integrity Verification	Cryptographically-protected Provenance	A security scheme based on Blockchain and computing fuzzy hash values	Unal *et al.* [263]
		Optimization-based	A Bayesian Compromise Detection (BCD) algorithm to solve an optimization problem	Kuttichira *et al.* [264]
Inference Phase Defense Methods	Privacy Leakage Defense Mechanisms	Gradients Shielding	Secures exchanged aggregation with homomorphic encryption and differential privacy	Hao *et al.* [265]
		Self-Distillation	Generates equivalent member/non-member feedback to alleviate black-box membership inference threats	Tang *et al.* [262]

poisoning attacks.

- ***Preprocessor-based:*** these defensive mechanisms check the authenticity of the data before using it for training. Jangseung *et al.* [246] present Jangseung, a preprocessor for minimizing the effects of poisoning attacks without affecting model performance. The system is specifically developed to protect SVMs models from adversarial perturbations through the use of anomaly detection engines. The approach involves the identification of whether a given data point is an outlier based on historical training data, and newly entered data elements would be flagged and rejected as either faulty or malicious. The performance evaluation step using the MNIST and VCI Wisconsin breast cancer datasets involved training two replicate models under the same adversarial points, except one is protected by the proposed system. The results demonstrated that the protected model outperformed the unprotected model in all best-case scenarios, with 96.2% versus 53.2% with the MNIST dataset, as well as 88.18% versus 75.51% with the VCI breast cancer Wisconsin dataset.
- ***Reputation-Awareness:*** such defensive mechanisms check the genuineness level of the clients before allowing them to participate in the model training. Chen *et al.* [256] propose RAPFDL, a dynamic asynchronous anti-poisoning FL framework. The main concept is to provide a dynamic reputation measurement system among clients participating in learning. Individual clients can collect points by committing to update their local model parameters and exchange the earned points. As a result, customers are incentivized to download more updates to earn more reward points. The level of fairness is measured by the correlation coefficient between the accuracy of the final model and the individual contributions of the parties. Clients use a differentially private GAN (DPGAN) to generate artificial private data samples, and all of the exchange updates are encrypted using the improved additive homomorphic encryption. Then, the updates are stored in the blockchain as unmodifiable records, which ensures both audibility and transparency. It is demonstrated by the experimental results on MNIST, Fashion-MNIST, and CIFAR-10 datasets that RAPFDL can achieve a 30% reduction in learning time and is strong in resistance to representative poisoning attacks.
- ***Dimensionality-reduction and Clustering:*** these defensive mechanisms are designed to facilitate the data verification process. Li *et al.* [247] present a Kernel Principal Component Analysis (KPCA) and K-mean-based defense strategy against FL-targeted data-poisoning attacks, and

precisely, label-flipping attacks. The main reason for this combination is that KPCA, an algorithm for dimensionality reduction is able to handle non-linear data efficiently while coping well with linear data (e.g. skewed data). The KPCA algorithm is consequently effective in locating and filtering out malicious updates, as these updates contain unique features, while k-mean clustering is used to reduce noise.

- ***Knowledge Transfer:*** in these defensive mechanisms, the knowledge already acquired is used to sustain new training. Chan *et al.* [248] propose a transfer learning-based defense against label-flipping attacks. While different studies aim to exclude malicious data from the training process. The authors considered the problem of making full use of contaminated samples in training, in cases where the first approach cannot be achieved. The problem is expressed as transfer learning where the footprint of malicious samples is downplayed by retrieving only information similar to non-malicious samples in the affected datasets. In addition, a weight initialization technique is proposed which allocates a sample weight according to the appropriateness of its predicted cost provided by a classifier against poisoning attacks.
- ***Cryptography-based Authentication:*** these defensive mechanisms are designed to detect any possible manipulation of the previously verified data, which protects the data from tampering. Stokes *et al.* [249] present a cryptography-based authentication technique, called VAMP, to protect the integrity of training data. To provide the metadata for the media objects, the manifest is used. The main idea behind this concept involves protecting the data sets for training and validation, as well as safeguarding any existing software package utilized to train and evaluate the model. To accomplish the protection of the data sets, manifests for the training and validation datasets such as metadata and data bindings are created, to be either published on the VAMP service or integrated directly into the respective datasets. Then, both the training and evaluation packages are then uploaded to the service. The proposed system uses SHA2 (256 or 512) to create manifests, while their serialization can be performed using JSON for textual datasets or CBOR for binary implemented datasets.
- ***Historical Distance Detection:*** In such mechanisms, the objective is to inspect each client update before using it. Shi *et al.* [250] propose a defense mechanism based on the analysis of the statistical relationship of the Euclidean distance between clients' models. This defense mechanism is constructed based on the following statements. The distance between honest clients' models is not similar to the distance between harmful clients' models. The reason that most distances are similar is that the amount of malicious clients is much lower than the number of honest clients. Based on this, the main solution proposed is to force the aggregation server to choose the most possible benign clients, which are established on the minimal sum of their distance from each other. Experiment evaluations on MNIST with different data distribution techniques (IID and non-IID) validated the feasibility of the proposed system.

2) Backdoor Attacks Defense Mechanisms: We classify the works that belong to this class into two subclasses, namely federated filters and feedback-based mechanisms, as given below:

- ***Federated Filters:*** In these mechanisms, the implementation takes place on the aggregation server side and it can be used for client monitoring and security attack filter distribution. Hou *et al.* [257] focused on proposing a defensive mechanism against backdoor attacks in IIoT-FL environments. The authors trained multiple backdoor filters along with various combinations of XAI models and classifiers on the server side in order to guarantee the identification of the backdoor entries. In addition, the authors proposed a blur-label-flipping strategy to sanitize the backdoor locations, which allows reclaiming of data availability. The results of experimental evaluations on MNIST and CIFAR10 datasets prove the validity of the system with an accuracy of up to 99% in recognizing backdoor samples.
- ***Feedback-based:*** such mechanisms are designed to continuously monitor and provide authenticity feedback on participating clients. Andreina *et al.* [251] propose a feedback-based FL backdoor detection technique called BaFFLE, which makes use of several clients' data for both training and model tampering discovery. The defense strategy consists of taking advantage of the existence of diverse datasets across clients by embedding a feedback loop within the FL workflow. To address the various challenges of running a distributed protocol in a Byzantine environment, the authors used an off-the-shelf procedure to locally (clients-side) benchmark and compare the updated model's classification performance against the former one, while excluding updates that demonstrate unexpected behavior. The proposed system is evaluated using the CIFAR-10 and FEMNIST datasets, which results show 100% accuracy with less than 5% false positive rate.

3) Byzantine Attacks Defense Mechanisms: We organize the works that belong to this class into two subclasses, namely architectural style, and aggregation rules, as provided below:

- ***Architectural Style:*** such mechanisms exploit the architectural learning style to detect and eliminate malicious updates. Gu *et al.* [252] proposed an unsupervised Conditional Variational Autoencoder (CVAE)-based training anomaly detection system, called Fedcvae, which aims to accurately discover and eradicate malicious model updates to negate their harmful consequences. The architecture of Fedcvae consists of an *encoder-decoder* design. Using the encoder, the original variables are mapped into low-dimensional embeddings, while the decoder rebuilds the initial variables from them. The performance evaluation under four different datasets, namely Vehicle, Synthetic, MNIST, and FEMINIST (under both IID and non-IID data distribution approaches) shows that the proposed system can withstand Byzantine attacks and

targeted model poisoning attacks with up to 30% of the participating clients being malicious.

- ***Aggregation Rules:*** The main purpose of these mechanisms is to ensure safe learning, whether all clients are honest or there is a subset of malicious clients. Blanchard *et al.* [243] investigated Byzantine failure robustness for distributed Stochastic Gradient Descent (SGD)-based frameworks. The research seeks to address the problem of Byzantine failure resilience of an SGD system with no limitation on the dimension or parameter space size. The proposed solution entails the formulation of a resilience property of the aggregation rule that grasps the core demands for ensuring convergence regardless of having a subset of Byzantine parties. This technique chooses and eliminates the most distant vectors with respect to the calculated average from the aggregated gradient updates. The distance is computed using the Euclidean spacing among the gradient vectors. The approach works well against attacks involving up to 33% of adversely affected parties.

4) Sybil Attacks Defense Mechanisms: We classify the works that belong to this class into two subclasses, including verifiable random functions and contribution similarity, as presented below:

- ***Verifiable Random Functions:*** in short, a verifiable random function (VRF) is given two inputs (a secret key and a seed) and returns two associated values (hash and proof). The proof allows everyone who is carrying the public key of the given peer to confirm that the given hash was indeed produced by a peer who is in possession of the private key. Shayan *et al.* [254] present Biscotti, a secure blockchain-based framework for fully decentralized multi-party privacy-preserving ML. The proposed system implements a robust hash protocol built upon the most recent block hash and VRFs for subgroup selection among peers who are in the process of carrying out individual steps (noise addition, updates validations, and secure aggregations). The proposed system is based on the Proof of Federation (PoF) protocol, a proposed blockchain consensus protocol that employs FL defenses and then renders them enforceable across decentralized P2P environments. As the selection of subgroups is performed by the VRF, the stake of the peer is treated as the individual's reputation acquired by contributing favorably to the shared model. This guarantees that an opponent is unable to expand its leverage in the system by establishing too many peers and still not enhancing the model, which is useful for mitigating the effect of Sybils. The experimental results with 26 committee sizes provide immunity against an opponent who controls 30% of the system's stakes.
- ***Contribution Similarity:*** Fung *et al.* [147] present a defensive discriminant technique to distinguish Sybils from legitimate users based on the diversity of their relative gradient updates. The proposed learning parameters make use of a per-client adaptive learning rate that relies on the contribution similarity among the clients. The identification of malicious users is based on a common goal, and therefore their model updates are susceptible to low variance, which is done by searching for similarities regarding indicative characteristics. Experimental evaluations on four datasets, namely MNIST, VGGFace2, KDDCup, and Amazon, with 5 attack scenarios indicated that the proposed system, in conjunction with other modules (such as Multi-Krum), can successfully mitigate a range of different attack types and even when Sybils submerge legitimate users.

5) Defense Mechanisms Against Combined Attacks: We organize the works that belong to this class into two subclasses, including Moving Target Defense and Asynchronous Convergence.

- ***Moving Target Defense:*** the purpose of MTD is to actively shift the surface available for attacks. Qiu *et al.* [180] propose MT-MTD, an MTD-based defense strategy against Trojan attacks on DNN networks. The framework takes an attack-defense game approach, specifically, a signaling game. MT-MTD has four main steps. The first is a dimensional division of the training set by the defender and then passed to the attacker for training. The second is a random selection from the resulting collection of dimensional combinations. The third step is the weight adjustment. The last is the consensus process.
- ***Asynchronous Convergence:*** used in situations when the aggregation can be done asynchronously. Liu *et al.* [253] propose a security architecture dedicated to a secure FL-based learning workflow. The architecture is composed of two parts. The first one, named FedBlock, is proposed to introduce decentralization in the learning process through the use of the blockchain, while the second one, named FedAC, permits the FL to carry out a global aggregation in an asynchronous manner while considering a staleness coefficient. The system is dedicated to securing the learning process against a multitude of threats, including Single Point Of Failure (SPOF), unconventional learning failures, and dedicated attacks. Experimental evaluation on a physical deployment, namely a Raspberry Pi (4b) under the MNIST dataset, shows that the system can reach accuracy scores as high as 98.96% in horizontal FL and 95.84% in vertical FL. The reported results of experimental evaluations on the CIFAR-10 and Fashion-MNIST datasets show that the proposed approach can be effective in identifying and differentiating honest updates from malicious ones.

B. Post-Training Phase Defense Methods

The defense mechanisms in this class tend to safeguard the model from tempering after the training phase.

1) Trained Model Integrity Verification: We organize the works that belong to this class into two subclasses, namely cryptographically-protected Provenance and optimization-based techniques, as provided below:

- ***Cryptographically-protected Provenance:*** these defensive mechanisms protect the model trained using encryption. Unal *et al.* [263] propose a security scheme

based on Blockchain and computing *fuzzy hash* values for protecting FL algorithms operating in IoT systems. Notably, the proposed scheme stores the model parameters in the blockchain, although this is true to some extent, the model parameters on the blockchain are not stored unchanged. In other words, the authors applied a one-way hash function upon these model parameters prior to their storage on the blockchain. Thus, the introduced scheme provides an efficient approach that does not affect FL's privacy concerns. In another approach under the same context, Stokes *et al.* [249] propose a cryptography-based authentication, which provides a cryptography-based provenance module to protect trained models against alteration of their settings or underlying structure. The proposed authentication can protect against model poisoning attacks.

- ***Optimization-based:*** in a work by Kuttichira *et al.* [264], the authors propose a Bayesian Compromise Detection (BCD) algorithm that aims to address the potential security risk associated with malignant modification of stored models in the cloud. The main solution consists of solving an optimization problem that seeks to maximize the discrepancies in prediction outcomes by comparing the genuine model with the damaged model. Although this task seems easy at first glance, the difficulty of the problem lies in three different points according to the authors. First, its non-convex nature, second, the large space dimensionality when searching through the training input distribution for the sensitive sample, and third, a black-box view of the compromised model is all that cloud clients can have. The proposed solution consists of two parts. First, a Variational Autoencoder (VAE) is used to associate high-dimensional data with a low-dimensional nonlinear space, and second, Bayesian optimization (BO) is applied to determine the generally ideal sensitive sample, which can identify the model corruption with little overhead. Experiments on MNIST, Olivetti, and CIFAR-100 datasets illustrate the capability of the proposed approach, with results of up to 100% detection rate.

C. Inference Phase Defense Methods

The defense mechanisms in this class are designed to prevent attacks against ML models during the inference phase.

1) Privacy Leakage Defense Mechanisms: We classify the work belonging to this class into three subclasses, namely privacy leakage, self-distillation, and overfitting control, as provided below:

- ***Gradients Shielding:*** the main objective of these mechanisms is to protect the gradient of the model in a cryptographic way. Hao *et al.* [265] implemented a privacy-preserving IDS for FL-based industrial environments, named PEFL. The proposed system secures exchanged aggregation with homomorphic encryption and differential privacy. In essence, the homomorphically encrypted data of the private gradients is incorporated into the A-LWE (augmented learning with error). Benchmarking on the MNIST dataset exhibits good performance, along with computational and communication cost efficiency.
- ***Self-Distillation:*** in a work by Tang *et al.* [262], the authors propose SELENA, a privacy-preserving model learning framework that generates equivalent member/non-member feedback to alleviate black-box membership inference threats. The proposed system consists of two major components. The first is the Split Adaptive Inference Ensemble (Split-AI), which allows the model to behave similarly to members and non-members samples. This is done by training sub-models through random subsets within the training set. The second is a self-distillation mechanism that transfers knowledge about the model created by Split-AI to deliver a protected end model. To do this, Split-AI is first interrogated with the exact training data to retrieve the related prediction sequences. Then, with these predictions as soft labels, the protected end model is trained. Evaluations on three different datasets (Purchase100, Texas100, and CIFAR100), with ResNet-18 and a fully connected four-layer NN, and different MIA attacks, including direct single-query attacks, label-only attacks, and adaptive attacks, showed that the proposed system achieved a good tradeoff between practicality and privacy.
- ***Overfitting Control:*** when given a data point on which models have been trained, they return a high aftereffect value on a class relative to the others, reflecting the underlying overfitting nature of ML models, which is taken to be one of the reasons for the effectiveness of MIAs. Hence, according to Salem *et al.* [266] controlling overfitting is one way to mitigate such attacks. The authors propose the use of two approaches: one classical and the other relatively new, namely *dropout* and *model stacking*. The first one is a regularization technique that prevents complex co-matching on training data and is specifically used for DL models. The second one is proposed to be used for other ML classifiers. The main concept is to hierarchically organize several ML models in order to avoid overfitting.

IX. Lessons learned, Open issues and Challenges

Despite the significant efforts made by the scientific community to strengthen security in the cyber world in general, and for future networks such as 6G, in collaboration with emerging technologies such as IoT and AI in particular, there is however a long way to be traveled, if we are to achieve a fully secure cyber environment. In this section, we present the lessons learned, open issues, and challenges.

A. Lessons learned

Through comprehensive reviews and in-depth analysis, we were able to classify the datasets used by the scientific community for experimenting and evaluating ML techniques on cyber attacks into seven main categories, including, IoT-based multi-purpose security, attack classification, image classification, time series classification, human activity recognition,

sentiment classification, location awareness, and text classification. From the attacks against machine learning systems, we found twenty-one attacks. According to the actual context of the attack in edge learning, we were able to classify them into eight categories, including, backdoor attacks, adversarial examples, combined attacks, poisoning attacks, Sybil attacks, byzantine attacks, inference attacks, and drop attacks. Based on the deployment strategy of each security defense, we were able to classify the defense methods against edge learning vulnerabilities into three categories, including, training phase defense methods, post-training phase defense methods, and inference phase defense methods.

From the above analysis and reviews that we completed, we suggest the following steps for proposing defense methods against edge learning vulnerabilities in 6G-enabled IoT networks:

- Definition of the infrastructure of the 6G-enabled IoT network as well as the emerging technologies adopted for each layer (e.g., Edge layer, Fog Layer, SDN Layer, Blockchain Layer, Digital twins layer...etc).
- Definition of the edge learning model (i.e., centralized learning, federated learning, distributed learning).
- Identification of the attacks against machine learning systems (e.g., backdoor attacks, adversarial examples, poisoning attacks, inference attacks, ... etc.)
- Systematize the threat models based on three dimensions: adversarial goal, attack strategies, and malicious client selection.
- Selection of the defense method against edge learning vulnerabilities (i.e., training phase defense methods, post-training phase defense methods, or inference phase defense methods).
- Selection of the datasets adopted for the 6G-enabled network (e.g., attack classification dataset, image classification dataset, human activity recognition dataset, sentiment classification dataset, and text classification dataset).
- Selection of the data distribution (i.e., IID and Non-IID).
- Experimenting and evaluating the defense method in terms of accuracy, precision, recall, and attack rate.
- Study the performances of the system with the application of the proposed defense method in terms of scalability and interoperability.

The use of network slicing, multi-access edge computing, cloud computing, virtualization, terahertz and sub-terahertz communications, and artificial intelligence in the 5G/6G IoT testbeds demonstrates the complexity and diversity of the technologies being developed. These testbeds offer an opportunity to explore new possibilities in smart transportation, energy management, healthcare, and industrial automation.

In summary, the development of effective defense methods against edge learning vulnerabilities in 6G-enabled IoT networks requires a systematic approach that considers the network infrastructure, edge learning models, attack types, threat models, defense methods, datasets, data distribution, and system performance. With a comprehensive understanding of these factors, researchers can better develop and evaluate defense methods to improve the security and resilience of 6G-enabled IoT networks.

B. Open issues and Challenges

Table XVII provides an overview of the open issues and challenges in 6G-IoT machine learning vulnerabilities. The challenges covered include reliable and trustworthy learning for 6G-IoT intelligence, security solutions adaptability, ethics by design, datasets formation/availability, learning complexity, high-quality data availability, ML vulnerabilities elimination, and defense strategies implementations.

1) Reliable and Trustworthy Learning for 6G-IoT Intelligence: AI capabilities have proven to be a key component of future technologies, however, there is more involved in ensuring security. Here we present some of the challenges to be addressed when creating reliable and trustworthy learning environments.

- Natively Secured AI: Adversary-aware AI, or natively secured AI, is an upcoming class for AI-based systems that focuses on building security into the data preparation, learning, storage, and inference stages. This paradigm is necessary because cyberattacks can compromise the integrity, confidentiality, and availability of AI models and their associated data, leading to adverse consequences. Similar to how cyber attacks forced developers to write secure code and libraries for programming languages, the increasing adoption of AI-based systems will necessitate the adoption of secure AI practices. This will involve not only the development of secure AI algorithms but also the design and implementation of secure hardware and software architectures to support these algorithms. The challenges associated with implementing natively secured AI are considerable, given the complexity and uncertainty associated with AI techniques. Additionally, the limitations of IoT devices, such as their limited processing and storage resources, may pose a challenge to running complex AI models. To address this challenge, lightweight and efficient ML algorithms tailored for these devices must be developed, similar to the situation encountered in the case of lightweight IoT-based authentication protocols. In summary, natively secured AI is an upcoming class for AI-based systems that focuses on building security into every aspect of AI, from data preparation to inference. While there are significant challenges associated with implementing this paradigm, it is necessary to ensure the security of AI-based systems in the face of increasingly sophisticated cyber attacks.
- Trustworthy AI: Trustworthy AI is a key component of natively secure AI, assuring that AI-based systems operate reliably, transparently, and ethically. Specifically, Trustworthy AI refers to AI that is designed to be transparent, explainable, and accountable. Trustworthy AI systems focus on minimizing the risks associated with AI algorithms and their decision-making processes by ensuring that they operate within specific ethical constraints. To achieve trustworthy AI, secure AI algorithms must be developed that incorporate ethical and legal standards, including human rights, data protection, and privacy.

TABLE XVII: Open Issues and Challenges in 6G-IoT edge learning vulnerabilities.

Challenge	Description	Key Considerations
Reliable and Trustworthy Learning	Ensuring the security of AI-based systems by building security into every aspect of AI, from data preparation to inference	- Adoption of secure AI practices - Development of secure hardware and software architectures - Development of lightweight and efficient ML algorithms
Security Solutions Adaptability	Developing security solutions that are highly adaptive to the dynamic and heterogeneous nature of wireless networks	- Use of pre-trained models, unified datasets, and agreed-upon models between gNB and UEs.
Ethics by Design	Addressing the ethical dimensions of AI systems at the earliest stage of their design through the "Ethics by Design" approach	- Adoption of ethical guidelines - Consideration of the specific requirements and constraints of different applications and industries.
Datasets Formation/Availability	Creating dependable edge learning datasets as well as managing complexity, and ensuring high-quality data availability	- Use of Explainable AI (XAI) techniques - Introduction of a confidence level and validations, and layered security models
Learning Complexity	Managing complexity in mobile communication systems and other complex environments through Explainable AI (XAI) techniques	- Use of XAI to provide transparency, understanding, and trust in AI models
High-Quality Data Availability	Ensuring the availability of high-quality data and using techniques that introduce a confidence level and validations for FL	- Adoption of techniques that introduce a confidence level and validations
ML Vulnerabilities Elimination	Balancing security, performance, profitability, simplicity and complexity, and backward compatibility with future security	- Striking a balance between security, performance, and profitability - Implementing a layered security model
Defense Strategies Implementations	Ongoing research and development of security solutions to stay ahead of emerging threats, such as zero-day attacks	- Use of ML-based security for detecting zero-day attacks
Large language models at the Edge	Deploying large language models at the edge for 6G-enabled IoT systems introduces several challenges	- Edge devices are often battery-powered and have strict energy consumption limits - Latency affect real-time responses and system efficiency.
Real-time Decision Making	Enabling 6G-IoT devices to make real-time decisions despite some ML models being computationally expensive	- Development of lightweight ML algorithms - Optimizing computational resources for real-time processing
Model Generalization	Ensuring ML models generalize well across various 6G-IoT devices despite differences in data distributions	- Development of diverse training datasets - Adoption of model validation techniques to assess generalization
Scalability	Dealing with the increasing number of IoT devices and the data they generate	- Deployment of distributed ML algorithms - Development of efficient data storage solutions

This is essential to ensure that AI-based systems are transparent and explainable and that users can understand how the system works, how it makes decisions, and how it uses data.

- Security Solutions Adaptability: The dynamic nature of settings shifts across wireless networks necessitates that the proposed security solutions must be highly adaptive. Due to the fact that many operators collaborate with each other, this diversity can pose a significant heterogeneity problem if not handled effectively. Solutions to this problem have been briefly discussed, including pre-trained models, unified datasets, and agreed-upon models between Next Generation NodeB (gNB) and user equipment (UEs) [267]. However, real-world implementations and further tests are different stories with varying perspectives, posing a future challenge for these networks. Hence, the adaptability of security solutions is crucial for the security of wireless networks, and ongoing research and testing are necessary to address the challenges posed by the dynamic and heterogeneous nature of these networks.
- Ethics by Design: While machine intelligence is a promising addition to future networks, making them fully automated, they do not accommodate ethical guidelines in the same way as humans [5]. The "Ethics by Design" approach addresses the ethical dimensions of AI systems at the earliest stage of their design [268]. And as much as it is not an easy subject for a machine to learn, the concept itself is difficult to define, as even among humans and nations, the concept sometimes has wide variations in how it is described. By adopting the Ethics by Design approach, AI designers and developers can help ensure that AI systems are developed in a responsible and ethical manner.

2) Datasets Formation/Availability for Edge Learning: In addition to introducing security into AI, other challenges may arise along the way, including the availability and the creation of dependable edge learning datasets. We highlight two important issues, namely managing complexity and ensuring reliable data resource availability.

- Learning Complexity: In a complex environment such as mobile communication systems, which is likely to be even more complex with the projected introduction of a range of technologies for future networks such as 6G, and given the limited hardware capabilities of IoT devices and cellular entities, particularly in terms of processing power, battery life, and storage, this is going to be a serious problem in terms of how well these entities can handle complexity in the interests of functionality. Explainable AI (XAI) can solve part of the problem, unlike the black-box concept, as it effectively introduces understanding, management, and a level of trust in AI models, making it easier to reduce complexity without affecting accuracy. XAI is an important tool for managing complexity in mobile communication systems and other complex environments. By providing transparency, understanding, and trust in AI models, XAI can help stakeholders navigate complex environments while maintaining functionality and accuracy.
- High-Quality Data Availability: The FL paradigm solves

the problem of compromising the privacy of private data in the learning phase. However, as we have seen in the previous sections, this paradigm is subject to different types of threats. One of them is related to data, which can be a significant danger for future wireless networks since distributed learning is supposed to be a key component of future networks. It can be tricky to trust models trained on data that the operator has not seen or validated, such as when relying on UEs for the training task. Similarly, even when dealing with internal components such as gNBs and Intelligent Radios (IRs), it can be dangerous to trust data collected from the surrounding environment and used for training directly without the proper validations. Techniques that introduce a confidence level and validations must be applied before incorporating the models directly into the networks. Overall, the availability of high-quality data and the use of techniques that introduce a confidence level and validations are critical to ensuring the success of federated learning in wireless networks. By adopting these techniques, operators can trust the models generated by the learning process and use them to make informed decisions about the network's configuration and performance.

3) ML Vulnerabilities Elimination and Defense Strategies Implementations: It is difficult to develop a security solution that guarantees equal levels for conflicting aspects. Here we discuss the trade-offs that can arise when trying to eliminate ML vulnerabilities and develop security solutions.

- Security vs. Performance and/or Profit: The very high data rates promised by 6G will enable a variety of time-sensitive IoE applications, requiring real-time security techniques to be deployed in turn, as conventional security schemes are subject to short-term and long-term failure under such circumstances. This results in different trade-offs, on the one hand, between security and performance, e.g., between privacy and accuracy in FL, since introducing a lot of noise to protect the data will result in reduced accuracy, and vice versa. On the other hand, the effort for real-time security techniques is costly and depends on what the telecom industry is willing to pay and what are their priorities. The effort for developing and implementing real-time security techniques is costly, and the telecom industry may have to prioritize security, performance, and profitability. In some cases, the industry may decide to prioritize performance or profitability over security, which could result in compromises in terms of the security of the network and the data transmitted over it. Therefore, it is essential to strike a balance between security, performance, and profitability while deploying real-time security techniques for 6G networks. This will require careful consideration of the specific requirements and constraints of different applications and industries and collaboration between stakeholders in the telecom industry, academia, and government.
- Simplicity vs. Complexity: Since future networks are expected to be more complex than previous generations, which is natural, and this has persisted in the evolution of all preceding generations. A trivial conclusion that could be drawn directly is that security solutions need to be complex as well, however, this can be seen as a double-edged sword, since, in order to keep the system stable, more secure, and have fewer vulnerabilities, simplicity, and transparency are keys [44]. In the case of 6G-enabled IoT networks, which are expected to be more complex, it is important to consider both simplicity and complexity in security solutions. One approach could be to implement a layered security model, where multiple layers of security are used to protect against different types of attacks. Each layer can be designed with a simple and transparent solution that is easy to understand and audit, while also being integrated into a larger, more complex security architecture. ML-based security is a promising approach for detecting zero-day attacks, which are attacks that exploit vulnerabilities that are unknown to security experts. However, as mentioned, zero-day attacks that specifically target the ML algorithms themselves can be difficult to detect. This highlights the need for ongoing research and development in security solutions to stay ahead of emerging threats.
- Backwards Compatibility vs. Future Security: In addition to the increased complexity, maintaining backward compatibility is another feature that persists in all previous generations. While it has its benefits, including reduced deployment costs compared to a new standalone deployment and ease of upgrading, the backward compatibility functionality potentially exposes old vulnerabilities. In addition, AI-based security must also take into account some of the previous generation's parameters in order to function properly. This may be something to be carefully examined in the future. Overall, balancing the need for backward compatibility with the need for future security is a complex challenge that requires careful consideration. It may be necessary to prioritize security over backward compatibility in certain cases, but in other cases, it may be possible to reach a trade-off between the two.

4) Large language models at the Edge: Large language models, like OpenAI's GPT-4, FalconLLM, and Bert, are increasingly being used in a wide range of applications, from content generation to customer service and much more [269]–[271]. However, deploying large language models at the edge for 6G-enabled IoT systems introduces several challenges. First, security and privacy issues are magnified in a 6G environment due to the sheer volume and variety of devices [272]. The immense data flow, which includes sensitive data, presents a significant risk of cyber threats and vulnerabilities. Second, despite 6G's promise of higher data rates and lower latencies, managing the scale and complexity of these models can be a challenge due to the inherent resource constraints of IoT devices, potentially affecting model performance and efficiency. The high density of IoT devices in 6G networks could also strain network resources and introduce latency issues, impacting real-time responses and decision-making [95]. Third, maintaining consistency of the model's deployment across a diverse range of IoT devices is difficult, and software

updates become challenging due to the vast distribution of devices. Therefore, the deployment of large language models at the edge of 6G IoT environments necessitates careful attention to these challenges to ensure security, efficiency, and compliance.

X. Conclusions

While 5G is being deployed and commercialized around the world, researchers are getting excited about what 6G can and should be. One of the most endorsed views is that AI should be a core component of 6G rather than just an add-on utility. edge learning involves making the edge of the network intelligent, where models are trained at the edge using coordinated distributed learning paradigms such as FL, with data available across a variety of edge devices. However, given the existing vulnerabilities in ML, its adoption in the absence of adequate security considerations can expose the network to various threats. In this paper, we have surveyed the state-of-the-art of existing vulnerabilities and defenses of federated machine learning for 6G-enabled IoTs. We have summarized the existing surveys on machine learning for 6G–IoT security as well as machine learning-associated threats in three different learning modes, namely, centralized, federated, and distributed. Through extensive research and analysis that has been conducted, we have classified the threat models against machine learning into eight categories, including, backdoor attacks, adversarial examples, combined attacks, poisoning attacks, Sybil attacks, byzantine attacks, inference attacks, and dropping attacks. In addition, we have analyzed the state-of-the-art defense methods against federated machine learning vulnerabilities. Finally, as new attacks and defense technologies are realized, new research and future overall prospects for 6G-enabled IoTs are discussed. There still exist several challenging research areas on new attacks and defense technologies, which should be further investigated in the near future.

References

[1] F. Tariq, M. R. Khandaker, K.-K. Wong, M. A. Imran, M. Bennis, and M. Debbah, "A speculative study on 6g," IEEE Wireless Communications, vol. 27, no. 4, pp. 118–125, 2020.

[2] S. Dang, O. Amin, B. Shihada, and M.-S. Alouini, "What should 6g be?" Nature Electronics, vol. 3, no. 1, pp. 20–29, 2020.

[3] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, "Toward an intelligent edge: Wireless communication meets machine learning," IEEE communications magazine, vol. 58, no. 1, pp. 19–25, 2020.

[4] W. Saad, M. Bennis, and M. Chen, "A vision of 6g wireless systems: Applications, trends, technologies, and open research problems," IEEE network, vol. 34, no. 3, pp. 134–142, 2019.

[5] Y. Siriwardhana, P. Porambage, M. Liyanage, and M. Ylianttila, "Ai and 6g security: Opportunities and challenges," in 2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit). IEEE, 2021, pp. 616–621.

[6] A. Chouman, D. M. Manias, and A. Shami, "Towards supporting intelligence in 5g/6g core networks: Nwdaf implementation and initial analysis," in 2022 International Wireless Communications and Mobile Computing (IWCMC). IEEE, 2022, pp. 324–329.

[7] M. A. Ferrag, O. Friha, L. Maglaras, H. Janicke, and L. Shu, "Federated deep learning for cyber security in the internet of things: Concepts, applications, and experimental analysis," IEEE Access, vol. 9, pp. 138 509–138 542, 2021.

[8] A. Oseni, N. Moustafa, H. Janicke, P. Liu, Z. Tari, and A. Vasilakos, "Security and privacy for artificial intelligence: Opportunities and challenges," arXiv preprint arXiv:2102.04661, 2021.

[9] M. Kuzlu, C. Fair, and O. Guler, "Role of artificial intelligence in the internet of things (iot) cybersecurity," Discover Internet of things, vol. 1, no. 1, pp. 1–14, 2021.

[10] V. Shejwalkar, A. Houmansadr, P. Kairouz, and D. Ramage, "Back to the drawing board: A critical evaluation of poisoning attacks on production federated learning," in 2022 IEEE Symposium on Security and Privacy (SP). IEEE, 2022, pp. 1354–1371.

[11] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," ACM Transactions on Intelligent Systems and Technology (TIST), vol. 10, no. 2, pp. 1–19, 2019.

[12] F. Hussain, R. Hussain, S. A. Hassan, and E. Hossain, "Machine learning in iot security: Current solutions and future challenges," IEEE Communications Surveys & Tutorials, vol. 22, no. 3, pp. 1686–1721, 2020.

[13] B. K. Mohanta, D. Jena, U. Satapathy, and S. Patnaik, "Survey on iot security: challenges and solution using machine learning, artificial intelligence and blockchain technology," Internet of Things, vol. 11, p. 100227, 2020.

[14] L. Lyu, H. Yu, and Q. Yang, "Threats to federated learning: A survey," arXiv preprint arXiv:2003.02133, 2020.

[15] O. A. Wahab, A. Mourad, H. Otrok, and T. Taleb, "Federated machine learning: Survey, multi-level classification, desirable criteria and future directions in communication and networking systems," IEEE Communications Surveys & Tutorials, vol. 23, no. 2, pp. 1342–1397, 2021.

[16] D. C. Nguyen, M. Ding, Q.-V. Pham, P. N. Pathirana, L. B. Le, A. Seneviratne, J. Li, D. Niyato, and H. V. Poor, "Federated learning meets blockchain in edge computing: Opportunities and challenges," IEEE Internet of Things Journal, vol. 8, no. 16, pp. 12 806–12 825, 2021.

[17] M. Alazab, S. P. RM, M. Parimala, P. K. R. Maddikunta, T. R. Gadekallu, and Q.-V. Pham, "Federated learning for cybersecurity: concepts, challenges, and future directions," IEEE Transactions on Industrial Informatics, vol. 18, no. 5, pp. 3501–3509, 2021.

[18] S. Zaman, K. Alhazmi, M. Aseeri, M. R. Ahmed, R. T. Khan, M. S. Kaiser, and M. Mahmud, "Security threats and artificial intelligence based countermeasures for internet of things networks: a comprehensive survey," Ieee Access, 2021.

[19] V. Mothukuri, R. M. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, and G. Srivastava, "A survey on security and privacy of federated learning," Future Generation Computer Systems, vol. 115, pp. 619–640, 2021.

[20] B. Ghimire and D. B. Rawat, "Recent advances on federated learning for cybersecurity and cybersecurity for federated learning for internet of things," IEEE Internet of Things Journal, 2022.

[21] X. Ma, J. Zhu, Z. Lin, S. Chen, and Y. Qin, "A state-of-the-art survey on solving non-iid data in federated learning," Future Generation Computer Systems, vol. 135, pp. 244–258, 2022.

[22] Z. Liu, J. Guo, W. Yang, J. Fan, K.-Y. Lam, and J. Zhao, "Privacy-preserving aggregation in federated learning: A survey," IEEE Transactions on Big Data, 2022.

[23] P. Boobalan, S. P. Ramu, Q.-V. Pham, K. Dev, S. Pandya, P. K. R. Maddikunta, T. R. Gadekallu, and T. Huynh-The, "Fusion of federated learning and industrial internet of things: A survey," Computer Networks, vol. 212, p. 109048, 2022.

[24] Z. Yang, M. Chen, K.-K. Wong, H. V. Poor, and S. Cui, "Federated learning for 6g: Applications, challenges, and opportunities," Engineering, vol. 8, pp. 33–41, 2022.

[25] I. H. Sarker, A. I. Khan, Y. B. Abushark, and F. Alsolami, "Internet of things (iot) security intelligence: a comprehensive overview, machine learning solutions and research directions," Mobile Networks and Applications, pp. 1–17, 2022.

[26] L. Qian, P. Yang, M. Xiao, O. A. Dobre, M. Di Renzo, J. Li, Z. Han, Q. Yi, and J. Zhao, "Distributed learning for wireless communications: Methods, applications and challenges," IEEE Journal of Selected Topics in Signal Processing, vol. 16, no. 3, pp. 326–342, 2022.

[27] C. Ma, J. L. Wei, B. Liu, M. Ding, L. Yuan, Z. Han, H. V. Poor et al., "Trusted ai in multi-agent systems: An overview of privacy and security for distributed learning," arXiv preprint arXiv:2202.09027, 2022.

[28] T. Zhang, L. Gao, C. He, M. Zhang, B. Krishnamachari, and A. S. Avestimehr, "Federated learning for the internet of things: applications, challenges, and opportunities," IEEE Internet of Things Magazine, vol. 5, no. 1, pp. 24–29, 2022.

[29] B. Veith, D. Krummacker, and H. D. Schotten, "The road to trustworthy 6g: A survey on trust anchor technologies," IEEE Open Journal of the Communications Society, 2023.

[30] B. Mao, J. Liu, Y. Wu, and N. Kato, "Security and privacy on 6g network edge: A survey," IEEE Communications Surveys & Tutorials, 2023.

[31] A. Alotaibi and A. Barnawi, "Securing massive iot in 6g: Recent solutions, architectures, future directions," Internet of Things, p. 100715, 2023.

[32] H. Hua, Y. Li, T. Wang, N. Dong, W. Li, and J. Cao, "Edge computing with artificial intelligence: A machine learning perspective," ACM Computing Surveys, vol. 55, no. 9, pp. 1–35, 2023.

[33] M. M. Y. Al-Quraan, L. Mohjazi, L. Bariah, A. Centeno, A. Zoha, K. Arshad, K. Assaleh, S. Muhaidat, M. Debbah, and M. A. Imran, "Edge-native intelligence for 6g communications driven by federated learning: a survey of trends and challengesf," IEEE Transactions on Emerging Topics in Computational Intelligence, 2023.

[34] G. Xia, J. Chen, C. Yu, and J. Ma, "Poisoning attacks in federated learning: A survey," IEEE Access, vol. 11, pp. 10 708–10 722, 2023.

[35] W. Issa, N. Moustafa, B. Turnbull, N. Sohrabi, and Z. Tari, "Blockchain-based federated learning for securing internet of things: A comprehensive survey," ACM Computing Surveys, vol. 55, no. 9, pp. 1–43, 2023.

[36] J. Zhu, J. Cao, D. Saxena, S. Jiang, and H. Ferradi, "Blockchain-empowered federated learning: Challenges, solutions, and future directions," ACM Computing Surveys, vol. 55, no. 11, pp. 1–31, 2023.

[37] M. Mitev, A. Chorti, H. V. Poor, and G. Fettweis, "What physical layer security can do for 6g security," IEEE Open Journal of Vehicular Technology, 2023.

[38] Z. M. Fadlullah, B. Mao, and N. Kato, "Balancing qos and security in the edge: Existing practices, challenges, and 6g opportunities with machine learning," IEEE Communications Surveys & Tutorials, 2022.

[39] Y. Sun, J. Liu, J. Wang, Y. Cao, and N. Kato, "When machine learning meets privacy in 6g: A survey," IEEE Communications Surveys & Tutorials, vol. 22, no. 4, pp. 2694–2724, 2020.

[40] N. Kato, B. Mao, F. Tang, Y. Kawamoto, and J. Liu, "Ten challenges in advancing machine learning technologies toward 6g," IEEE Wireless Communications, vol. 27, no. 3, pp. 96–103, 2020.

[41] U. Demirhan and A. Alkhateeb, "Integrated sensing and communication for 6g: Ten key machine learning roles," IEEE Communications Magazine, 2023.

[42] L. Xiao, X. Wan, X. Lu, Y. Zhang, and D. Wu, "Iot security techniques based on machine learning: How do iot devices use ai to enhance security?" IEEE Signal Processing Magazine, vol. 35, no. 5, pp. 41–49, 2018.

[43] S. M. Tahsien, H. Karimipour, and P. Spachos, "Machine learning based solutions for security of internet of things (iot): A survey," Journal of Network and Computer Applications, vol. 161, p. 102630, 2020.

[44] V.-L. Nguyen, P.-C. Lin, B.-C. Cheng, R.-H. Hwang, and Y.-D. Lin, "Security and privacy for 6g: A survey on prospective technologies and challenges," IEEE Communications Surveys & Tutorials, vol. 23, no. 4, pp. 2384–2428, 2021.

[45] Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu, and V. C. Leung, "A survey on security threats and defensive techniques of machine learning: A data driven view," IEEE access, vol. 6, pp. 12 103–12 117, 2018.

[46] M. Xue, C. Yuan, H. Wu, Y. Zhang, and W. Liu, "Machine learning security: Threats, countermeasures, and evaluations," IEEE Access, vol. 8, pp. 74 720–74 742, 2020.

[47] Y. Hu, W. Kuang, Z. Qin, K. Li, J. Zhang, Y. Gao, W. Li, and K. Li, "Artificial intelligence security: threats and countermeasures," ACM Computing Surveys (CSUR), vol. 55, no. 1, pp. 1–36, 2021.

[48] J. Liu, M. Nogueira, J. Fernandes, and B. Kantarci, "Adversarial machine learning: A multilayer review of the state-of-the-art and challenges for wireless and mobile systems," IEEE Communications Surveys & Tutorials, vol. 24, no. 1, pp. 123–159, 2022.

[49] J. Hao and Y. Tao, "Adversarial attacks on deep learning models in smart grids," Energy Reports, vol. 8, pp. 123–129, 2022.

[50] Z. Chen, J. Liu, Y. Shen, M. Simsek, B. Kantarci, H. T. Mouftah, and P. Djukic, "Machine learning-enabled iot security: Open issues and challenges under advanced persistent threats," ACM Computing Surveys (CSUR), 2022.

[51] M. Al-Rubaie and J. M. Chang, "Privacy-preserving machine learning: Threats and solutions," IEEE Security & Privacy, vol. 17, no. 2, pp. 49–58, 2019.

[52] P. Liu, X. Xu, and W. Wang, "Threats, attacks and defenses to federated learning: issues, taxonomy and perspectives," Cybersecurity, vol. 5, no. 1, pp. 1–19, 2022.

[53] M. S. Jere, T. Farnan, and F. Koushanfar, "A taxonomy of attacks on federated learning," IEEE Security & Privacy, vol. 19, no. 2, pp. 20–28, 2020.

[54] M. Chen, D. Gündüz, K. Huang, W. Saad, M. Bennis, A. V. Feljan, and H. V. Poor, "Distributed learning in wireless networks: Recent progress and future challenges," IEEE Journal on Selected Areas in Communications, 2021.

[55] J. He, S. Guo, M. Li, and Y. Zhu, "Acefl: Federated learning accelerating in 6g-enabled mobile edge computing networks," IEEE Transactions on Network Science and Engineering, 2022.

[56] 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. 40 281–40 306, 2022.

[57] T. Gong, I. Vinieratou, R. Ji, C. Huang, G. C. Alexandropoulos, L. Wei, Z. Zhang, M. Debbah, H. V. Poor, and C. Yuen, "Holographic mimo communications: Theoretical foundations, enabling technologies, and future directions," arXiv preprint arXiv:2212.01257, 2022.

[58] S. Naser, L. Bariah, S. Muhaidat, P. C. Sofotasios, M. Al-Qutayri, E. Damiani, and M. Debbah, "Toward federated-learning-enabled visible light communication in 6g systems," IEEE Wireless Communications, vol. 29, no. 1, pp. 48–56, 2022.

[59] G. C. Alexandropoulos, A. Mokh, R. Khayatzadeh, J. de Rosny, M. Kamoun, A. Ourir, A. Tourin, M. Fink, and M. Debbah, "Time reversal for 6g spatiotemporal focusing: Recent experiments, opportunities, and challenges," IEEE Vehicular Technology Magazine, 2022.

[60] L. Wei, C. Huang, G. C. Alexandropoulos, E. Wei, Z. Zhang, M. Debbah, and C. Yuen, "Multi-user wireless communications with holographic mimo surfaces: A convenient channel model and spectral efficiency analysis," in 2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit). IEEE, 2022, pp. 488–493.

[61] C. Chaccour, M. N. Soorki, W. Saad, M. Bennis, P. Popovski, and M. Debbah, "Seven defining features of terahertz (thz) wireless systems: A fellowship of communication and sensing," IEEE Communications Surveys & Tutorials, vol. 24, no. 2, pp. 967–993, 2022.

[62] M. Mukherjee, M. A. Ferrag, L. Maglaras, A. Derhab, and M. Aazam, "Security and privacy issues and solutions for fog," Fog and fogonomics: challenges and practices of fog computing, communication, networking, strategy, and economics, pp. 353–374, 2020.

[63] S. Iftikhar, S. S. Gill, C. Song, M. Xu, M. S. Aslanpour, A. N. Toosi, J. Du, H. Wu, S. Ghosh, D. Chowdhury et al., "Ai-based fog and edge computing: A systematic review, taxonomy and future directions," Internet of Things, p. 100674, 2022.

[64] S. Iftikhar, M. M. M. Ahmad, S. Tuli, D. Chowdhury, M. Xu, S. S. Gill, and S. Uhlig, "Hunterplus: Ai based energy-efficient task scheduling for cloud–fog computing environments," Internet of Things, vol. 21, p. 100667, 2023.

[65] C. Liu, C. Liu, Y. Shang, S. Chen, B. Cheng, and J. Chen, "An adaptive prediction approach based on workload pattern discrimination in the cloud," Journal of Network and Computer Applications, vol. 80, pp. 35–44, 2017.

[66] "Nvidia edge ai chips," accessed: 12-03-2023. [Online]. Available: https://www.nvidia.com/en-in/deep-learning-ai/solutions/ai-at-the-edge/

[67] "Google cloud iot edge," accessed: 12-03-2023. [Online]. Available: https://cloud.google.com/blog/products/gcp/bringing-intelligence-edge-cloud-iot

[68] F. Morishita, N. Kato, S. Okubo, T. Toi, M. Hiraki, S. Otani, H. Abe, Y. Shinohara, and H. Kondo, "A cmos image sensor and an ai accelerator for realizing edge-computing-based surveillance camera systems," in 2021 Symposium on VLSI Circuits. IEEE, 2021, pp. 1–2.

[69] "Philips rpm," accessed: 12-03-2023. [Online]. Available: https://www.usa.philips.com/healthcare/services/population-health-management/patient-engagement/remote-patient-monitoring

[70] "Ge - predix platform," accessed: 12-03-2023. [Online]. Available: https://www.ge.com/digital/iiot-platform

[71] P. Joshi, M. Hasanuzzaman, C. Thapa, H. Afli, and T. Scully, "Enabling all in-edge deep learning: A literature review," IEEE Access, 2023.

[72] T. Cui, R. Yang, C. Fang, and S. Yu, "Deep reinforcement learning-based resource allocation for content distribution in iot-edge-cloud computing environments," Symmetry, vol. 15, no. 1, p. 217, 2023.

[73] G. Pan, H. Zhang, S. Xu, S. Zhang, and X. Chen, "Joint optimization of video-based ai inference tasks in mec-assisted augmented reality systems," IEEE Transactions on Cognitive Communications and Networking, 2023.

[74] N. Cheng, F. Lyu, W. Quan, C. Zhou, H. He, W. Shi, and X. Shen, "Space/aerial-assisted computing offloading for iot applications: A learning-based approach," IEEE Journal on Selected Areas in Communications, vol. 37, no. 5, pp. 1117–1129, 2019.

[75] "5g-transformer," accessed: 12-03-2023. [Online]. Available: https://5g-ppp.eu/5g-transformer/

[76] "5g mobile network architecture," accessed: 12-03-2023. [Online]. Available: https://5g-monarch.eu/

[77] "H2020 project 5g-picture," accessed: 12-03-2023. [Online]. Available: https://www.5g-picture-project.eu/

[78] "5g-coral: A 5g convergent virtualised radio access network living at the edge," accessed: 12-03-2023. [Online]. Available: https://cordis.europa.eu/project/id/761586

[79] "5g-empower project," accessed: 12-03-2023. [Online]. Available: https://5g-empower.io/

[80] "6g flagship project," accessed: 12-03-2023. [Online]. Available: https://www.6gflagship.com/

[81] "5g open innovation lab," accessed: 12-03-2023. [Online]. Available: https://www.5goilab.com/about/

[82] "Encqor 5g," accessed: 12-03-2023. [Online]. Available: https://www.encqor.ca/

[83] "5g city," accessed: 12-03-2023. [Online]. Available: https://www.5gcity.eu/

[84] "5g-acia," accessed: 12-03-2023. [Online]. Available: https://5g-acia.org/

[85] "6g-platform," accessed: 12-03-2023. [Online]. Available: https://www.6g-platform.com/

[86] K. B. Letaief, Y. Shi, J. Lu, and J. Lu, "Edge artificial intelligence for 6g: Vision, enabling technologies, and applications," IEEE Journal on Selected Areas in Communications, vol. 40, no. 1, pp. 5–36, 2021.

[87] S. Ali, W. Saad, N. Rajatheva, K. Chang, D. Steinbach, B. Sliwa, C. Wietfeld, K. Mei, H. Shiri, H.-J. Zepernick et al., "6g white paper on machine learning in wireless communication networks," arXiv preprint arXiv:2004.13875, 2020.

[88] A. Chorti, A. N. Barreto, S. Köpsell, M. Zoli, M. Chafii, P. Sehier, G. Fettweis, and H. V. Poor, "Context-aware security for 6g wireless: the role of physical layer security," IEEE Communications Standards Magazine, vol. 6, no. 1, pp. 102–108, 2022.

[89] K. B. Letaief, W. Chen, Y. Shi, J. Zhang, and Y.-J. A. Zhang, "The roadmap to 6g: Ai empowered wireless networks," IEEE communications magazine, vol. 57, no. 8, pp. 84–90, 2019.

[90] Z. Yang, M. Chen, K.-K. Wong, H. V. Poor, and S. Cui, "Federated learning for 6g: Applications, challenges, and opportunities," Engineering, 2021.

[91] J. Park, S. Samarakoon, H. Shiri, M. K. Abdel-Aziz, T. Nishio, A. Elgabli, and M. Bennis, "Extreme urllc: Vision, challenges, and key enablers," arXiv preprint arXiv:2001.09683, 2020.

[92] S. A. Abdel Hakeem, H. H. Hussein, and H. Kim, "Security requirements and challenges of 6g technologies and applications," Sensors, vol. 22, no. 5, p. 1969, 2022.

[93] C.-X. Wang, X. You, X. Gao, X. Zhu, Z. Li, C. Zhang, H. Wang, Y. Huang, Y. Chen, H. Haas et al., "On the road to 6g: Visions, requirements, key technologies and testbeds," arXiv preprint arXiv:2302.14536, 2023.

[94] M. U. A. Siddiqui, H. Abumarshoud, L. Bariah, S. Muhaidat, M. A. Imran, and L. Mohjazi, "Urllc in beyond 5g and 6g networks: An interference management perspective," IEEE Access, 2023.

[95] L.-H. Shen, K.-T. Feng, and L. Hanzo, "Five facets of 6g: Research challenges and opportunities," ACM Computing Surveys, vol. 55, no. 11, pp. 1–39, 2023.

[96] J. Wang, J. Liu, J. Li, and N. Kato, "Artificial intelligence-assisted network slicing: Network assurance and service provisioning in 6g," IEEE Vehicular Technology Magazine, 2023.

[97] X. Zhou, M. Bilal, R. Dou, J. J. Rodrigues, Q. Zhao, J. Dai, and X. Xu, "Edge computation offloading with content caching in 6g-enabled iov," IEEE Transactions on Intelligent Transportation Systems, 2023.

[98] C. Han, Y. Wang, Y. Li, Y. Chen, N. A. Abbasi, T. Kürner, and A. F. Molisch, "Terahertz wireless channels: A holistic survey on measurement, modeling, and analysis," IEEE Communications Surveys & Tutorials, vol. 24, no. 3, pp. 1670–1707, 2022.

[99] Y. Liu, Y. Deng, A. Nallanathan, and J. Yuan, "Machine learning for 6g enhanced ultra-reliable and low-latency services," IEEE Wireless Communications, vol. 30, no. 2, pp. 48–54, 2023.

[100] X. Zhu and C. Jiang, "Creating efficient integrated satellite-terrestrial networks in the 6g era," IEEE Wireless Communications, vol. 29, no. 4, pp. 154–160, 2022.

[101] A. Behravan, V. Yajnanarayana, M. F. Keskin, H. Chen, D. Shrestha, T. E. Abrudan, T. Svensson, K. Schindhelm, A. Wolfgang, S. Lindberg et al., "Positioning and sensing in 6g: Gaps, challenges, and opportunities," IEEE Vehicular Technology Magazine, 2022.

[102] M. Al-Ali and E. Yaacoub, "Resource allocation scheme for embb and urllc coexistence in 6g networks," Wireless Networks, pp. 1–20, 2023.

[103] C. She, T. Q. Duong, T. Q. Quek, H. Viswanathan, and D. Lopez-Perez, "Guest editorial: Intelligent ultra-reliable and low-latency communications in 6g," IEEE Wireless Communications, vol. 30, no. 2, pp. 12–13, 2023.

[104] C.-W. Hsu and H.-S. Kim, "Hyper-dimensional modulation for robust short packets in massive machine-type communications," IEEE Transactions on Communications, vol. 71, no. 3, pp. 1388–1402, 2023.

[105] J. Huang, F. Yang, C. Chakraborty, Z. Guo, H. Zhang, L. Zhen, and K. Yu, "Opportunistic capacity based resource allocation for 6g wireless systems with network slicing," Future Generation Computer Systems, vol. 140, pp. 390–401, 2023.

[106] W. Xu, Z. Yang, D. W. K. Ng, M. Levorato, Y. C. Eldar, and M. Debbah, "Edge learning for b5g networks with distributed signal processing: Semantic communication, edge computing, and wireless sensing," IEEE journal of selected topics in signal processing, vol. 17, no. 1, pp. 9–39, 2023.

[107] H.-J. Song and N. Lee, "Terahertz communications: Challenges in the next decade," IEEE Transactions on Terahertz Science and Technology, vol. 12, no. 2, pp. 105–117, 2021.

[108] M. A. Ferrag, M. Debbah, and M. Al-Hawawreh, "Generative ai for cyber threat-hunting in 6g-enabled iot networks," arXiv preprint arXiv:2303.11751, 2023.

[109] H. Yu, T. Taleb, K. Samdanis, and J. Song, "Towards supporting holographic services over deterministic 6g integrated terrestrial & non-terrestrial networks," IEEE Network, 2023.

[110] A. Alkhateeb, G. Charan, T. Osman, A. Hredzak, J. Morais, U. Demirhan, and N. Srinivas, "Deepsense 6g: A large-scale real-world multi-modal sensing and communication dataset," IEEE Communications Magazine, 2023.

[111] O. Friha and M. A. Ferrag, "Blockchain technology for 6g communication networks: A vision for the future," in Cybersecurity Issues in Emerging Technologies. CRC Press, 2021, pp. 77–96.

[112] N. Docomo, "White paper: 5g evolution and 6g," Tech. Rep.

[113] ITU-R, "Imt traffic estimates for the years 2020 to 2030," Tech. Rep.

[114] U. Cisco, "Cisco annual internet report (2018–2023) white paper," Cisco: San Jose, CA, USA, 2020.

[115] T. B. Ahammed, R. Patgiri, and S. Nayak, "A vision on the artificial intelligence for 6g communication," ICT Express, vol. 9, no. 2, pp. 197–210, 2023.

[116] W. Wu, C. Zhou, M. Li, H. Wu, H. Zhou, N. Zhang, X. S. Shen, and W. Zhuang, "Ai-native network slicing for 6g networks," IEEE Wireless Communications, vol. 29, no. 1, pp. 96–103, 2022.

[117] J. Pei, S. Li, Z. Yu, L. Ho, W. Liu, and L. Wang, "Federated learning encounters 6g wireless communication in the scenario of internet of things," IEEE Communications Standards Magazine, vol. 7, no. 1, pp. 94–100, 2023.

[118] J. Ye, L. Xiang, and X. Ge, "Spatial-temporal modeling and analysis of reliability and delay in urban v2x networks," IEEE Transactions on Network Science and Engineering, 2023.

[119] E. Peltonen, M. Bennis, M. Capobianco, M. Debbah, A. Ding, F. Gil-Castiñeira, M. Jurmu, T. Karvonen, M. Kelanti, A. Kliks et al., "6g white paper on edge intelligence," arXiv preprint arXiv:2004.14850, 2020.

[120] Q. N. Le, L. Bariah, O. A. Dobre, and S. Muhaidat, "Reconfigurable intelligent surface-enabled federated learning for power-constrained devices," IEEE Communications Letters, vol. 26, no. 11, pp. 2725–2729, 2022.

[121] S. Zarandi and H. Tabassum, "Federated double deep q-learning for joint delay and energy minimization in iot networks," in 2021 IEEE International Conference on Communications Workshops (ICC Workshops). IEEE, 2021, pp. 1–6.

[122] K. Zheng, G. Jiang, X. Liu, K. Chi, X. Yao, and J. Liu, "Drl-based offloading for computation delay minimization in wireless-powered multi-access edge computing," IEEE Transactions on Communications, 2023.

[123] S. Truex, N. Baracaldo, A. Anwar, T. Steinke, H. Ludwig, R. Zhang, and Y. Zhou, "A hybrid approach to privacy-preserving federated learning," in Proceedings of the 12th ACM workshop on artificial intelligence and security, 2019, pp. 1–11.

[124] C. Zhou, A. Fu, S. Yu, W. Yang, H. Wang, and Y. Zhang, "Privacy-preserving federated learning in fog computing," IEEE Internet of Things Journal, vol. 7, no. 11, pp. 10 782–10 793, 2020.

[125] V. Mothukuri, P. Khare, R. M. Parizi, S. Pouriyeh, A. Dehghantanha, and G. Srivastava, "Federated-learning-based anomaly detection for iot security attacks," IEEE Internet of Things Journal, vol. 9, no. 4, pp. 2545–2554, 2021.

[126] L. Cui, Y. Qu, G. Xie, D. Zeng, R. Li, S. Shen, and S. Yu, "Security and privacy-enhanced federated learning for anomaly detection in iot infrastructures," IEEE Transactions on Industrial Informatics, vol. 18, no. 5, pp. 3492–3500, 2021.

[127] Z. Xu, Y. Guo, C. Chakraborty, Q. Hua, S. Chen, and K. Yu, "A simple federated learning-based scheme for security enhancement over internet of medical things," IEEE Journal of Biomedical and Health Informatics, 2022.

[128] O. Friha, M. A. Ferrag, L. Shu, L. Maglaras, K.-K. R. Choo, and M. Nafaa, "Felids: Federated learning-based intrusion detection system for agricultural internet of things," Journal of Parallel and Distributed Computing, vol. 165, pp. 17–31, 2022.

[129] H. Gao, N. He, and T. Gao, "Sverifl: Successive verifiable federated learning with privacy-preserving," Information Sciences, vol. 622, pp. 98–114, 2023.

[130] L. Ouyang, F.-Y. Wang, Y. Tian, X. Jia, H. Qi, and G. Wang, "Artificial identification: a novel privacy framework for federated learning based on blockchain," IEEE Transactions on Computational Social Systems, 2023.

[131] O. Friha, M. A. Ferrag, M. Benbouzid, T. Berghout, B. Kantarci, and K.-K. R. Choo, "2df-ids: Decentralized and differentially private federated learning-based intrusion detection system for industrial iot," Computers & Security, p. 103097, 2023.

[132] J. Li, X. Tong, J. Liu, and L. Cheng, "An efficient federated learning system for network intrusion detection," IEEE Systems Journal, 2023.

[133] X. You, X. Liu, X. Lin, J. Cai, and S. Chen, "Accuracy degrading: Towards participation-fair federated learning," IEEE Internet of Things Journal, 2023.

[134] M. J. Baucas, P. Spachos, and K. N. Plataniotis, "Federated learning and blockchain-enabled fog-iot platform for wearables in predictive healthcare," IEEE Transactions on Computational Social Systems, 2023.

[135] Z. Abou El Houda, A. S. Hafid, and L. Khoukhi, "Mitfed: A privacy preserving collaborative network attack mitigation framework based on federated learning using sdn and blockchain," IEEE Transactions on Network Science and Engineering, 2023.

[136] Q. Chen, Z. Wang, W. Zhang, and X. Lin, "Ppt: a privacy-preserving global model training protocol for federated learning in p2p networks," Computers & Security, vol. 124, p. 102966, 2023.

[137] M. U. Hassan, M. H. Rehmani, and J. Chen, "Differential privacy techniques for cyber physical systems: a survey," IEEE Communications Surveys & Tutorials, vol. 22, no. 1, pp. 746–789, 2019.

[138] A. Mohammed and R. Kora, "A comprehensive review on ensemble deep learning: Opportunities and challenges," Journal of King Saud University-Computer and Information Sciences, 2023.

[139] A. Dunmore, J. Jang-Jaccard, F. Sabrian, and J. Kwak, "Generative adversarial networks for malware detection: a survey," arXiv preprint arXiv:2302.08558, 2023.

[140] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," Journal of Information Security and Applications, vol. 50, p. 102419, 2020.

[141] C. Marcolla, V. Sucasas, M. Manzano, R. Bassoli, F. H. Fitzek, and N. Aaraj, "Survey on fully homomorphic encryption, theory, and applications," Proceedings of the IEEE, vol. 110, no. 10, pp. 1572–1609, 2022.

[142] M. A. Ferrag and L. Shu, "The performance evaluation of blockchain-based security and privacy systems for the internet of things: A tutorial," IEEE Internet of Things Journal, vol. 8, no. 24, pp. 17 236–17 260, 2021.

[143] W. Rafique, L. Qi, I. Yaqoob, M. Imran, R. U. Rasool, and W. Dou, "Complementing iot services through software defined networking and edge computing: A comprehensive survey," IEEE Communications Surveys & Tutorials, vol. 22, no. 3, pp. 1761–1804, 2020.

[144] L. Sadineni, E. S. Pilli, and R. B. Battula, "Provnet-iot: Provenance based network layer forensics in internet of things," Forensic Science International: Digital Investigation, vol. 43, p. 301441, 2022.

[145] A. Ghourabi, "A security model based on lightgbm and transformer to protect healthcare systems from cyberattacks," IEEE Access, vol. 10, pp. 48 890–48 903, 2022.

[146] C. Chen and J. Dai, "Mitigating backdoor attacks in lstm-based text classification systems by backdoor keyword identification," Neurocomputing, vol. 452, pp. 253–262, 2021.

[147] C. Fung, C. J. Yoon, and I. Beschastnikh, "Mitigating sybils in federated learning poisoning," arXiv preprint arXiv:1808.04866, 2018.

[148] X. Li, Z. Qu, S. Zhao, B. Tang, Z. Lu, and Y. Liu, "Lomar: A local defense against poisoning attack on federated learning," IEEE Transactions on Dependable and Secure Computing, 2021.

[149] Z. Xiang, D. J. Miller, and G. Kesidis, "Reverse engineering imperceptible backdoor attacks on deep neural networks for detection and training set cleansing," Computers & Security, vol. 106, p. 102280, 2021.

[150] T. Zheng and B. Li, "First-order efficient general-purpose clean-label data poisoning," in IEEE INFOCOM 2021-IEEE Conference on Computer Communications. IEEE, 2021, pp. 1–10.

[151] H. Aghakhani, D. Meng, Y.-X. Wang, C. Kruegel, and G. Vigna, "Bullseye polytope: A scalable clean-label poisoning attack with improved transferability," arXiv preprint arXiv:2005.00191, 2020.

[152] X. Zhou, M. Xu, Y. Wu, and N. Zheng, "Deep model poisoning attack on federated learning," Future Internet, vol. 13, no. 3, p. 73, 2021.

[153] J. Zhang, B. Chen, X. Cheng, H. T. T. Binh, and S. Yu, "Poisongan: Generative poisoning attacks against federated learning in edge computing systems," IEEE Internet of Things Journal, vol. 8, no. 5, pp. 3310–3322, 2021.

[154] Y. Guo, Q. Wang, T. Ji, X. Wang, and P. Li, "Resisting distributed backdoor attacks in federated learning: A dynamic norm clipping approach," in 2021 IEEE International Conference on Big Data (Big Data). IEEE, 2021, pp. 1172–1182.

[155] H. Liu, D. Li, and Y. Li, "Poisonous label attack: Black-box data poisoning attack with enhanced conditional dcgan," Neural Processing Letters, vol. 53, no. 6, pp. 4117–4142, 2021.

[156] A. Uprety and D. B. Rawat, "Mitigating poisoning attack in federated learning," in 2021 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE, 2021, pp. 01–07.

[157] D. Cao, S. Chang, Z. Lin, G. Liu, and D. Sun, "Understanding distributed poisoning attack in federated learning," in 2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS). IEEE, 2019, pp. 233–239.

[158] G. Liu, I. Khalil, A. Khreishah, and N. Phan, "A synergetic attack against neural network classifiers combining backdoor and adversarial examples," in 2021 IEEE International Conference on Big Data (Big Data). IEEE, 2021, pp. 834–846.

[159] W. Yang, J. Yuan, X. Wang, and P. Zhao, "Tsadv: Black-box adversarial attack on time series with local perturbations," Engineering Applications of Artificial Intelligence, vol. 114, p. 105218, 2022.

[160] G. Sun, Y. Cong, J. Dong, Q. Wang, L. Lyu, and J. Liu, "Data poisoning attacks on federated machine learning," IEEE Internet of Things Journal, 2022.

[161] J. Chen, L. Zhang, H. Zheng, X. Wang, and Z. Ming, "Deeppoison: Feature transfer based stealthy poisoning attack for dnns," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 68, no. 7, pp. 2618–2622, 2021.

[162] F. Khan, R. L. Kumar, M. H. Abidi, S. Kadry, H. Alkhalefah, and M. K. Aboudaif, "Federated split learning model for industry 5.0: A data poisoning defense for edge computing," Electronics, vol. 11, no. 15, p. 2393, 2022.

[163] M. Biolková and B. Nguyen, "Neural predictor for black-box adversarial attacks on speech recognition," arXiv preprint arXiv:2203.09849, 2022.

[164] M. Fang, X. Cao, J. Jia, and N. Gong, "Local model poisoning attacks to {Byzantine-Robust} federated learning," in 29th USENIX Security Symposium (USENIX Security 20), 2020, pp. 1605–1622.

[165] V. Tolpegin, S. Truex, M. E. Gursoy, and L. Liu, "Data poisoning attacks against federated learning systems," in European Symposium on Research in Computer Security. Springer, 2020, pp. 480–501.

[166] X. Xiao, Z. Tang, C. Li, B. Xiao, and K. Li, "Sca: Sybil-based collusion attacks of iiot data poisoning in federated learning," IEEE Transactions on Industrial Informatics, 2022.

[167] G. Severi, M. Jagielski, G. Yar, Y. Wang, A. Oprea, and C. Nita-Rotaru, "Network-level adversaries in federated learning," in 2022 IEEE Conference on Communications and Network Security (CNS). IEEE, 2022, pp. 19–27.

[168] L. Jiang, X. Ma, S. Chen, J. Bailey, and Y.-G. Jiang, "Black-box adversarial attacks on video recognition models," in Proceedings of the 27th ACM International Conference on Multimedia, 2019, pp. 864–872.

[169] J. Ebrahimi, A. Rao, D. Lowd, and D. Dou, "Hotflip: White-box adversarial examples for text classification," arXiv preprint arXiv:1712.06751, 2017.

[170] Y. Wang, J. Liu, X. Chang, J. Mišić, and V. B. Mišić, "Iwa: integrated gradient-based white-box attacks for fooling deep neural networks," International Journal of Intelligent Systems, vol. 37, no. 7, pp. 4253–4276, 2022.

[171] S. Tabatabai, I. Mohammed, B. Qolomany, A. Albaseer, K. Ahmad, M. Abdallah, and A. Al-Fuqaha, "Exploration and exploitation in federated learning to exclude clients with poisoned data," in 2022 International Wireless Communications and Mobile Computing (IWCMC). IEEE, 2022, pp. 407–412.

[172] Z. Zhang, Y. Zhang, D. Guo, L. Yao, and Z. Li, "Secfednids: Robust defense for poisoning attack against federated learning-based network intrusion detection system," Future Generation Computer Systems, vol. 134, pp. 154–169, 2022.

[173] R. Doku and D. B. Rawat, "Mitigating data poisoning attacks on a federated learning-edge computing network," in 2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC). IEEE, 2021, pp. 1–6.

[174] N. M. Jebreel, J. Domingo-Ferrer, D. Sánchez, and A. Blanco-Justicia, "Defending against the label-flipping attack in federated learning," arXiv preprint arXiv:2207.01982, 2022.

[175] J. Zhang, J. Chen, D. Wu, B. Chen, and S. Yu, "Poisoning attack in federated learning using generative adversarial nets," in 2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE). IEEE, 2019, pp. 374–380.

[176] P. Zhao, H. Jiang, J. Li, Z. Xiao, D. Liu, J. Ren, and D. Guo, "Garbage in, garbage out: Poisoning attacks disguised with plausible mobility in data aggregation," IEEE Transactions on Network Science and Engineering, vol. 8, no. 3, pp. 2679–2693, 2021.

[177] J. Wu and J. He, "Indirect invisible poisoning attacks on domain adaptation," in Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 1852–1862.

[178] R. Ning, J. Li, C. Xin, and H. Wu, "Invisible poison: A blackbox clean label backdoor attack to deep neural networks," in IEEE INFOCOM 2021-IEEE Conference on Computer Communications. IEEE, 2021, pp. 1–10.

[179] A. Takiddin, M. Ismail, U. Zafar, and E. Serpedin, "Robust electricity theft detection against data poisoning attacks in smart grids," IEEE Transactions on Smart Grid, vol. 12, no. 3, pp. 2675–2684, 2020.

[180] Y. Qiu, J. Wu, S. Mumtaz, J. Li, A. Al-Dulaimi, and J. J. Rodrigues, "Mt-mtd: Muti-training based moving target defense trojaning attack in edged-ai network," in ICC 2021-IEEE International Conference on Communications. IEEE, 2021, pp. 1–6.

[181] G. Apruzzese and V. Subrahmanian, "Mitigating adversarial gray-box attacks against phishing detectors," IEEE Transactions on Dependable and Secure Computing, 2022.

[182] "Kdd cup 1999," http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. last accessed 04 November 2022.

[183] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in 2009 IEEE symposium on computational intelligence for security and defense applications. Ieee, 2009, pp. 1–6.

[184] N. Moustafa and J. Slay, "Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set)," in 2015 military communications and information systems conference (MilCIS). IEEE, 2015, pp. 1–6.

[185] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization." in ICISSP, 2018, pp. 108–116.

[186] "Ton_iot datasets," https://ieee-dataport.org/documents/toniot-datasets, last accessed 3 March 2021.

[187] I. Corona, B. Biggio, M. Contini, L. Piras, R. Corda, M. Mereu, G. Mureddu, D. Ariu, and F. Roli, "Deltaphish: Detecting phishing webpages in compromised websites," in European Symposium on Research in Computer Security. Springer, 2017, pp. 370–388.

[188] "The mnist database of handwritten digits." .http://yann.lecun.com/exdb/mnist/, last accessed 11-2022.

[189] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," arXiv preprint arXiv:1708.07747, 2017.

[190] S. Caldas, S. M. K. Duddu, P. Wu, T. Li, J. Konečnỳ, H. B. McMahan, V. Smith, and A. Talwalkar, "Leaf: A benchmark for federated settings," arXiv preprint arXiv:1812.01097, 2018.

[191] A. Krizhevsky, G. Hinton et al., "Learning multiple layers of features from tiny images," 2009.

[192] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein et al., "Imagenet large scale visual recognition challenge," International journal of computer vision, vol. 115, no. 3, pp. 211–252, 2015.

[193] A. Khosla, N. Jayadevaprakash, B. Yao, and F.-F. Li, "Novel dataset for fine-grained image categorization: Stanford dogs," in Proc. CVPR workshop on fine-grained visual categorization (FGVC), vol. 2, no. 1. Citeseer, 2011.

[194] W. Wolberg, N. Street, and O. Mangasarian, "Uci machine learning repository: breast cancer wisconsin (diagnostic) data set," 2011.

[195] A. Cambridge, "The database of faces," 2002, http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.

[196] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in Workshop on faces in'Real-Life'Images: detection, alignment, and recognition, 2008.

[197] S. Li, D. Yi, Z. Lei, and S. Liao, "The casia nir-vis 2.0 face database," in Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2013, pp. 348–353.

[198] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "Vggface2: A dataset for recognising faces across pose and age," in 2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018). IEEE, 2018, pp. 67–74.

[199] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in Proceedings of the IEEE international conference on computer vision, 2015, pp. 3730–3738.

[200] K. Nazeri, E. Ng, T. Joseph, F. Z. Qureshi, and M. Ebrahimi, "Edgeconnect: Generative image inpainting with adversarial edge learning," arXiv preprint arXiv:1901.00212, 2019.

[201] Y. Chen, E. Keogh, B. Hu, N. Begum, A. Bagnall, A. Mueen, and G. Batista, "The ucr time series classification archive," July 2015, www.cs.ucr.edu/~eamonn/time_series_data/.

[202] D. Anguita, A. Ghio, L. Oneto, X. Parra Perez, and J. L. Reyes Ortiz, "A public domain dataset for human activity recognition using smartphones," in Proceedings of the 21th international European symposium on artificial neural networks, computational intelligence and machine learning, 2013, pp. 437–442.

[203] M. Little, P. Mcsharry, S. Roberts, D. Costello, and I. Moroz, "Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection," Nature Precedings, pp. 1–1, 2007.

[204] A. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies, 2011, pp. 142–150.

[205] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, "Adaptive recursive neural network for target-dependent twitter sentiment classification," in Proceedings of the 52nd annual meeting of the association for computational linguistics (volume 2: Short papers), 2014, pp. 49–54.

[206] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The german traffic sign recognition benchmark: a multi-class classification competition," in The 2011 international joint conference on neural networks. IEEE, 2011, pp. 1453–1460.

[207] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 6, pp. 1452–1464, 2017.

[208] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer et al., "Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia," Semantic web, vol. 6, no. 2, pp. 167–195, 2015.

[209] I. Arshad, M. N. Asghar, Y. Qiao, B. Lee, and Y. Ye, "Pixdoor: A pixel-space backdoor attack on deep learning models," in 2021 29th European Signal Processing Conference (EUSIPCO). IEEE, 2021, pp. 681–685.

[210] B. Wang, Y. Yao, S. Shan, H. Li, B. Viswanath, H. Zheng, and B. Y. Zhao, "Neural cleanse: Identifying and mitigating backdoor attacks in neural networks," in 2019 IEEE Symposium on Security and Privacy (SP). IEEE, 2019, pp. 707–723.

[211] Y. Gao, C. Xu, D. Wang, S. Chen, D. C. Ranasinghe, and S. Nepal, "Strip: A defence against trojan attacks on deep neural networks," in Proceedings of the 35th Annual Computer Security Applications Conference, 2019, pp. 113–125.

[212] H. Wang, S. Wang, Z. Jin, Y. Wang, C. Chen, and M. Tistarelli, "Similarity-based gray-box adversarial attack against deep face recognition," in 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021). IEEE, 2021, pp. 1–8.

[213] S. Venkatesan, H. Sikka, R. Izmailov, R. Chadha, A. Oprea, and M. J. De Lucia, "Poisoning attacks and data sanitization mitigations for machine learning models in network intrusion detection systems," in MILCOM 2021-2021 IEEE Military Communications Conference (MILCOM). IEEE, 2021, pp. 874–879.

[214] Y. E. Sagduyu, Y. Shi, and T. Erpek, "Adversarial deep learning for over-the-air spectrum poisoning attacks," IEEE Transactions on Mobile Computing, vol. 20, no. 2, pp. 306–319, 2021.

[215] I. Rosenberg, A. Shabtai, Y. Elovici, and L. Rokach, "Adversarial machine learning attacks and defense methods in the cyber security domain," ACM Computing Surveys (CSUR), vol. 54, no. 5, pp. 1–36, 2021.

[216] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," in International Conference on Artificial Intelligence and Statistics. PMLR, 2020, pp. 2938–2948.

[217] M. A. Ferrag, B. Kantarci, L. C. Cordeiro, M. Debbah, and K.-K. R. Choo, "Poisoning attacks in federated edge learning for digital twin 6g-enabled iots: An anticipatory study," 2023.

[218] X. Li, K. Qiu, C. Qian, and G. Zhao, "An adversarial machine learning method based on opcode n-grams feature in malware detection," in 2020 IEEE Fifth International Conference on Data Science in Cyberspace (DSC). IEEE, 2020, pp. 380–387.

[219] S. Lee, H. Kim, and J. Lee, "Graddiv: Adversarial robustness of randomized neural networks via gradient diversity regularization," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022.

[220] M. Cisse, Y. Adi, N. Neverova, and J. Keshet, "Houdini: Fooling deep structured prediction models," arXiv preprint arXiv:1707.05373, 2017.

[221] S. Sarkar, A. Bansal, U. Mahbub, and R. Chellappa, "Upset and angri: Breaking high performance image classifiers," arXiv preprint arXiv:1707.01159, 2017.

[222] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in 2017 ieee symposium on security and privacy (sp). Ieee, 2017, pp. 39–57.

[223] J. Su, D. V. Vargas, and K. Sakurai, "One pixel attack for fooling deep neural networks," IEEE Transactions on Evolutionary Computation, vol. 23, no. 5, pp. 828–841, 2019.

[224] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in 2016 IEEE European symposium on security and privacy (EuroS&P). IEEE, 2016, pp. 372–387.

[225] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in Artificial intelligence safety and security. Chapman and Hall/CRC, 2018, pp. 99–112.

[226] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," arXiv preprint arXiv:1412.6572, 2014.

[227] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," arXiv preprint arXiv:1312.6199, 2013.

[228] A. Rozsa, M. Günther, E. M. Rudd, and T. E. Boult, "Facial attributes: Accuracy and adversarial robustness," Pattern Recognition Letters, vol. 124, pp. 100–108, 2019.

[229] J. Hendrik Metzen, M. Chaithanya Kumar, T. Brox, and V. Fischer, "Universal adversarial perturbations against semantic image segmentation," in Proceedings of the IEEE international conference on computer vision, 2017, pp. 2755–2764.

[230] Y.-C. Lin, Z.-W. Hong, Y.-H. Liao, M.-L. Shih, M.-Y. Liu, and M. Sun, "Tactics of adversarial attack on deep reinforcement learning agents," arXiv preprint arXiv:1703.06748, 2017.

[231] N. Papernot, P. McDaniel, A. Swami, and R. Harang, "Crafting adversarial input sequences for recurrent neural networks," in MILCOM 2016-2016 IEEE Military Communications Conference. IEEE, 2016, pp. 49–54.

[232] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," Ieee Access, vol. 6, pp. 14 410–14 430, 2018.

[233] B. Ru, A. Cobb, A. Blaas, and Y. Gal, "Bayesopt adversarial attack," in International Conference on Learning Representations, 2019.

[234] M. Cheng, S. Singh, P. Chen, P.-Y. Chen, S. Liu, and C.-J. Hsieh, "Sign-opt: A query-efficient hard-label adversarial attack," arXiv preprint arXiv:1909.10773, 2019.

[235] W. Brendel and M. Bethge, "Approximating cnns with bag-of-local-features models works surprisingly well on imagenet," arXiv preprint arXiv:1904.00760, 2019.

[236] P.-Y. Chen and P. Das, "Ai maintenance: A robustness perspective," arXiv preprint arXiv:2301.03052, 2023.

[237] P. Zhao, P.-Y. Chen, P. Das, K. N. Ramamurthy, and X. Lin, "Bridging mode connectivity in loss landscapes and adversarial robustness," arXiv preprint arXiv:2005.00060, 2020.

[238] N. Papernot, P. McDaniel, and I. Goodfellow, "Transferability in machine learning: from phenomena to black-box attacks using adversarial samples," arXiv preprint arXiv:1605.07277, 2016.

[239] M. Fang, M. Sun, Q. Li, N. Z. Gong, J. Tian, and J. Liu, "Data poisoning attacks and defenses to crowdsourcing systems," in Proceedings of the Web Conference 2021, 2021, pp. 969–980.

[240] M. Kravchik, B. Biggio, and A. Shabtai, "Poisoning attacks on cyber attack detectors for industrial control systems," in Proceedings of the 36th Annual ACM Symposium on Applied Computing, 2021, pp. 116–125.

[241] 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 of Things Journal, vol. 9, no. 5, pp. 3962–3977, 2021.

[242] H. Zhang, J. Gao, and L. Su, "Data poisoning attacks against outcome interpretations of predictive models," in Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 2165–2173.

[243] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," Advances in Neural Information Processing Systems, vol. 30, 2017.

[244] Y. Qu, S. R. Pokhrel, S. Garg, L. Gao, and Y. Xiang, "A blockchained federated learning framework for cognitive computing in industry 4.0 networks," IEEE Transactions on Industrial Informatics, vol. 17, no. 4, pp. 2964–2973, 2021.

[245] Q. Wang, Y. Guo, L. Yu, X. Chen, and P. Li, "Deep q-network-based feature selection for multisourced data cleaning," IEEE Internet of Things Journal, vol. 8, no. 21, pp. 16 153–16 164, 2021.

[246] S. Wolf, W. Gamboa, and M. Borowczak, "Jangseung: A guardian for machine learning algorithms to protect against poisoning attacks," in 2021 IEEE International Smart Cities Conference (ISC2). IEEE, 2021, pp. 1–7.

[247] D. Li, W. E. Wong, W. Wang, Y. Yao, and M. Chau, "Detection and mitigation of label-flipping attacks in federated learning systems with kpca and k-means," in 2021 8th International Conference on Dependable Systems and Their Applications (DSA). IEEE, 2021, pp. 551–559.

[248] P. P. Chan, F. Luo, Z. Chen, Y. Shu, and D. S. Yeung, "Transfer learning based countermeasure against label flipping poisoning attack," Information Sciences, vol. 548, pp. 450–460, 2021.

[249] J. W. Stokes, P. England, and K. Kane, "Preventing machine learning poisoning attacks using authentication and provenance," in MILCOM 2021-2021 IEEE Military Communications Conference (MILCOM). IEEE, 2021, pp. 181–188.

[250] Z. Shi, X. Ding, F. Li, Y. Chen, and C. Li, "Mitigation of poisoning attack in federated learning by using historical distance detection," in 2021 5th Cyber Security in Networking Conference (CSNet). IEEE, 2021, pp. 10–17.

[251] S. Andreina, G. A. Marson, H. Möllering, and G. Karame, "Baffle: Backdoor detection via feedback-based federated learning," in 2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS). IEEE, 2021, pp. 852–863.

[252] Z. Gu and Y. Yang, "Detecting malicious model updates from federated learning on conditional variational autoencoder," in 2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS). IEEE, 2021, pp. 671–680.

[253] Y. Liu, Y. Qu, C. Xu, Z. Hao, and B. Gu, "Blockchain-enabled asynchronous federated learning in edge computing," Sensors, vol. 21, no. 10, p. 3335, 2021.

[254] M. Shayan, C. Fung, C. J. Yoon, and I. Beschastnikh, "Biscotti: A blockchain system for private and secure federated learning," IEEE Transactions on Parallel and Distributed Systems, vol. 32, no. 7, pp. 1513–1525, 2021.

[255] F. Xie, Y. Gao, J. Wang, and W. Zhao, "Defending local poisoning attacks in multi-party learning via immune system," Knowledge-Based Systems, vol. 238, p. 107850, 2022.

[256] Z. Chen, H. Cui, E. Wu, and X. Yu, "Dynamic asynchronous anti poisoning federated deep learning with blockchain-based reputation-aware solutions," Sensors, vol. 22, no. 2, p. 684, 2022.

[257] B. Hou, J. Gao, X. Guo, T. Baker, Y. Zhang, Y. Wen, and Z. Liu, "Mitigating the backdoor attack by federated filters for industrial iot applications," IEEE Transactions on Industrial Informatics, 2022.

[258] Y. Wang, D.-H. Zhai, Y. He, and Y. Xia, "An adaptive robust defending algorithm against backdoor attacks in federated learning," Future Generation Computer Systems, vol. 143, pp. 118–131, 2023.

[259] Y. Jiang, W. Zhang, and Y. Chen, "Data quality detection mechanism against label flipping attacks in federated learning," IEEE Transactions on Information Forensics and Security, 2023.

[260] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in Proceedings of the 22nd ACM SIGSAC conference on computer and communications security, 2015, pp. 1322–1333.

[261] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in 2017 IEEE symposium on security and privacy (SP). IEEE, 2017, pp. 3–18.

[262] X. Tang, S. Mahloujifar, L. Song, V. Shejwalkar, M. Nasr, A. Houmansadr, and P. Mittal, "Mitigating membership inference attacks by {Self-Distillation} through a novel ensemble architecture," in 31st USENIX Security Symposium (USENIX Security 22), 2022, pp. 1433–1450.

[263] D. Unal, M. Hammoudeh, M. A. Khan, A. Abuarqoub, G. Epiphaniou, and R. Hamila, "Integration of federated machine learning and blockchain for the provision of secure big data analytics for internet of things," Computers & Security, vol. 109, p. 102393, 2021.

[264] D. P. Kuttichira, S. Gupta, D. Nguyen, S. Rana, and S. Venkatesh, "Verification of integrity of deployed deep learning models using bayesian optimization," Knowledge-Based Systems, p. 108238, 2022.

[265] M. Hao, H. Li, X. Luo, G. Xu, H. Yang, and S. Liu, "Efficient and privacy-enhanced federated learning for industrial artificial intelligence," IEEE Transactions on Industrial Informatics, vol. 16, no. 10, pp. 6532–6542, 2019.

[266] A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, and M. Backes, "Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models," arXiv preprint arXiv:1806.01246, 2018.

[267] M. K. Shehzad, L. Rose, M. M. Butt, I. Z. Kovacs, M. Assaad, and M. Guizani, "Artificial intelligence for 6g networks: Technology advancement and standardization," arXiv preprint arXiv:2204.00914, 2022.

[268] M. d'Aquin, P. Troullinou, N. E. O'Connor, A. Cullen, G. Faller, and L. Holden, "Towards an" ethics by design" methodology for ai research projects," in Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, 2018, pp. 54–59.

[269] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," Advances in neural information processing systems, vol. 30, 2017.

[270] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.

[271] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," The Journal of Machine Learning Research, vol. 21, no. 1, pp. 5485–5551, 2020.

[272] K. Ramezanpour, J. Jagannath, and A. Jagannath, "Security and privacy vulnerabilities of 5g/6g and wifi 6: Survey and research directions from a coexistence perspective," Computer Networks, vol. 221, p. 109515, 2023.