Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.053861

REVIEW

# Enhancing Internet of Things Intrusion Detection Using Artificial Intelligence

**Shachar Bar[1], P. W. C. Prasad[2] and Md Shohel Sayeed[3,*]**

[1]School of Computing, Mathematics, Charles Sturt University, Bathurst, NSW 2795, Australia

[2]International School, Duy Tan University, Da Nang, 550000, Vietnam

[3]Faculty of Information Science and Technology, Multimedia University, Melaka, 75450, Malaysia

*Corresponding Author: Md Shohel Sayeed. Email: shohel.sayeed@mmu.edu.my

Received: 12 May 2024 Accepted: 31 July 2024 Published: 15 October 2024

## ABSTRACT

Escalating cyber security threats and the increased use of Internet of Things (IoT) devices require utilisation of the latest technologies available to supply adequate protection. The aim of Intrusion Detection Systems (IDS) is to prevent malicious attacks that corrupt operations and interrupt data flow, which might have significant impact on critical industries and infrastructure. This research examines existing IDS, based on Artificial Intelligence (AI) for IoT devices, methods, and techniques. The contribution of this study consists of identification of the most effective IDS systems in terms of accuracy, precision, recall and F1-score; this research also considers training time. Results demonstrate that Graph Neural Networks (GNN) have several benefits over other traditional AI frameworks through their ability to achieve in excess of 99% accuracy in a relatively short training time, while also capable of learning from network traffic the inherent characteristics of different cyber-attacks. These findings identify the GNN (a Deep Learning AI method) as the most efficient IDS system. The novelty of this research lies also in the linking between high yielding AI-based IDS algorithms and the AI-based learning approach for data privacy protection. This research recommends Federated Learning (FL) as the AI training model, which increases data privacy protection and reduces network data flow, resulting in a more secure and efficient IDS solution.

## KEYWORDS

Anomaly detection; artificial intelligence; cyber security; data privacy; deep learning; federated learning; industrial internet of things; internet of things; intrusion detection system; machine learning

## Nomenclature

| | |
|---|---|
| **AI** | Artificial Intelligence |
| **CL** | Collaborative Learning |
| **CNN** | Convolution Neural Network |
| **DCAE** | Deep Convolutional AutoEncoder |
| **DL** | Deep Learning |
| **DNN** | Deep Neural Network |
| **DoS** | Denial of Service |

| | |
|---|---|
| **DDoS** | Distributed Denial of Service |
| **DT** | Decision Tree |
| **DSF** | Decentralised Secure Framework |
| **EIDM** | Enhanced anomaly-based Intrusion Detection Deep Learning Multi-class |
| **FAR** | False Alarm Rate |
| **FcTH** | Fuzzy color Texture Histogram |
| **FFNN** | Feed Forward Neural network |
| **FL** | Federate Learning |
| **FT** | Fine Tree |
| **GCN** | Graph Convolutional Network |
| **GNN** | Graph Neural Network |
| **GRU** | Gated Recurrent Unit |
| **GSL** | Graph Structure Learning |
| **IDS** | Intrusion Detection System |
| **IoT** | Internet of Things |
| **IDSAI** | Intrusion Detection System Artificial Intelligence |
| **KNN** | K-Nearest Neighbors |
| **LSTM** | Long Short-Term Memory |
| **MItM** | Man In the Middle |
| **ML** | Machine Learning |
| **MT** | Mean Teachers |
| **PFI** | Permutation Feature Importance |
| **RF** | Random Forest |
| **RFE** | Recursive Feature Elimination |
| **RNN** | Random Neural Network |
| **ROC** | Receiver Operating Characteristics |
| **SDN** | Software Defined Network |
| **SHAP** | Shapley Additive Explanation |
| **SPIP** | S: Shapley Additive exPlanations,<br>P: Permutation Feature Importance,<br>I: Individual Conditional Expectation,<br>P: Partial Dependence Plot |
| **SSAE** | Stacked Sparse AutoEncoders |
| **SVM** | Support Vector Machine |
| **TCN** | Temporal Convolutional Network |
| **VGG** | Visual Geometry Group |
| **XAI** | Explainable Artificial Intelligence |

## 1 Introduction

Cyber security threats continue to escalate, particularly for complex systems like the Internet of Things which are widely integrated across industries and are forecast to reach 30 billion devices by 2030 [1]. This calls for a comprehensive approach to the development of robust Detection Intrusion Systems (IDS) [2–5]. Although effective IDS exist, they often focus on specific vulnerabilities rather than taking a holistic approach. An exception is the work of Alzubaidi et al. [6] whose work covered a significant number of vulnerabilities and achieved the highest prediction accuracy. Their research was based on Machine Learning (ML) with Recursive Feature Elimination (RFE) as its selection

method, also highlighting the importance of developing an appropriate preprocessing phase to prepare data for training during the model training phase and identify Random Forest (RF) as achieved the highest prediction accuracy. Liu et al. [7] concurred but point to the importance of accuracy in terms of individual decision trees and the dependencies between them. Lu et al. [8] focused on deep learning algorithms with multilayer network models that allow them to learn data features of data for the identification of anomalous traffic with some accuracy. Alani [9] focused on the selection of appropriate feature reduction algorithms, to increase prediction accuracy, thus reducing training time and computational resources required for the IDS framework. In this research, we also compared filter-, wrapper-, and embedding-based feature selection types. Jing et al. [10] acknowledged the resulting improved detection performance but cautioned that it resulted in increased model training and inference time compared to feature selection. Saika et al. [11] agreed but critiqued that research around this topic had omitted to point out the impact on the entire framework. This brief discussion clearly demonstrates the current state of fragmentation around IDS.

The aim of this work is to evaluate recent research into Artificial Intelligence (AI)-based IDS across a range of cyber-attack types including DDoS [12], Label flipping [13], MiTM [14], and Zero-day exploits [15], etc., to identify all possible points of vulnerability as well as 'best practice' for the defense against attacks based on Prediction accuracy, Training efficiency, Dataset preprocessing, Data privacy, Model performance, and Resource utilization. Outcomes demonstrate that two frameworks were identified as providing the highest prediction accuracy in AI-based IDS systems for IoT devices and networks: the E2I3DS framework and the GNN-based network IDS.

The remainder of this work is structured as follows: Section 2 outlines the research methodology, followed by the literature review in Section 3. Section 4 presents a comprehensive analysis of each contribution included in this research, followed by a discussion of outcomes. Recommendations for future work are made in Section 5 while Section 6 provides the research conclusion.

## 2 Research Methodology

The key terms 'Anomaly Detection; Artificial Intelligence; Cyber Security; Data Privacy; Deep Learning; Federated Learning (FL) [16]; Industrial Internet of things; Internet of things; Intrusion Detection System; and Machine Learning' were entered into a range of journal databases. Results were subsequently filtered for Q1 peer-reviewed articles from 2023 and 2024. A comparison of relevant results was drawn from these works with the aim of finding approaches that provide a combination of high accuracy prediction and precision along with the ability to protect sensitive data, as part of the training model process. The process of gathering the selected journal articles is depicted in Fig. 1.

The final selection included AI methods, i.e., training time and prediction accuracy. Models were tested for their ability to offload high processing and large memory storage requirements of the targeted IoT devices by utilising external servers and cloud-based infrastructure. The above process was repeated until an adequate list of sources had been identified to permit a meaningful comparison.

## 3 Literature Review

The journal articles used for this research project were selected based on a set of keywords showing the main goals of enhancing IoT IDS using AI. The keyword relationships to the selected journal articles are depicted in Fig. 2.

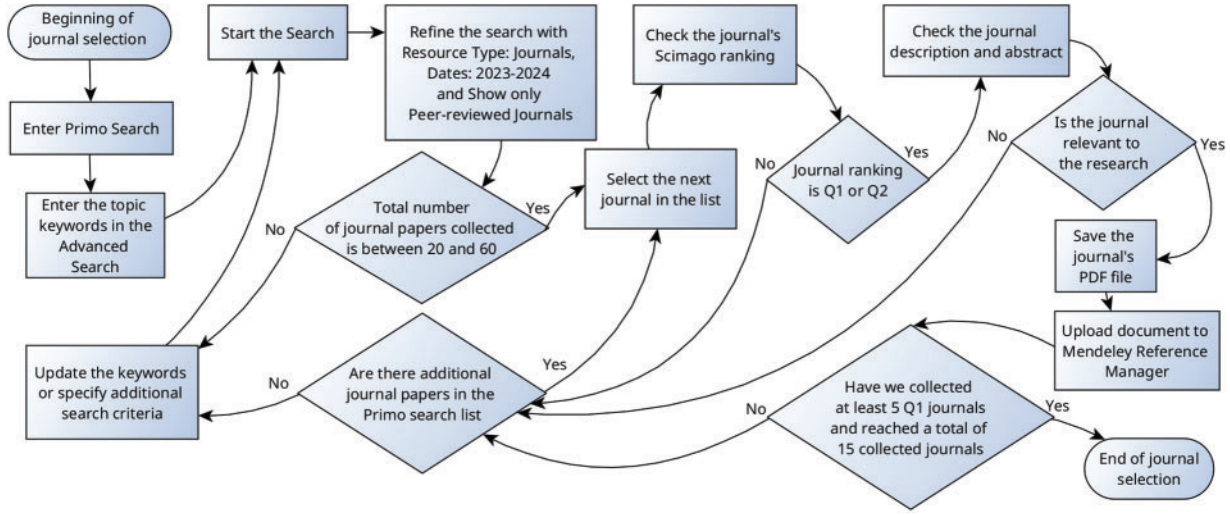


**Figure 1:** The collection process

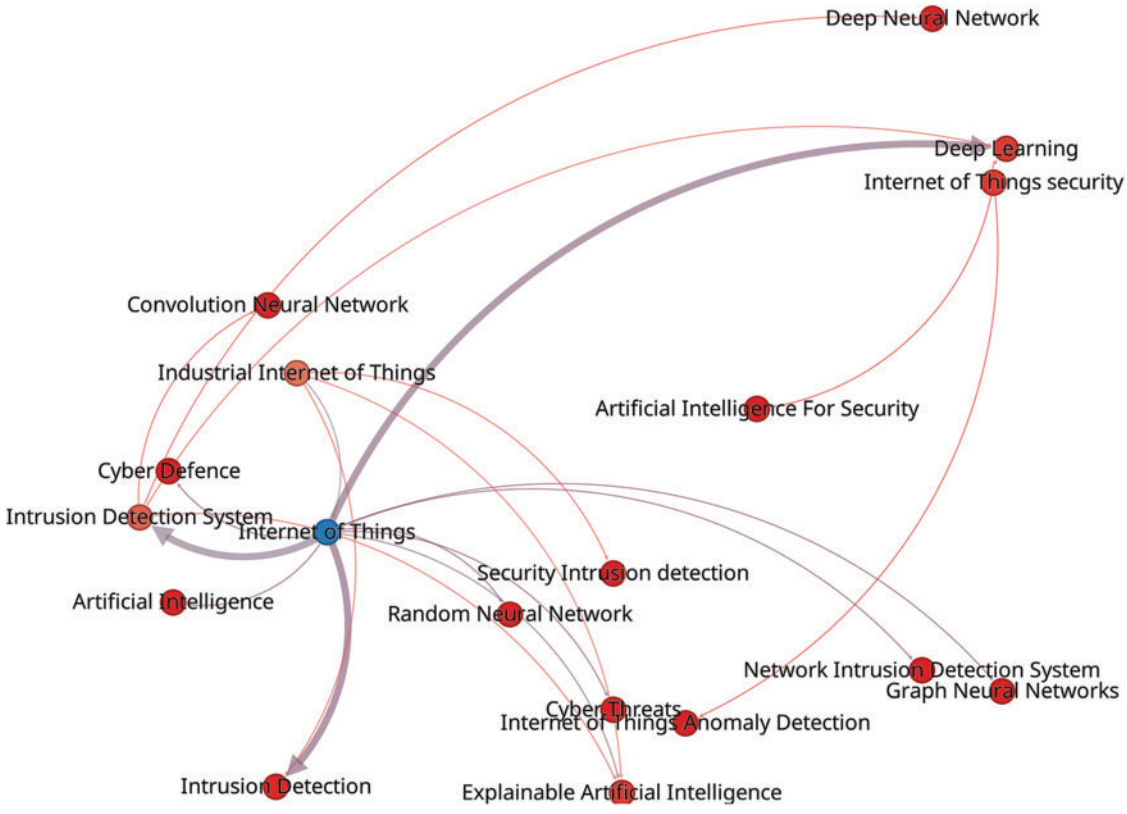


**Figure 2:** The network visualization map for co-occurring keywords

As an evolving technology, AI is comprised of several branches of technological methods, frameworks, and algorithms. Vila et al. [4] mentioned Big Data and the different branches that arise from AI technology.

Fig. 3 illustrates the relationship between AI and some of its primary areas of use, methods, frameworks, and related algorithms.

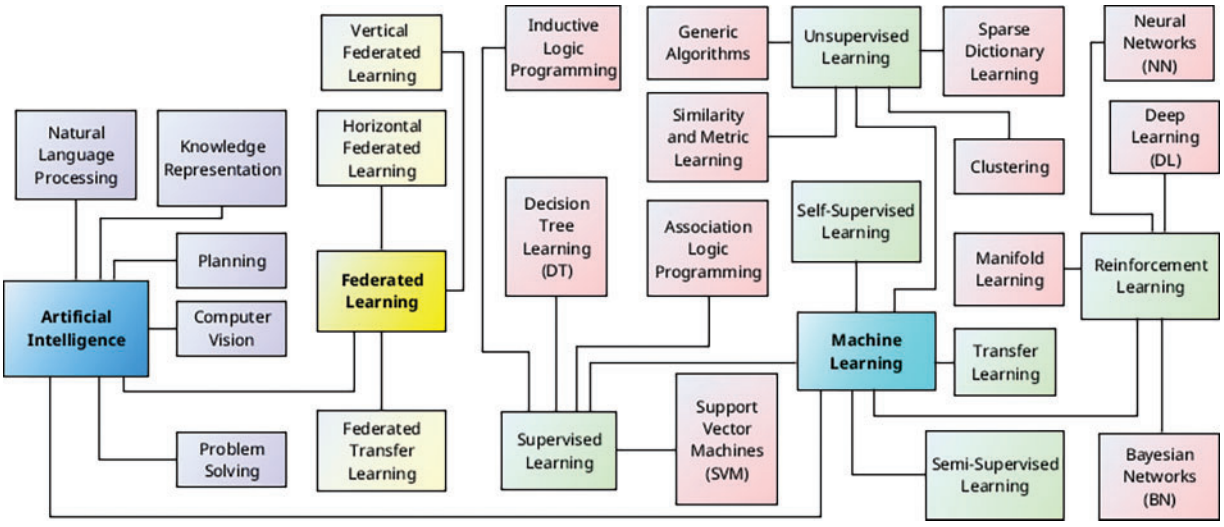


**Figure 3:** The relationships between AI and some of its primary area of use

After analysing the different algorithms, techniques and models mentioned in the selected articles, we can conclude that there are several differences among them, especially in the variety of algorithms used for the IDS and frameworks, as depicted in Fig. 4.

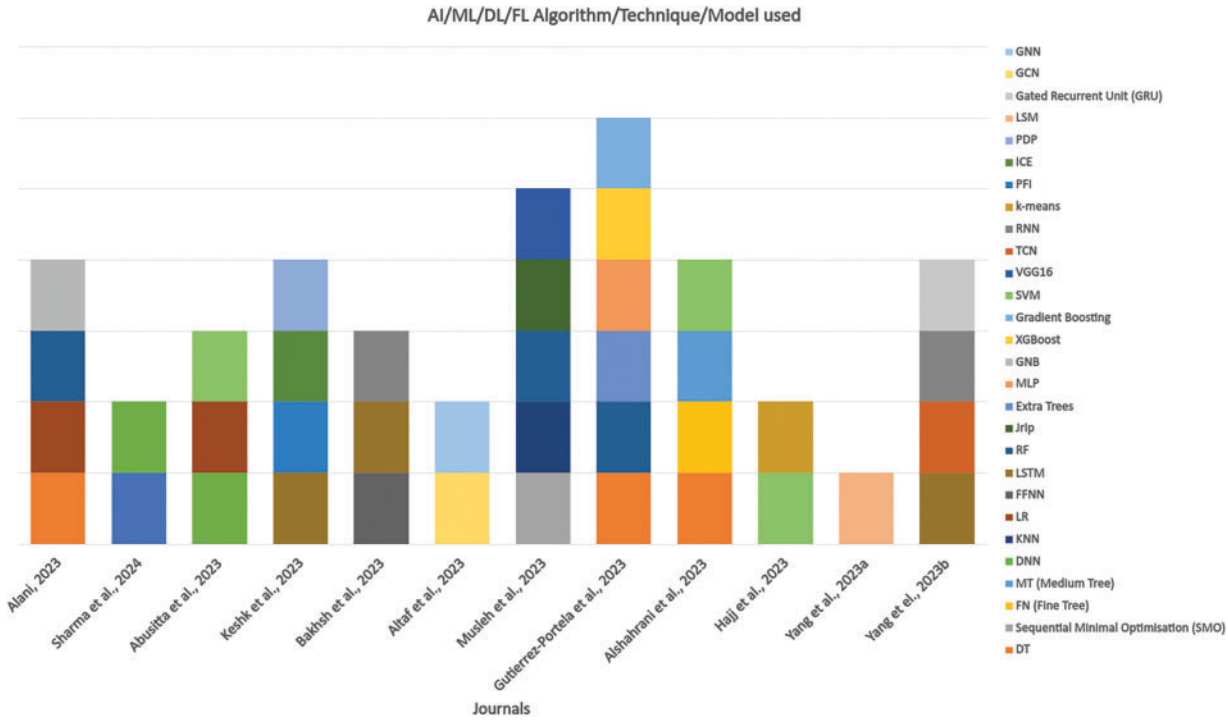


**Figure 4:** Different AI/ML/DL/FL algorithms, techniques and models used in the selected articles

However, there is a single common objective that is clearly shown in these articles, and this is their ability to achieve high prediction accuracy rates as depicted in Fig. 5.

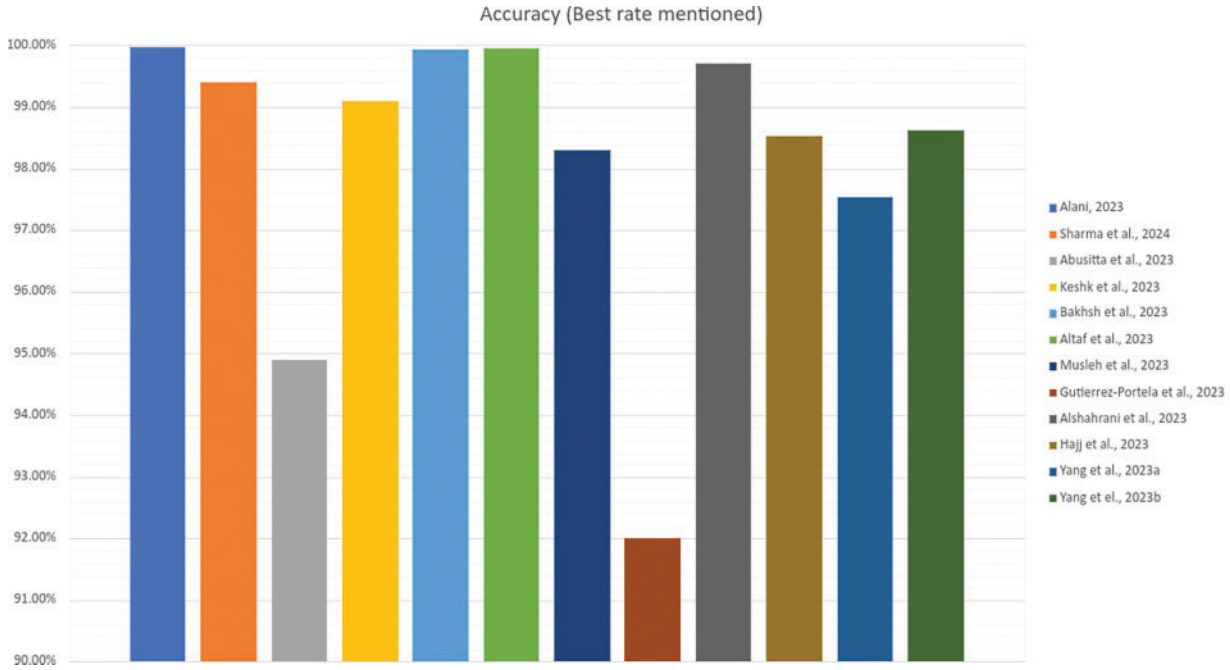


**Figure 5:** The IDS accuracy comparison mentioned for each of the included journals

The highest prediction accuracy was 99.99% [9], ML-based with a Recursive Feature Elimination (RFE) as its feature selection method with strong emphasis on the development of an appropriate preprocessing phase to prepare data for training. This research identifies Random Forest (RF) as the ML model that achieved the highest prediction accuracy. Liu et al. [7] were using a similar approach indicated that the performance of the RF algorithm is most impacted by the accuracy of individual decision trees and the dependencies between them (i.e., diversity).

Deep learning algorithms use multilayer network models to learn the features of data enabling them to identify anomalous traffic more accurately based on large amount of network traffic data, as mentioned by Lu et al. [8]. Research using DL models, such as the one conducted by Sharma et al. [17], indicates the importance of selecting appropriate feature reduction algorithms, to increase prediction accuracy, as well as reducing the model's training time and the computational resources required for the IDS framework. It also indicates the differences between filter-based, wrapper-based, and embedding-based feature selection types and the benefits of reducing feature selection to increase prediction accuracy. The use of feature extraction or reduction gives an overall better detection performance than feature selection; however, it increases the model's training and inference time compared to feature selection, as mentioned by Jing et al. [10]. Saika et al. [11] also commented that foregrounded feature selection provides greater effectiveness of the entire framework.

Another aspect contributing to higher model prediction accuracy is the use of a denoising autoencoder. Its advantages over the traditional autoencoder are mentioned by Sharma et al. [17]. The denoising autoencoder can map the input to an intermediate representation, which increases the IDS effectiveness and accuracy of malicious data detection in heterogeneous environments. Furthermore,

this mapping constructs a better representation of the data before it is being inserted into the classifier, resulting in a higher accuracy.

The same concept of carefully selecting the essential features for extraction to train the AI-based IDS and increase prediction accuracy is further clarified by Abusitta et al. [18]. This research also articulates the direct relation between the chosen IDS framework and the model's ability to produce high accuracy predictions. However, there is a limitation in identifying the vulnerability of an attack class exploit.

A new framework using FFNN [19], LTSM [20] and RNN [21] models was introduced by Keshk et al. [22] with the aim to provide a broader IDS and classification for the entire IoT network, as opposed to other ML and DL algorithms mentioned earlier. This approach was compared with multiple state-of-the-art IDS, showing clear improvements in intrusion detection accuracy. The researchers listed as future work the development of a more robust algorithm and metrics. A notable limitation pointed out by the researchers relates to the lack of an effective testing platform which limits test reliability, inevitably putting this research results in doubt.

Another new framework included the introduction of a GNN-based [23] network IDS that maximises the ability to involve structural characteristics of normal and malicious network traffic, mentioned by Bakhsh et al. [24]. This model proposed a multi-edged graph structure that captures the entire communication between any pair of IoT nodes at the edge level of the network. This novel GNN model combines the benefits of spectral and spatial GNNs tailored to process and learn from complex multigraph-structured information. In addition, this approach is especially helpful in learning graph geometry and traffic patterns in complex networks, resulting in an effective framework for detecting network intrusions. This research proves that the proposed model is more effective at minimising False Alarm Rate (FAR) [25] compared to other models. Moreover, the results of this research demonstrated that the proposed model is capable of learning entirely from the inherent characteristics of network traffic, which may include attacks. This model achieved a high accuracy rate with competitive training times across all tested datasets. Furthermore, almost all evaluations demonstrated an almost perfect accuracy of 99.9%. Mirlashari et al. [26], also using GNN, extended the widely recognized GraphSAGE [27] algorithm to incorporate edge classification and edge embedding capabilities, achieving DDoS attack detection with a precision of 99.8%. Probably due to their high outcomes, the use of GNN-based models has gained popularity in recent years. Furthermore, they have demonstrated great ability to handle complex and structured data, which is difficult to model using traditional classification methods, as mentioned by Gillioz et al. [28]. Research by Hamdi et al. [29] introduced a GNN-based intrusion detection framework that enhances intrusion detection capabilities by generalisation of malicious behaviour patterns from the learning process. A slightly different approach was taken by Wei et al. [30] where a GNN-based traffic anomaly detection extracts traffic features from different channels as time series and then uses a GNN combined with structured learning to learn relationships between features.

A comprehensive intrusion detection accuracy comparison of seventeen different ML models was carried out by Altaf et al. [31]. The result of this research identified that the VGG-16 [32] CNN [33] architecture along with stacking of the KNN [34] ML algorithm achieved prediction accuracy of 98.3%. It showed that incorporating a stacked model along with advanced feature extraction enhances the performance of the trained model in achieving higher prediction accuracy rates. However, without a cleaned and balanced dataset, the entire model prediction capability will be negatively impacted.

Recognising the importance of training datasets when comparing prediction accuracy in ML, a new balanced dataset called IDSAI was introduced by Musleh et al. [35]. It provides a comparison of a

range of datasets with the IDSAI. Furthermore, this study identified the ML algorithm that achieved the highest intrusion detection accuracy among those compared. However, as with any dataset, new intrusion data should be added to increase the efficiency of the training model, which would also increase prediction accuracy.

Software Defined Network (SDN) [36] frameworks using ML techniques for intrusion detection in industrial IoT environments were introduced by Fernando et al. [37]. The ability to use SDN at the network edges, such as network switches, can offload some of the high resource-utilisation requirements of incorporating ML algorithms and training models operations from the IoT devices themselves onto network switching devices. The authors used SVM [38] and DT [39] classification models for evaluating their framework and provided a comparison between several models within each of the classifications to show the differences in accuracy, prediction speed, and training time. This framework produced an extremely high prediction accuracy of 99.7% using the chosen datasets. Results also showed the close accuracy results obtained for each of the models, which allows conclusion as to which classifier model better utilises its training time to produce a higher prediction accuracy. The authors plan to employ newer data sources to enhance the adaptability and efficacy of this study.

A completely different approach was taken by Alshahrani et al. [40] who introduced a cross-layer federated sampling and lightweight IDS for IoT networks using K-means for sampling network traffic and finding anomalies in a semi-supervised way. This system is designed to preserve data privacy by performing local clustering on each device and reducing the traffic by sharing only summary statistics with a central aggregator, which acts as a coordinator. Furthermore, this system is particularly suitable for resource constrained IoT devices, highlighting the advantages of merging operations on the performance of both coordinator and workers in the proposed model. However, some observed limitations were mentioned in this research, such as performance degradation over time, a decline in precision, and an increase in the false-positive rate when the coordinator and workers engage in FL (i.e., Collaborative Learning [41]) in a merging operation.

Both Hajj et al. [42] and Yang et al. [43] introduced new FL techniques which reduce IoT resource utilisation by offloading the global training model's processing operations of the aggregator server to a cloud-based infrastructure. FL also aims to reduce the traffic between IoT devices and the cloud-based components and protect sensitive data collected on each individual edge device from being exposed on the network, as mentioned by Lakhan et al. [44].

Hajj et al. [42] introduced a non-invasive and lightweight detecting mechanisms to enhance IoT intrusion detection in IoT networks while protecting sensitive data. This new method mitigates poisoning and label flipping attacks. The proposed model introduced a scoring mechanism for evaluating participants, based on loss results of the local model and the training dataset size. Furthermore, as this method is based on results obtained from training of local models for detection, it does not require significant additional processing analysis for the same model. Moreover, this research also emphasises the security vulnerabilities of the FL framework, which cannot ensure the robustness of the global models trained collaboratively, as each participant has access to the model's parameters and training data.

Yang et al. [43] introduced an efficient two-stage intrusion detection method considering the time-series properties and resource limitations of IoT devices, based on cloud-edge CL. This reduces computational workload to speed up training. Furthermore, this research found that the model training on edge devices with constrained resources, using TCN [45] models and dimensional reduction techniques, such as SSAE [46], is extremely efficient, compared to LSTM [20] and GRU [47]. In addition, this approach can share threat intelligence with other devices and hence has potential to

defend against unknown attacks collaboratively. The authors supplied convincing evidence of the benefit of their model in reducing memory utilisation and training time by more than 50%, even though the detection accuracy is close to centralised training models. In addition, IoT devices are widely connected to the cloud, which can be easily integrated with cloud-based services to offload resource utilisation for the intrusion detection process, as also mentioned by Figen et al. [48]. However, this research is extremely reliant on labelled data, which is difficult to provide in real-world situations. Alzubaidi et al. [6], Alani [9], and Sharma et al. [17], referred to XAI [49], a relatively simple method to achieve a high IoT IDS prediction accuracy.

As the current research is aimed to identify the most efficient AI models for enhancing IoT IDS, data privacy is also of high importance (see Yang et al. [50]). To illustrate data privacy differences of the main AI models, there is a need to look at the high-level differences between Centralised Learning (CL), Distributed Learning (DL), and Federated Learning (FL).

In Centralised Learning, each IoT device sends its input data along with its private data, allowing the main server to aggregate the information and train the global model that is stored in the cloud. Once new data is received, the IoT device sends a request to the cloud server and in return it receives a response, based on the global model decision. Hence, there is an ongoing traffic between all IoT devices and the cloud server, including sensitive data that identifies each IoT device and its characteristics, as articulated in Fig. 6a.

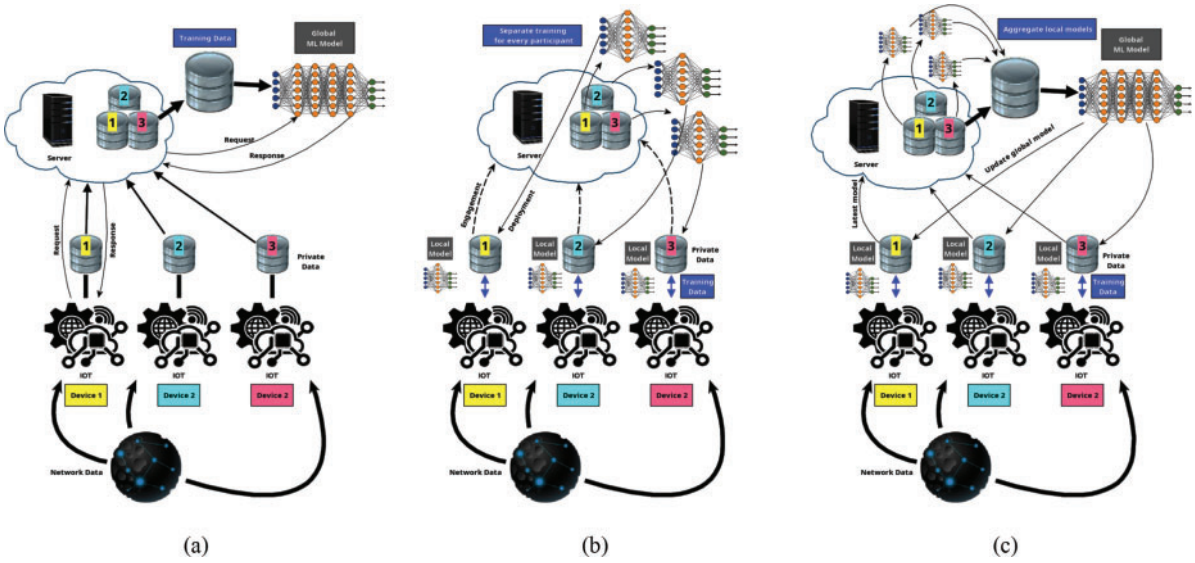


**Figure 6:** (a) Centralised learning. (b) Distributed learning. (c) Federated learning

The DL model is trained to use the same method as the CL; however there is a separate model for each IoT participant, as mentioned by Abdulrahman et al. [51]. During the training phase in a DL algorithm, the IoT participants independently train their model and send the weighted updates to the central server. During the same time, the central server receives updates from IoT participants and performs an output calculation, as illustrated in Fig. 6b.

FL also trains the models independently. The difference between DL and FL is that in FL each IoT participant initialises the training process independently. In addition, the training is conducted collaboratively and independently on individual IoT participants. FL allows decentralised training of

ML and DL models at or near the source of data generation, so that sensitive data does not need to be shared for centralised training, as mentioned by Figen et al. [48]. The model updates received by the cloud server does, thus, not include sensitive data, but only averages and aggregates for the next global model, which is then shared among all IoT participants, as illustrated in Fig. 6c. Therefore, there is an advantage for FL in its ability to provide better protection against the potential exposure of sensitive IoT related data. This also benefits the entire model by reducing excessive network traffic, which directly reduces the risk of eavesdropping and other potential malicious attacks. In addition, the FL model enables multiple clients to jointly train an ML model while keeping their local data decentralized, as mentioned by Zhang et al. [52].

There are distinctive differences between each of the journal articles, indicated by the type of technology used in the research, training phase data sets, classification algorithms, feature selection methods, and additional concepts. The articles chosen for this research include a balanced mix of AI technologies to show the various aspects of research that was recently conducted in a particular field. Researchers try to be creative in using different methods, and some include a significant amount of preparation phases to clean, balance and extract the best features from selected datasets, in the attempt to outfit the dataset with the appropriate training algorithm, aiming to reduce training time while gaining high accuracy prediction results.

Most of the analysed research mentioned the distinguishing relation of appropriate feature selection and the efficiency of prediction accuracy, as it affects the ability of the model to be trained upon relevant network traffic, which should be as close in nature to real-time malicious attack scenarios. In addition, it is interesting to see that classification algorithms vary from using the basic binary, multi-class up to the more advanced classification algorithms of using stacked models. The vast use of different feature selections is also distinctive; however the commonality related to feature selection reflects the aim to extract the most distinctive and useful features, which will contribute to reducing training time while enhancing prediction accuracy.

The AI technologies used and the main intrusion detection characteristics mentioned in each of the included journal articles are shown in Table 1.

**Table 1:** A summary of technology and main intrusion detection characteristics

| Reference | Type of technology | Datasets used | Classification | Additional methods used | Feature selection |
|---|---|---|---|---|---|
| Alani [9] | ML | WUSTL-IIOT-2021 | RF | Introducing a new system-E2I3DS | Recursive Feature Elimination (RFE) |
| Sharma et al. [17] | DL | NSL-KDD, UNSW-NB15 | DNN, CNN | | Filter based, wrapper based, embedded based |
| Abusitta et al. [18] | DL | DS2OS, BoTNeTIoT-L01 | DNN | Extract robust features and neural features together | Using an efficient model to extract robust features and isolate unnecessary features |
| Keshk et al. [22] | ML | NSL-KDD, UNSW-NB15, TON_IoT | Binary | SPIP | Using label encoder, SHAP and PFI to extract notable features |

(Continued)

**Table 1 (continued)**

| Reference | Type of technology | Datasets used | Classification | Additional methods used | Feature selection |
|---|---|---|---|---|---|
| Bakhsh et al. [24] | DL | CIC-IoT-2022 | Binary and multiclass | Feature scaling, data augmentation and balancing using SMOTE | Using Principal Component Analysis (PCA) |
| Altaf et al. [31] | DL | Ton-IoT, Bot-IoT, NF-Ton-IoT, NF-Bot-IoT | GCN model used to perform node classification, improving accuracy up to nearly 10% | | Using multi-edge features |
| Musleh et al. [35] | ML | IEEE DataPort | Comparing several models with and without stacked models and with and without meta classifiers (KNN and SMO) | Comparing stacked models: KNN, SMO, RF, (KNN+SMO) stacked with KNN, (Jrip+RF) stacked with SMO | Using feature extraction to reduce data dimensions while obtaining the relevant information |
| Fernando et al. [37] | ML | A new balanced dataset for training (IDSAI), Bot-IoT | | | Using supervised ML algorithms for binary and multiclass classification |
| Alshahrani et al. [40] | ML | NSL-KDD | DT, SVM, FT, MT | Using SVM and FT model on an SDN controller for early intrusion detection | Using Correlation-based feature selection (CFS) |
| Hajj et al. [42] | FL | NSL-KDD | | Using a cluster of coordinators and multiple workers as the IDS agent. The coordinator learns the baseline, analysing new data, aggregates and distributes it to the workers | Cluster-based feature reduction |
| Yang et al. [43] | FL | CIC-IDS-2017 | Using binary classification (k = 0\|1) | Evaluation of training intrusion detection models based on federated averaging (FedAvg) algorithm | |

(Continued)

**Table 1 (continued)**

| Reference | Type of technology | Datasets used | Classification | Additional methods used | Feature selection |
|---|---|---|---|---|---|
| Yang et al. [50] | FL | CIC-IDS-2017 | | Using Stacked Sparse Autoencoders (SSAE) to produce a more accurate output | Using SSAE to extract useful feature representations from high-dimensional network data. This method requires minimum processing and storage resources |

The data specification and preprocessing characteristics are articulated in Table 2. The data preprocessing, based on the selected datasets, is a critical phase before implementing feature extraction and feeding the normalised data into the training and testing phases of the model. As seen in Table 2, there is a need to conduct a preprocessing phase and prepare the dataset contents for most of the frameworks examined. Some of the dataset preparation processes may involve data cleanup, normalisation, rebalancing, transformation and additional data processing steps, to correlate the selected features more efficiently. As Dehghani et al. [53] mentioned, there is a specific mention of using a network analyser to drop redundant packets to reduce complexity of the algorithm, resulting in reducing the training time. Network components, data split and additional data processing stages may be needed as part of the selected framework requirements.

**Table 2:** A summary of data, preprocessing and network components discussed

| Reference | Network components | User/local components | Data pre-processing | Data split | Additional data processing |
|---|---|---|---|---|---|
| Alani [9] | – | – | – | – | – |
| Sharma et al. [17] | Big data network | | Cleaning, normalisation, encoding | | |
| Abusitta et al. [18] | | Adding noise to the data | Combine robust features and neural features | | Using a pre-training process |
| Keshk et al. [22] | | | To improve interpretation along with a decision engine module | | Introducing a new model-SPIP to enhance performance |

(Continued)

**Table 2 (continued)**

| Reference | Network components | User/local components | Data pre-processing | Data split | Additional data processing |
|---|---|---|---|---|---|
| Bakhsh et al. [24] | ✓ | | Cleaning, normalisation, Transformation | | Using network analyzer to gather network traffic. Data cleaned by dropping redundant packets and removing missing and infinite values |
| Altaf et al. [31] | Network data and edge components (IoT devices) | | Data transformation, identification of source-destination nodes and relative edges | | Graph construction used to create nodes and feature matrix and edges and feature matrix |
| Musleh et al. [35] | | | Dataset preprocessing and balancing using the SMOTE technique | Auto-color Correlogram filter, Auto-color and FcTH, DenseNet, VGG-16 | Using an image dataset created after data pre-processing as the input to the training model |
| Fernando et al. [37] | | | | | Used to contrast model generalisation from different datasets |
| Alshahrani et al. [40] | SDN controller, SDN switches | | Using Correlation-based feature selection (CFS) | | |
| Hajj et al. [42] | Using cloud-edge collaboration architecture | Using local anomaly detection on each worker | | | |

(Continued)

**Table 2 (continued)**

| Reference | Network components | User/local components | Data pre-processing | Data split | Additional data processing |
|---|---|---|---|---|---|
| Yang et al. [43] | IoT network, Edge servers, Central servers | | | | Using the Manhattan similarity between participants according to the scores |
| Yang et al. [50] | Using cloud-edge collaboration architecture | Requires a cloud central server to aggregate all edge servers for model training | | | |

Note: ✓ indicates the component name was mentioned but no specific information was disclosed.

Fernando et al. [37] mentioned the use of SDN on network switches, while Hajj et al. [42] and Yang et al. [50] referred to cloud-edge collaboration for FL frameworks.

The output parameters and metrics used, including the highest accuracy rate, precision, testing, and training times observed in each study are mentioned in Table 3.

**Table 3:** A summary of IDS results, output parameters/metrics discussed

| Reference | Accuracy (%) (for best rate) | Loss (%) (for best rate) | Precision | Recall (%) | F1 (%) | Performance | Cross-validation | False Alarm Rate (FAR) | ROC-AUC | Testing time | Training time (Sec) | Detection time |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Alani [9] | 99.99 | 0.05 | 99.99 | 99.99 | 99.99 | | | | ✓ | 4.7 µSec | 21 | 0.1517 µSec |
| Sharma et al. [17] | 99.40 | 10 | ✓ | ✓ | ✓ | | | | | | 0.455 | |
| Abusitta et al. [18] | 94.90 | | | | | | | | | | | |
| Keshk et al. [22] | 99.10 | | 99.90 | ✓ | ✓ | | | | | ✓ | ✓ | ✓ |
| Bakhsh et al. [24] | 99.93 | | ✓ | ✓ | ✓ | | ✓ | | | | | |
| Altaf et al. [31] | 99.96 | | 99.96 | 99.96 | 99.95 | | | ✓ | ✓ | | 2.89 | |
| Musleh et al. [35] | 98.30 | | 96.30 | 100 | 98 | | ✓ | | | | | |
| Fernando et al. [37] | 92.00 | | ✓ | ✓ | ✓ | | ✓ | | ✓ | ✓ | ✓ | ✓ |
| Alshahrani et al. [40] | 99.70 | | ✓ | ✓ | ✓ | | | | ✓ | | 11.029 | 1100 Obs/Sec |
| Hajj et al. [42] | 98.52 | | ✓ | ✓ | ✓ | | | | | | | |
| Yang et al. [43] | 97.53 | | | | | | | | | | | |
| Yang et al. [50] | 98.62 | | 98.90 | 98.45 | 98.71 | ✓ | | | | ✓ | ✓ | |

Note: ✓ indicates the parameter name was mentioned but no specific value was disclosed.

The E2I3DS system using RFE feature selection and an RF classifier, introduced by Alani [9] achieved an almost perfect prediction accuracy rate of 99.9% and the same precision, recall and F1-score rates, with an exceptionally low loss of 0.05%. Combining this with a training phase of 21 s, and a detection time of 0.1517 µsec, and we get an impressive overall IDS performance.

However, a slightly similar accuracy rate was achieved by Altaf et al. [31] using the GNN-based network IDS with a much faster training time of 2.89 s. This framework considers the full communication between every pair of IoT nodes at the edge level in the network. The result highlights the proposed model's capability to learn entirely from the inherent characteristics of network traffic made up of different attacks. This model introduces an alternative to Alani [9], which achieved the highest yielding accuracy IDS solution in all major output parameters and a high accuracy rate, with competitive training times across all datasets compared.

Additionally, Altaf et al. [31] also mentioned achieving an extremely high measurements related to precision, recall and F1-score. However, the downside of this model is the need to construct a multi-edge graph structure at the beginning of the preparation phase of the model.

Additionally, applying FL poses some challenges, like poor robustness against malicious attacks, unfair resource allocation among devices, imbalanced data distribution, varying data sources and quantities among different devices. These challenges cause significant performance discrepancies in the global model on different clients, as mentioned by Liu et al. [54].

Some reasonably high achieving IDS research was deliberately excluded from this evaluation as it did not fit our key criteria. This includes the work of Yesi et al. [55] based on Deep Covolutional Autoencoder (DCAE) [56], the Facebook Prophet model [57], an Intrusion Detection Deep Learning Multi-class classification model (EIDM) [58], research into ML-based IDS [59], IDS for IT devices operating only in cloud environments [60] reduction of processing power [61], and cost reduction [62].

## 4 Discussion

The aim of the above analysis has been to draw comparisons between recent IDS for IoT devices and identify best practice. General outcomes demonstrate that using AI technologies, IDS systems can provide efficient anomaly detection and increase IoT resiliency towards current and new cyber-attacks. However, it is paramount to protect sensitive data collected on IoT devices and prevent it from being exposed over the network. Moreover, any selected IDS model should improve the performance compared to other tested models, so that it will be able to identify network anomalies, as soon as they arrive from the network. In addition, as IoT devices have limited resources, the IDS model should be as lightweight as possible in its resource utilisation requirements. The same conclusions were drawn by Inam et al. [63], who indicated that due to the peculiarities of IoT devices, their security design is more difficult. Taking into consideration that there is a wide range of devices and protocols available, finding effective security solutions is highly challenging, particularly since conventional anomaly detection techniques generally depend on centralized systems, where data is gathered and analysed in a single location, such as a cloud server or local server may cause delays. These are too numerous to include in this research but they do increase communication expenses [64] and cause privacy hazards such as the Deep Convolutional Autoencoder (DCAE) [56], achieving a detection rate of 99.17%, as mentioned by Yesi et al. [55]; the Facebook Prophet model used with several classifications and algorithms and achieving the highest prediction average accuracy of 96.35% [57], and an enhanced anomaly-based Intrusion Detection Deep Learning Multi-class classification model (EIDM) that can classify 15 different traffic behaviours, including 14 attack types, and achieved an accuracy of 95% [58].

In summary, selecting the most suitable AI-based framework for enhancing IDS in IoT devices and networks is predominantly focused on achieving the highest prediction accuracy, however, there are other parameters to be considered, which have considerable influence on the selected framework. This paper discusses the following IDS characteristics:

- *Prediction accuracy*
- *Training efficiency*
- *Dataset preprocessing*
- *Data privacy*
- *Model performance*
- *Resource utilisation*

### *4.1 Prediction Accuracy*

The main aim of any IDS system is to provide the highest prediction accuracy possible, so that malicious traffic will be identified and distinguished from benign traffic. Alani [9] reported an almost perfect prediction accuracy, precision, recall and F1-score of 99.99% and Altaf et al. [31] achieved 99.96% in accuracy, precision and recall and almost the same F1-score. However, the framework used by Alani [9] needed a relatively long training time of 21 s. In contrast, the GNN-based framework introduced by Altaf et al. [31] achieved a slightly lower prediction accuracy, precision, recall and F1-score compared to Alani [9], but once the multi-edged graph structure was constructed, this model was highly effective in reducing false alarms, hence increasing precision. In addition, it is capable of learning from inherent characteristics of network traffic that constitutes different attacks. This results in high accuracy with competitive training times across several datasets which is why it is a preferred framework for prediction accuracy.

### *4.2 Training Efficiency*

The training phase in AI-based IDS frameworks is fundamental for the efficiency of the entire model. While incorporating an ML algorithm in IoT infrastructure, we must ensure that the model utilises the data efficiently and in as close to real-time as possible, as mentioned by Iqbal et al. [65]. However, additional time required to provide a more accurate prediction reduces the model's efficiency. Comparing the training times mentioned by Alshahrani et al. [40], and Alani [9], the former achieved 99.70% accuracy within 11.029 s, while the latter had slightly higher accuracy but took almost twice the time to achieve an almost perfect prediction accuracy of 99.99%. This suggests that both Alshahrani et al. [40] and Alani [9] require significantly longer training times when compared to Altaf et al. [31], which only requires 2.89 s to achieve prediction accuracy of 99.96%. Therefore, Altaf et al. [31] is the preferred framework for training efficiency.

### *4.3 Dataset Preprocessing*

Dataset preprocessing comparison was mentioned in Table 2, indicating several operations that were performed on the selected datasets, in preparation for feature selection and the training phase of the framework. Therefore, the preprocessing phase is extremely important to any framework. However, this causes the framework to be reliant on a specific dataset, for which the appropriate preprocessing algorithm was already set. When changing datasets, the preprocessing phase should be altered to suit the new dataset. This introduces another aspect that requires more generalisation and the ability to dynamically adjust the preprocessing algorithm for different datasets, while leaving the feature selection with dynamic capabilities.

The limitations of appropriate datasets for different AI-based IDS frameworks were mentioned by Abusitta et al. [18] specifically regarding existing datasets, capable of assisting DL-models in identifying new attacks, specifically Zero-day exploits.

### *4.4 Data Privacy*

Ensuring data privacy and security is one of the most important problems for artificial intelligence modelling of data, as mentioned by Zeng et al. [66]. When analysing IDS solutions, data privacy and data exposure are critical concerns, as malevolent actors might use eavesdropping techniques to expose sensitive information, which needs to be kept private. Such exposure might introduce MItM attacks. Sauter et al. [67] mentioned that a device adopting the IoT concept is generally utilising a connection to a back-end server, sometimes not compatible with industry best practices, which poses a security risk because it bypasses established security concepts, such as the defense-in-depth approach. This may allow malicious IoT devices to take part in the network, enabling them to inflict label-flipping attacks or contaminate the model's training data, which will reduce the model's accuracy prediction and its ability to provide appropriate malicious data prediction [42].

Reducing data exposure between IoT devices and the centralised server, holding the generic ML model is introduced by utilising Federated Learning instead of Centralised Learning or Distributed Learning, as mentioned by Hajj et al. [42], Yang et al. [43] and in Fig. 6, which provides a more secure training method. Similar perspectives were mentioned by Chandiramani et al. [68] and Ahsan et al. [69], who indicated that Federated Learning is highly promising as it is capable of training models on a user-device, without sharing the raw data, thereby preserving privacy while enabling edge devices to collaboratively learn a shared prediction model. However, additional studies by Jagarlamudi et al. [70], Yang et al. [71], and Wang et al. [72] showed that FL also has limitations regarding security and privacy, which require appropriate planning and design to be overcome. Additional research was conducted to establish a method of protecting the confidentiality of data in ML models, like Decentralised Secure Framework (DSF) [73], on a decentralised network without the need for a third-party server, as mentioned by Anh-Tu et al. [74].

### *4.5 Model Performance*

Malicious attack prevention is a core capability of any IDS, especially when these models are AI-based and rely on training data. Comparing several AI-based IDS frameworks and models with respect to prediction accuracy of malicious traffic, the model's resilience and its ability to identify new attacks and anomalies are paramount. Cyber-attacks such as DoS and DDoS are common in generic datasets and hence can be easily identified. However, Zero-day exploits are much harder to locate. Alani [9] mentioned that identifying Zero-day exploits in existing DL-based models will be their aim in future research. Relying on dataset data is another limitation when training a model to identify similar anomalies mentioned in the training data. Furthermore, selecting appropriate features is of high importance as they might also limit the prediction output for new and unfamiliar malicious traffic.

Training time also affects model performance [17], to achieve an accuracy of 99.40% within 455 ms, an extremely high amount of loss, measured as 10% of the input data, is introduced into the system. If slight changes to normal behaviour are identified as abnormal, a significant increase in false positives may be introduced, therefore reducing the efficiency of an IDS model. Song et al. [75] focused on performance related to detecting IoT time series anomalies, using multi-layer Perceptron Graph Convolutional Networks (GCN) [76]. Theoretical work showed a significant improvement over traditional methods. Additional aspects that might impact the performance of a proposed framework are wired and wireless network used to connect the IoT devices to the Internet, as mentioned by Neeti et al. [77] and energy-efficient power and mobility aspects [78] although these are not part of this research.

### *4.6 Resource Utilisation*

An FL-based model, like a distributed model (see Fig. 6) provides resource off-loading by utilising a remote cloud server for training a global ML-model. In addition, IoT devices can train their internal model with the input data they accumulate over time, reducing the need to share sensitive data with the remote server. As mentioned by Liu et al. [54], FL models can train their own models on the edge-device, without sharing the raw data, thereby protecting privacy. In addition, these models can collaborate on learning a shared prediction model, thereby reducing network traffic between each IoT device and the cloud server. However, a sudden rapid increase in resource utilisation of IoT devices will increase its power consumption, which may also be an indication of a potential cyber-attack, as mentioned by Miller et al. [79].

## 5 Future Work

AI-based IDS solutions are based on the ability to train a model and enhance its prediction accuracy of potential malicious cyber-attacks. The training phase consists of an input dataset, appropriate data preprocessing mechanisms and efficient training algorithms which aim to reduce the training time, while keeping model output at high prediction accuracy. The training phase requires appropriate dataset cleanup and balancing operations to allow better feature extraction, resulting in enhanced prediction accuracy while reducing training time. The lack of appropriate datasets, when dealing with IoT devices and networks was highlighted by Divyansh et al. [80] as a key challenge to providing robust and generalised IDS frameworks. Therefore, additional datasets should be created. These should contain the latest and most up-to-date information on network traffic with the latest known malicious and benign data for a range of IoT devices and network components. In addition, as mentioned by Vila et al. [4], new data sources are required to enhance the adaptability and efficacy of IDS study for various IoT devices, and more research is needed to explore new frameworks that can target new vulnerabilities, by studying the features of network traffic generated during malicious attacks. Significant effort has been made in the research of processing IoT data based on neural networks and this trend will most likely continue in the coming decades, as mentioned by Zhang et al. [81].

Additional research involving the adaptation and further implementations of GNN is required to enhance IDS capabilities to identify anomalies in individual devices within an IoT network. The abilities of the GNN to extract information from the inherent behaviour of a device through adjacent neighbours in a network has significant potential to contribute to the identification of Zero day exploits, which are particularly important in infrastructure, utility and military-based IoT networks. Such implementations of GNN capabilities can potentially provide solutions for current and future IDS frameworks.

In addition, identifying new frameworks and enhancing currently existing frameworks is of high importance, even if prediction accuracy of 99.99% is being achieved. There is still a need to reduce training time, so that the IDS ability to predict potential cyber-attacks with high accuracy is reduced as close to Realtime as possible, while keeping resource utilisation to a minimum.

As discussed earlier in this research, the preferred method the ML learning phase for IDS frameworks is FL. There is a need to further research FL methods and the data flow between edge IoT devices and the remote centralised server to identify potential bottle necks in highly congested networks. Special consideration should be given to securing FL frameworks in the event of contamination or malicious activity that has compromised a valid edge device within the internal.

## 6 Conclusions

The E2I3DS framework [9], yielded the highest prediction accuracy rate at 99.99% with an exceptionally low loss rate of 0.05% and a training phase of only 21 s. This also demonstrated that the RF classifier achieved significant prediction accuracy. Altaf et al. [31] achieved similar results with the GNN-based network IDS, with full communication between any pair of IoT nodes at the edge level in the network. This framework has the additional benefit of requiring only 2.89 s of training time, yielding a better overall performance. However, sharing sensitive data over the network still poses security risks and thus requires additional security intervention such as an FL training model which has the potential to reduce sensitive data exposure and excessive network traffic. Thus, combining the GNN-based model with FL has the potential to provide the most efficient overall AI-based IDS for IoT devices and networks.

**Acknowledgement:** We are grateful to Angelika Maag for proof reading and making corrections to this article. Without her support, it would have not been possible to submit this in the current form.

**Funding Statement:** The authors received no specific funding for this study.

**Author Contributions:** The authors confirm contribution to the paper as follows: study conception and design: Shachar Bar; data collection: Shachar Bar; analysis and interpretation of results: Shachar Bar; draft manuscript preparation: Shachar Bar; manuscript final layout and preparation for submission: P. W. C. Prasad, Md Shohel Sayeed. All authors reviewed the results and approved the final version of the manuscript.

**Availability of Data and Materials:** Not applicable.

**Ethics Approval:** Not applicable.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

[1] Statista, "IoT connected devices worldwide 2019–2030, with forecasts from 2022 to 2030," 2024. Accessed: May 11, 2024. [Online]. Available: https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide

[2] P. Agarwal, K. Khanna, and J. J. P. C. Rodrigues, "Role of machine learning and deep learning in internet of things enabled smart cities," in *IoT for Sustainable Smart Cities and Society*, Switzerland: Springer, 2022. doi: 10.1007/978-3-030-89554-9_1.

[3] A. A. Mohd, G. Casalino, and B. Bushan, *Enabling Technologies for Effective Planning and Management in Sustainable Smart Cities*, Switzerland: Springer, 2023.

[4] M. Vila, M. R. Sancho, E. Teniente, and X. Vilajosana, "Critical infrastructure awareness based on IoT context data," *Internet of Things*, vol. 23, 2023, Art. no. 100855. doi: 10.1016/j.iot.2023.100855.

[5] P. Shukla, C. R. Krishna, and N. V. Patil, "Distributed ensemble method using deep learning to detect DDoS attacks in IoT networks," *Arab. J. Sci. Eng.*, vol. 127, p. 925, 2024. doi: 10.1007/s13369-024-09144-w.

[6] L. Alzubaidi *et al.*, "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions," *J. Big Data*, vol. 8, no. 1, p. 307, 2021. doi: 10.1186/s40537-021-00444-8.

[7] J. Liu, X. Li, Q. Wei, S. Liu, Z. Liu and J. Wang, "A two-phase random forest with differential privacy," *Appl. Intell.*, vol. 53, no. 10, pp. 13037–13051, 2023. doi: 10.1007/s10489-022-04119-6.

[8] C. Lu, Y. Cao, and Z. Wang, "Research on intrusion detection based on an enhanced random forest algorithm," *J. Appl. Sci.*, vol. 14, no. 2, 2024, Art. no. 714. doi: 10.3390/app14020714.
[9] M. M. Alani, "An explainable efficient flow-based industrial IoT intrusion detection system," *Comput. Electr. Eng.*, vol. 108, 2023, Art. no. 108732. doi: 10.1016/j.compeleceng.2023.108732.
[10] L. Jing, S. O. Mohd, C. Hewan, and M. Y. Lizawati, "Optimizing IoT intrusion detection system: Feature selection versus feature extraction in machine learning," *J. Big Data*, vol. 11, no. 1, pp. 36–44, 2024. doi: 10.1186/s40537-024-00892-y.
[11] D. M. Saika, S. Ravi, R. Fizza, and S. Ninita, "Detection of botnet in IoT network through machine learning based optimized feature importance via ensemble models," *Int. J. Inform. Technol.*, vol. 16, no. 2, pp. 1203–1211, 2024. doi: 10.1007/s41870-023-01603-1.
[12] B. B. Gupta and A. Dahiya, *Distributed Denial of Service (DDoS) Attacks: Classification, Attacks, Challenges, and Countermeasures*, Boca Raton: CRC Press, 2021. doi: 10.1201/9781003107354.
[13] C. Ning and Y. Zhang, "Label flipping attacks against Naive Bayes on spam filtering systems," *Appl. Intell.*, vol. 51, no. 7, pp. 4503–4514, 2021. doi: 10.1007/s10489-020-02086-4.
[14] L. Constantinos, A. Peratikou, and S. Stavrou, "A novel evil twin MiTM attack through 802.11v protocol exploitation," *Comput. Secur.*, vol. 130, 2023. doi: 10.1016/j.cose.2023.103261.
[15] S. S. Sanjay, A. A. Patil, D. B. Narkhede, S. Singh, and N. Pulgam, "Zero-day attack detection and prevention," in *7th Int. Conf. Comput., Commun., Control Autom. (ICCUBEA)*, 2023. doi: 10.1109/ICCUBEA58933.2023.10392272.
[16] Q. Yang, L. Yang, C. Yong, K. Yan, T. Chen and Y. Han, *Federated Learning*, Cham, Switzerland: Springer, 2020. doi: 10.2200/S00960ED2V01Y201910AIM043.
[17] B. Sharma, L. Sharma, C. Lal, and S. Roy, "Explainable artificial intelligence for intrusion detection in IoT networks: A deep learning based approach," *Expert Syst. Appl.*, vol. 238, 2023, Art. no. 121751. doi: 10.1016/j.eswa.2023.121751.
[18] A. Abusitta, G. H. S. Carvalho, O. A. Wahab, T. Halabi, B. C. M. Fung and S. A. Mamoori, "Deep learning-enabled anomaly detection for IoT systems," *Internet of Things*, vol. 21, 2023, Art. no. 100656. doi: 10.1016/j.iot.2022.100656.
[19] R. Arnaud, F. Carlier, and P. Leroux, "Feed-forward neural network for network intrusion detection," in *IEEE 91st Vehicular Technol. Conf. (VTC2020-Spring)*, Antwerp, Belgium, IEEE, 2020. doi: 10.1109/VTC2020-Spring48590.2020.9129472.
[20] C. Qin, C. L. Chen, C. Zangtai, L. Mei, and J. Long, "Long short-term memory with activation on gradient," *Neural Netw.*, vol. 164, pp. 135–145, 2023. doi: 10.1016/j.neunet.2023.04.026.
[21] R. X. Ma, F. Guo, Z. Li, and L. Zhao, "Knowledge graph random neural networks for recommender systems," *Expert. Syst. Appl.*, vol. 201, 2022, Art. no. 117120. doi: 10.1016/j.eswa.2022.117120.
[22] M. Keshk, N. Koroniotis, N. Pham, N. Moustafa, B. Turnbull and A. I. Zomaya, "An explainable deep learning-enabled intrusion detection framework in IoT networks," *Inform. Sci.*, vol. 639, 2023, Art no. 119000. doi: 10.1016/j.ins.2023.119000.
[23] A. Drif and A. C. Hocine, "Graph neural networks," in *Int. Conf. Complex Netw. Appl. XII*, Cham, Springer, 2024, pp. 61–73. doi: 10.1007/978-3-031-53468-36.
[24] S. A. Bakhsh, M. A. Khan, F. Ahmed, M. S. Alshehri, H. Ali and J. Ahmad, "Enhancing IoT network security through deep learning-powered intrusion detection system," *Internet of Things*, vol. 24, 2023, Art. no. 100936. doi: 10.1016/j.iot.2023.100936.
[25] L. Layman and W. Roden, "A controlled experiment on the impact of intrusion detection false alarm rate on analyst performance," in *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, Sage, 2023, vol. 67, pp. 220–225. doi: 10.1177/21695067231192573.
[26] M. Mirlashari and S. A. M. Ritzi, "Enhancing IoT intrusion detection system with modified E-GraphSAGE: A graph neural network approach," *Int. J. Inform. Technol.*, vol. 16, pp. 2705–2713, 2024. doi: 10.1007/s41870-024-01746-9.

[27] C. Chen, Q. Li, L. Chen, Y. Liang, and H. Huang, "An improved GraphSAGE to detect power system anomaly based on time-neighbor feature," *Energy Rep.*, vol. 9, pp. 930–937, 2023. doi: 10.1016/j.egyr.2022.11.116.

[28] A. Gillioz and K. Riesen, "Graph-based vs. vector-based classification: A fair comparison," in *Graph-Based Rep. Pattern Recognit. 13th IAPR-TC-15 Int. Workshop*, Italy, Springer, Sep. 2023, pp. 25–34.

[29] F. Hamdi, O. Alexis, and S. Mireille, "Efficient network representation for GNN-based intrusion detection," in *Applied Cryptography and Network Security*, 1st Switzerland, Springer, 2023, vol. 13905, pp. 532–554.

[30] C. Wei, G. Xie, and Z. Diao, "Network flow based IoT anomaly detection using graph neural network," in *Knowledge Science, Engineering and Management*, Switzerland, Springer, 2023, pp. 432–445.

[31] T. Altaf, X. Wang, W. Ni, Y. Guo, R. P. Liu and R. Braun, "A new concatenated multigraph neural network for IoT intrusion detection," *Internet of Things*, vol. 22, 2023, Art. no. 100818. doi: 10.1016/j.iot.2023.100818.

[32] R. Reshma and A. J. Anand, "Predictive and comparative analysis of LENET, ALEXNET and VGG-16 network architecture in smart behavior monitoring," in *Seventh Int. Conf. Image Inform. Process. (ICIIP)*, Solan, India, IEEE, 2023, pp. 450–453. doi: 10.1109/ICIIP61524.2023.10537732.

[33] M. Arslan, J. Kong, and M. A. Qureshi, *Accelerators for Convolutional Neural Networks*, Hoboken, New Jersey: John Wiley & Sons, Inc., 2024. doi: 10.1002/9781394171910.

[34] A. Muskan, K. G. Singh, R. Chauhan, A. Kapruwan, and D. Banerjee, "Classification of network security attack using KNN (K-Nearest Neighbour) and comparison of different attacks through different machine learning techniques," in *3rd Int. Conf. Innov. Technol. (INOCON)*, Bangalore, India, IEEE, 2024, pp. 1–7. doi: 10.1109/INOCON60754.2024.10512250.

[35] D. Musleh, M. Alotaibi, F. Alhaidari, A. Rahman, and R. M. Mohammad, "Intrusion detection system using feature extraction with machine learning algorithms in IoT," *J. Sens. Actuator Netw.*, vol. 12, no. 2, 2023, Art. no. 29. doi: 10.3390/jsan12020029.

[36] P. Zhang *et al.*, "Network-wide forwarding anomaly detection and localization in software defined networks," in *IEEE/ACM Trans. Netw.*, IEEE, 2021, vol. 29, pp. 332–345. doi: 10.1109/TNET.2020.3033588.

[37] G. M. X. Fernando, H. B. Arteaga-Arteaga, F. Almenárez, L. Calderón-Benavides, H. G. Acosta-Mesa and R. Tabares-Soto, "Enhancing intrusion detection in IoT communications through ML Model generalization with a new dataset (IDSAI)," *IEEE Access*, vol. 11, pp. 70542–70559, 2023. doi: 10.1109/access.2023.3292267.

[38] R. Du, Y. Li, X. Liang, and J. Tian, "Support vector machine intrusion detection scheme based on cloud-fog collaboration," in *Mobile Networks and Applications*. Netherlands, Springer, 2022, vol. 27, pp. 431–440. doi: 10.1007/s11036-021-01838-x.

[39] D. S. Chen, Q. Song, Y. Zhang, L. Li, Z. Yang and D. Chen, "Identification of network traffic intrusion using decision tree," *J. Sens.*, vol. 2023, p. 1848, 2023. doi: 10.1155/2023/5997304.

[40] H. Alshahrani, A. Khan, M. Rizwan, M. Saleh, A. Sulaiman and L. Vladareanu, "Intrusion detection framework for industrial Internet of Things using software defined network," *Sustainability*, vol. 15, no. 11, 2023, Art. no. 9001. doi: 10.3390/su15119001.

[41] Z. C. Ma, L. Liu, W. Meng, X. Luo, L. Wang and W. Li, "ADCL: Toward an adaptive network intrusion detection system using collaborative learning in IoT networks," *IEEE Internet Things*, vol. 10, pp. 12521–12536, 2023.

[42] S. Hajj *et al.*, "Cross-layer federated learning for lightweight IoT intrusion detection systems," *Sensors*, vol. 23, no. 16, 2023, Art. no. 7038. doi: 10.3390/s23167038.

[43] R. Yang, H. He, Y. Wang, Y. Qu, and W. Zhang, "Dependable federated learning for IoT intrusion detection against poisoning attacks," *Comput. Secur.*, vol. 132, 2023, Art. no. 103381. doi: 10.1016/j.cose.2023.103381.

[44] A. Lakhan, T. M. Grønli, P. Bellavista, S. Memon, M. Alharby and O. Thinnukool, "IoT workload offloading efficient intelligent transport system in federated ACNN integrated cooperated edge-cloud networks," *J. Cloud Comput.*, vol. 13, no. 1, pp. 79–95, 2023. doi: 10.1186/s13677-024-00640-w.

[45] J. F. Chen, S. Yin, S. Cai, C. Zhang, Y. Yin and L. Zhou, "An efficient network intrusion detection model based on temporal convolutional networks," in *IEEE 21st Int. Conf. Softw. Qual., Reliab. Secur. (QRS)*, Hainan, China, IEEE, 2021, pp. 768–775. doi: 10.1109/QRS54544.2021.00086.

[46] J. Yin and X. Yan, "Stacked sparse autoencoders that preserve the local and global feature structures for fault detection," *Trans. Inst. Meas. Control*, vol. 43, no. 16, pp. 3555–3565, 2021. doi: 10.1177/01423312211037621.

[47] G. S. Zhao, C. Ren, J. Wang, Y. Huang, and H. Chen, "IoT intrusion detection model based on gated recurrent unit and residual network," *Peer-to-Peer Netw. Appl.*, vol. 16, no. 4, pp. 1887–1899, 2023. doi: 10.1007/s12083-023-01510-z.

[48] Ö. Figen and S. Alireza, "Cloud-based disaster management architecture using hybrid machine learning approach," *Multimed. Tools Appl.*, vol. 83, pp. 72357–72370, 2024. doi: 10.1007/s11042-024-18333-6.

[49] T. Senevirathna, B. Siniarski, M. Liyanage, and S. Wang, "Deceiving post-hoc explainable AI (XAI) methods in network intrusion detection," in *IEEE 21st Consum. Commun. Netw. Conf. (CCNC)*, IEEE, 2024, pp. 107–112. doi: 10.1109/CCNC51664.2024.10454633.

[50] R. Yang *et al.*, "Efficient intrusion detection toward IoT networks using cloud–edge collaboration," *Comput. Netw.*, vol. 228, 2023, Art. no. 109724. doi: 10.1016/j.comnet.2023.109724.

[51] S. Abdulrahman, H. Tout, H. Ould-Slimane, A. Mourad, C. Talhi and M. Guizani, "A survey on federated learning: The journey from centralized to distributed on-site learning and beyond," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5476–5497, 2021. doi: 10.1109/JIOT.2020.3030072.

[52] Y. Zhang, Y. Y. Li, Y. Wand, S. Wei, Y. Xu and X. Shang, "Federated learning-outcome prediction with multi-layer privacy protection," *Front. Comput. Sci.*, vol. 18, no. 6, pp. 186604–186613, 2024. doi: 10.1007/s11704-023-2791-8.

[53] M. A. Dehghani and Z. Yazdanparast, "From distributed machine to distributed deep learning: A comprehensive survey," *J. Big Data*, vol. 10, no. 1, 2023, Art. no. 158. doi: 10.1186/s40537-023-00829-x.

[54] F. Liu, M. Li, X. Liu, T. Xue, J. Ren and C. Zhang, "A review of federated meta-learning and its application in cyberspace security," *Electronics*, vol. 12, no. 15, 2023, Art. no. 3295. doi: 10.3390/electronics12153295.

[55] N. K. Yesi, N. Siti, S. Deris, and Y. S. Bhakti, "An end-to-end intrusion detection system with IoT dataset using deep learning with unsupervised feature extraction," *Int. J. Inf. Secur.*, vol. 23, pp. 1619–1648, 2024. doi: 10.1007/s10207-023-00807-7.

[56] J. Wong and Q. Zhang, "Deep convolutional autoencoder for energy-efficient smart health wearables in the era of big data," in *IEEE Int. Perform., Comput., Commun. Conf. (IPCCC)*, Austin, TX, USA, 2022, pp. 202–206. doi: 10.1109/IPCCC55026.2022.9894341.

[57] A. S. Mohd, K. C. Rama, and M. Kalra, "Anomaly detection framework for IoT-enabled appliances using machine learning," *Cluster Comput.*, 2024. doi: 10.1007/s10586-024-04461-z.

[58] O. Elnakib, E. Shaabani, M. Mahmoud, and K. Emara, "EIDM: Deep learning model for IoT intrusion detection systems," *J. Supercomput.*, vol. 79, no. 12, pp. 13241–13261, 2023. doi: 10.1007/s11227-023-05197-0.

[59] S. Bajpai, K. Sharma, and B. K. Chaurasia, "Intrusion detection framework in IoT networks," *SN Comput. Sci.*, vol. 4, no. 4, 2023, Art. no. 350. doi: 10.1007/s42979-023-01770-9.

[60] S. Biswas and A. A. M. Sarfaraj, "Securing IoT networks in cloud computing environments: A real-time IDS,"*J. Supercomput.*, vol. 80, pp. 14489–14519, 2024. doi: 10.1007/s11227-024-06021-z.

[61] M. Doaa and I. Osama, "Enhancement of an IoT hybrid intrusion detection system based on fog-to-cloud computing," *J. Cloud Comput.*, vol. 12, 2023, Art. no. 41. doi: 10.1186/s13677-023-00420-y.

[62] A. Kumar and D. Singh, "Detection and prevention of DDoS attacks on edge computing of IoT devices through reinforcement learning," *Int. J. Inform. Technol.*, vol. 16, no. 3, pp. 1365–1376, 2024. doi: 10.1007/s41870-023-01508-z.

[63] U. Inam, N. Asra, N. Shah, A. Farhad, G. Y. Yazeed and A. Nida, "Protecting IoT devices from security attacks using effective decision-making strategy of appropriate features," *J. Supercomput.*, vol. 80, no. 5, pp. 5870–5899, 2024. doi: 10.1007/s11227-023-05685-3.

[64] P. Sharma, S. K. Sharma, and D. Diksha, "Edge-assisted federated learning for anomaly detection in diverse IoT network," *Int. J. Inf. Tecnol.*, 2024. doi: 10.1007/s41870-024-01728-x.
[65] S. Iqbal and S. Qureshi, *Securing IoT Using Supervised Machine Learning in Artificial Intelligence of Things*. 2023, pp. 3–17. doi: 10.1007/978-3-031-48774-3_1.
[66] Q. Zeng *et al.*, "FedProLs: Federated learning for IoT perception data prediction," *Appl. Intell.*, vol. 53, no. 3, pp. 3563–3575, 2023. doi: 10.1007/s10489-022-03578-1.
[67] T. Sauter and A. Treytl, "IoT-enabled sensors in automation systems and their security challenges in sensor networks," *Sens. Netw.*, vol. 7, no. 12, pp. 1–4, 2023. doi: 10.1109/LSENS.2023.3332404.
[68] K. Chandiramani, D. Garg, and N. Maheswari, "Performance analysis of distributed and federated learning models on private data," *Procedia Comput. Sci.*, vol. 165, pp. 349–355, 2019. doi: 10.1016/j.procs.2020.01.039.
[69] N. Ahsan, J. S. He, Z. Nafei, A. M. Shahid, and S. M. Pathan, "Enhancing IoT security: A collaborative framework integrating federated learning, dense neural networks, and blockchain," *Cluster Comput.*, vol. 27, pp. 8367–8392, 2024. doi: 10.1007/s10586-024-04436-0.
[70] G. K. Jagarlamudi, A. Yazdinejad, R. M. Parizi, and S. Pouriyeh, "Exploring privacy measurement in federated learning," *J. Supercomput.*, vol. 80, pp. 10511–10551, 2023. doi: 10.1007/s11227-023-05846-4.
[71] Q. Yang *et al.*, "Federated learning with privacy-preserving and model IP-right-protection," *Mach. Intell. Res.*, vol. 20, no. 1, pp. 19–37, 2023. doi: 10.1007/s11633-022-1343-2.
[72] H. Wang, Q. Wang, Y. Ding, S. Tang, and Y. Wang, "Privacy-preserving federated learning based on partial low-quality data," *J. Cloud Comput.: Adv., Syst. Appl.*, vol. 13, no. 1, pp. 62–77, 2024. doi: 10.1186/s13677-024-00618-8.
[73] I. E. Carvajal-Roca and J. Wang, "A semi-decentralized security framework for connected and autonomous vehicles," in *IEEE 94th Veh. Technol. Conf. (VTC2021-Fall)*, Norman, OK, USA, IEEE, 2021, pp. 1–6. doi: 10.1109/VTC2021-Fall52928.2021.9625336.
[74] T. Anh-Tu, L. The-Dung, K. Jessada, and H. Van-Nam, "An efficient approach for privacy preserving decentralized deep learning models based on secure multi-party computation," *Neurocomputing*, vol. 422, pp. 245–262, 2021. doi: 10.1016/j.neucom.2020.10.014.
[75] W. Song *et al.*, "A novel semi-supervised IoT time series anomaly detection model using graph structure learning," in *Collaborative Computing: Networking, Applications and Worksharing*. USA, Springer, 2024, pp. 375–391. doi: 10.1007/978-3-031-54528-3_21.
[76] R. Reka, R. Karthick, R. S. Ram, and G. Singh, "Multi head self-attention gated graph convolutional network based multi-attack intrusion detection in MANET," *Comput. Secur.*, vol. 136, 2024, Art. no. 103526. doi: 10.1016/j.cose.2023.103526.
[77] G. Neeti and S. Vidushi, "Context aware hybrid network architecture for IoT with machine learning based intelligent gateway," *SN Comput. Sci.*, vol. 4, no. 3, pp. 297–314, 2023. doi: 10.1007/s42979-023-01736-x.
[78] S. Regilan and L. K. Hema, "Machine learning based low redundancy prediction model for IoT-enabled Wireless Sensor Network," *SN Comput. Sci.*, vol. 4, no. 5, pp. 545–555, 2023. doi: 10.1007/s42979-023-01898-8.
[79] J. Miller, L. Egharevba, Y. Hariprasad, K. K. J. Latesh, and N. K. Chaudhary, "Cyber security attack detection framework for DODAG control message flooding in an IoT network," in *Information Security, Privacy and Digital Forensics*. Springer, 2023, pp. 213–230. doi: 10.1007/978-981-99-5091-1_16.
[80] T. Divyansh, K. S. Jaspal, and S. Srikant, "DeepThink IoT: The strength of deep learning in Internet of Things," *Artif. Intell. Rev.*, vol. 56, no. 12, pp. 14663–14730, 2023. doi: 10.1007/s10462-023-10513-4.
[81] L. Zhang, L. Li, B. Dong, Y. Ma, and Y. Liu, "Understanding the trend of Internet of Things data prediction," in *Multimedia Technology and Enhanced Learning*. Springer, 2024, pp. 308–318. doi: 10.1007/978-3-031-50580-5_27.