DEEP ANOMALY DETECTION USING EFFICIENT FEDERATED LEARNING

Krupa Tharahunise Krishnappa
Paderborn University
krupa08@campus.uni-paderborn.de

25th July 2021

Abstract

IIOT (Industrial Internet of Things) has contributed immeasurably to the technology. Detecting anomalies/abnormalities in IIOT to enhance systems has been an interest of research for long, where deep learning (DL) has been immensely beneficial. In addition to DL, federated learning(FL) has been used in detecting anomalies through recent years. So, the present report aspires to present a representative analysis to address various types of anomaly detection in IIOT. Aiming to show the demand for FL in the future, few FL based models and suitable approaches in ML and DL are analysed. The report shows a comprehensive description of three lately proposed FL frameworks for detecting anomalies in broader aspects. Besides the implementations in FL, associated problems are also deduced, for which will aid the horizon of future research inclinations. Further the report will eventually provide possible solutions with a coherence.

***Keywords*— Federated learning(FL), Deep Anomaly Detection (DAD), Attention Mechanism-based Convolutional Neural Network-Long Short Term Memory (AMCNN-LSTM) model, Gradient compression scheme(GCM), FLAD(Federated Deep reinforcement learning empowered anomaly detection)**

1 Introduction

IIOT (Industrial-internet-of-things) is a state-of-art technology in automation and smart manufacturing. Despite its advancements, there exists a variety of anomalies/abnormalities. According to [1], "anomalies are the points in the data which deviate from normal observations". Whereas according to [2], "anomalies are the users that expose sensitive data of other authenticated users". Also according to [3], "anomalies are the malignant attackers who attempt to theft and manipulate data". Therefore there exists various kinds of anomalies. These anomalies will undoubtedly lead to system failures or may sometimes lead to industrial disasters in IIOT, when left unchecked. Thus, anomaly detection becomes a significant aspect for device performance [1] and security issues [[2], [3]], which becomes primary in avoiding system failures and intrusion detection (ID) respectively. In order to detect these anomalies accurately while ensuring efficiency in communication, developing efficient architectures becomes critical. As a consequence, researchers have applied many innovative methods to detect anomalies for enhancing systems. Out of various methods, Deep learning (DL) based methods detect anomalies by identifying unique features from the data [4], have been ultimately accurate. Despite their accuracy, DL based methods have many drawbacks. They are centralised and hence not flexible, possess various security and privacy concerns, lack adaptability in dealing with dynamic and automatic updation of detected models. Alternatively, Federated learning (FL) based approaches are decentralized and therefore overcome the drawbacks of DL.([1] - page 1 - 3)

Generally FL has four basic steps namely: Local training, model transmission, model aggregation and model deployment. Where in the beginning, the algorithm used in FL .i.e. FedAvg algorithm selects a set of clients based on the availability and computational resources. The main server initially deploys its model to all the connected clients, these clients use their local data to train their models, then the trained and updated model (model weights) is transmitted to the main server. Where the main server uses FedAvg algorithm to aggregate all the local models to obtain a global model, then the obtained global model is deployed back to the clients. This cycle continues

until the model meets an optimised convergence. Depending on the client distribution, FL is classified into three categories: Horizontal FL (HFL), Vertical FL (VFL) and federated transfer learning (TFL). One of the simplest examples one could think of in FL is: timely android updates.([5] , [6])

Even though DL based approaches are centralized with drawbacks, the deep models have served as better foundations in anomaly detection. For example, a recent work presented in [4] provides deeper insights on the research in the field of DL, out of which a brief discussion is provided in the present report. When we compare the complexity of DL in its entire research area, it does not match with the complexity that it has achieved in detecting anomalies [4]. Which means more deep models are needed in anomaly detection. Also, there exists plenty of ML and DL models for anomaly detection that has to be migrated to FL ([4], [7], [5], [6]). In addition, foundations based on FL for anomaly detection qualifies the ML/DL models to bestow exclusive and optimised architectures. Besides architectures, FL based foundations also aids in mitigating communication overhead, provides better performance, better privacy and much more advantages [1]. There has been a large growth in the research on foundations of FL in detecting anomalies.

Recently an FL based framework was unveiled in [3] that is based on ensemble learning. The anomalies where static approaches fail to detect intrusion in the networks of IOT, were addressed in this framework. The framework detects intrusion in the networks of IOT devices in a dynamic way. GRU's and LSTM's (variants of recurrent neural networks) were experimented to ensure high detection accuracy. GRU's performed better in accurate anomaly detection and were computationally less expensive. Ensemble learning was introduced to maintain dynamicity in anomaly detection. An ensembler was used for model aggregation and to provide accurate attack detection. The test bed used was a modbus network data set .i.e. a de-facto standard communication protocol used in industries. The framework was proven to mitigate the false alarm rates and error rates, in-turn providing a better accuracy in predicting anomalies. The framework will be further discussed in the present report.([3] - page 1 - 9)

Prior to [3], another framework was unveiled in [2] that is based on mobile edge computing architecture. Consider a scenario, when an end-user can expose other authenticated user's raw information like name, sex, age, occupation and other private details. Detecting such users behaving abnormally will increase the security in systems. Detection of such abnormal users cannot be done by other users, but can be authorised to a higher level in the network for a better accuracy. Authorising anomaly detection becomes significant. Hence [2] adopted authorisation for anomaly detection. Authorisation was provided locally, regionally and globally. The framework uses a Federated deep reinforcement learning (FDRL) algorithm DDPG (Deep deterministic poly gradient algorithm). DDPG algorithm provides a relation between privacy leakage degree, actions of the users and anomaly detection. Such a relation will make it flexible to detect anomalies accurately. Two mechanisms : a feedback mechanism and an appeal mechanism were introduced to improve detection accuracy and to reduce negative impacts in the network. The framework was proven to achieve high throughput, low latency and high values of accuracy in detecting anomalies.The framework will be further discussed in the present report.([2] - page 1 - 9)

Prior to [2], another important framework was unveiled in [1]. Consider the anomalies in the industrial edge devices. These devices aggregate time series data of the industrial machines. Detecting anomalies that occur in the time series data of edge devices will lead to a better performance in industrial machines. Such anomalies were addressed in the framework presented in [1]. The framework was initially setup based on FL architecture. Then a model was introduced, which aggregates AMCNN (Attention mechanism in Convolutional neural networks) and LSTM (a variant of Recurrent neural networks) for accurately detecting anomalies. Regardless of Modelling, performance metrics (accuracy, errors, efficiency) becomes really important. Hence gradient compression mechanism(GCM) was introduced to increase the efficiency in communication. The test beds used for investigations were MNIST and CIFAR-10 datasets. The framework was proven to possess high accuracy, low prediction errors and mitigates communication overhead to almost 50% when compared with classical ML approaches. The framework will be further discussed in the present report. ([1] - page 1 - 9)

Although FL has acquired an immense growth, the proposed architectures still possess certain limitations. Hence to full fill the gap, the present report analyses different FL- based architectures to provide the research gaps. Main focus of the analysis is on the above three architectures, which are analysed based on advancements and fundamental instincts. But other architectures were outwardly studied to cover wider aspects in ML, DL and FL. Overall, all the analysed models use: higher level architectures of ML, DL and FL, networking aspects, test beds (data sets for detecting anomalies), security models and communication mechanisms.([1], [2], [3], [7], [6], [5])

Since DAD is a major area of concern, reason being that complexities of DL are still way far behind when it comes to anomaly detection [4], further research is required. As a consequence, the present report gives a quick overview on deep anomaly detection. Keeping DAD as a starting point, the present report discusses how things can be made dynamic for the future research using foundations of FL. Forthcoming challenges can be resolved by integrating DAD with high level architectures of ML, DL and FL [4]. The present report seeks the heed of the individuals working and studying in the field of ML, industrial domains and in IOT domain. The main motivation

behind this report is to investigate deep anomaly detection in FL broadly. The aspiration is to provide useful context and implications on how things could be improved. Specifically in terms of the modelling and performance metrics, considering all kinds of interdisciplinary aspects. Additionally, the report provides advantages, disadvantages, advancements and the questions left unanswered in some studies that could be carried out in the near future to address complex forthcoming problems.

Structure of the report: Section 2 is mainly focused on methods used in anomaly detection. It initially explains about Deep anomaly detection (DAD) since DAD requires further research as stated in [4]. Then about federated anomaly detection. Where the present report provides its best efforts to study the models on FL for anomaly detection and categorise them based on the key advantages provided by FL. Then the three significant FL based frameworks in detail for detecting anomalies to get deeper insights. Then a brief analysis of the three models is done. Section 3 gives a brief summary. Finally concluding the report in section 4. Section 5 explains about findings from all the provided resources and is hence named as: Outlook.

2 Methods

This section gives a detailed overview on the works being studied. Initially an overview of DAD from the work presented in [4] is summarised. Further, about federated anomaly detection with a taxonomic classification. Then, the three FL based frameworks: an AMCNN-LSTM model and a GCM mechanism presented in [1], authorization of anomaly detection (FLAD framework) presented in [2] and then about the ensembler based FL framework presented in [3]. Then all the three models are analysed in section 2.2.4, where strengths and weaknesses are analysed.

2.1 Deep anomaly detection

This section summarises the work presented in [4]. DAD is the usage of deep learning in anomaly detection. DAD aims to learn features in a model or the scores of anomalies to perform anomaly detection ([4], [8]). It's approaches are of three types: Supervised, semi-supervised and unsupervised. There has been a large research on developing deep models for anomaly detection, which is an ultimate focus on resolving many challenges. In order to get an extensive insight on the developed deep approaches for anomaly detection, this section discusses an overview of the deep architectures being developed till date (presented in [4]). Initially, the nature of complications in anomaly detection, then about the complicated challenges and then selectively delineate the deep architectures with their competence on resolving these challenges [4]. These deep architectures were categorised taxonomically in the work presented in [4], hence the taxonomy proposed in [4] will be briefly discussed. ([4])

Attributing to the nature of complications in anomaly detection, there exists various inherent complications. First being uncertainty. Anomalies possess large variety of uncertainties for example: unexpected system behavior, unexpected data manipulation, devious attacks and unexpected intrusion in the network etc. Second being irregularity and imbalance in the classes of data sets. Since anomalies occur rarely in a data set, it becomes challenging to aggregate a huge number of labelled anomalous data points. Irregularities also include the anomalies that occur in a single data set with contrasting features, which makes it difficult to classify them. Third being diversity in anomalies. There exist varieties of anomalies namely: point anomalies, conditioned anomalies and grouped anomalies. Even though the nature of complications seem simpler, they are challenging to detect. These challenges have been proactively handled by deep learning.([4])

Challenges being: First being an active recall of anomalies detected (.i.e. shallow recall) and supervision of anomalies. As we saw above that anomalies occur rarely, it becomes difficult for the model to interpret them accurately. Hence, improving the recall rate is still an existing challenge. Also when it comes to supervision of anomalies, they are mostly sparsed instances in a given data set. Reason being that, supervision of anomalies is computationally expensive. For example: the rare malicious attacks that occur in a network. These rare instances have to be efficiently labelled for accurate detection. Hence providing evidence on why a data was labelled as an anomaly becomes a challenge. Second being, detecting anomalies in higher dimensions and non-IID data sets. Anomalies become inconspicuous in higher dimensions. It becomes challenging to detect such anomalies. It also becomes challenging when it comes to anomalies that occur in non-IID data sets because the data is spatially, temporally and graphically dependent. Third being efficient learning of the measure of normality or abnormality. In learning the measure of normality or abnormality, supervised methods become unsuitable and inefficient because they assume that all the data sets are labelled, which is not the case always. In, contrast when it comes to unsupervised methods they do not require labelled data for anomaly detection, hence they become more feasible. Research on such methods like for example in [1] are continuously being carried out. As described above, anomalies occur

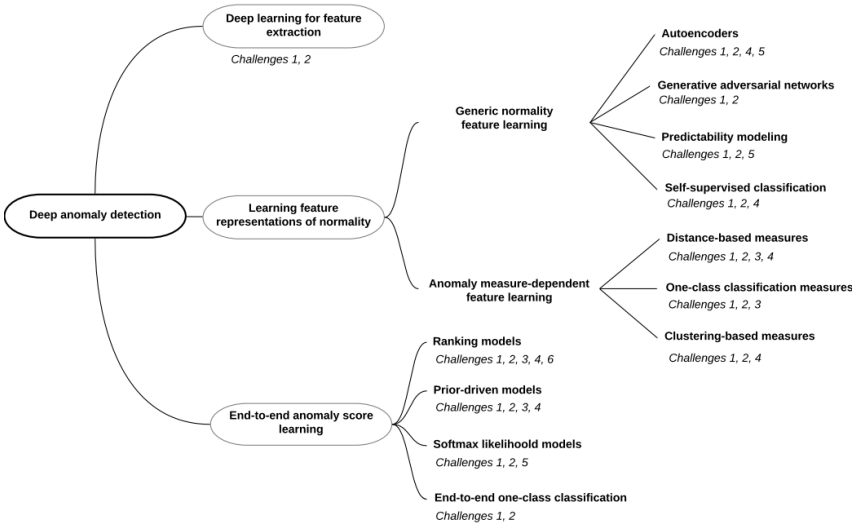


Figure 1: Taxonomy of DAD approaches [4]

rarely and thus there exists a small amount of labelled anomalous data. Hence learning abnormalities from such a data becomes challenging. One research direction could be to utilize semi supervised methods in such scenarios. Subsequently, anomalies that are not covered by labelled anomalous data also becomes challenging. Fourth being, detecting anomalies resilient to noise and detecting complex anomalies. A data space can be sparsed with different variations of noise occurrences. So, it becomes challenging to detect anomalies in such scenarios. When it comes to complex anomalies, various detection methods are bench-marked to detect point anomalies. Whereas, including the aspects of diverse anomalies in the existing anomaly models becomes challenging. All these challenges are explained in detail in [4]. Deep learning solves all the above challenges to some extent. ([4])

In order to address all the above challenges, DL has been utilized to implement a variety of models and algorithms to detect anomalies. As, the work [4] explains all the above challenges in detail, it also presents a taxonomic classification of the existing DL methods in anomaly detection (DAD). These DAD methods aid in solving the corresponding challenges as illustrated pictorially in figure 1 according to the detailed list of challenges in [4]. Where the classification was done into three categories namely : DL for extracting features, DL for learning normality representation and learning anomaly scores end-to-end. Where the first basic category "DL for extracting features", helps in reducing false positives, improves recall rates in anomaly detection and helps in detecting anomalies in higher dimensions. Then the second basic category "DL for learning normality representation", is implicitly based on the degree of normality or abnormality. It is further divided into two categories, learning normality features generally and learning features based on the measure of abnormality. Where the category "learning normality features generally" has various categories based on generic modelling architectures. Then the category "learning features based on the measure of abnormality" includes categories which are based on the degree of abnormality. The third basic category "learning anomaly scores end-to-end", includes classification of models which learn anomaly scores end to end on a higher level. The taxonomy is explained in detail in [4] with the models that fall under these these categories. The full taxonomy is as illustrated in figure 1.([4])

2.2 Anomaly detection with efficient Federated Learning

As discussed in the introduction, foundations of FL have been immensely beneficial in deep anomaly detection. Subsequently, this section discusses about the usage of FL in anomaly detection. As, federated learning aims to enhance data privacy, provide higher flexibility of data transmission, reduce communication overhead and much more to it [1]. Due to these extraordinary features, FL bestows advanced options for deep methods in anomaly detection [1]. There is an exclusive research going on in developing optimized federated models to detect anomalies. So, this section discusses an overview of the detection models based on FL developed till date. Initially we will see how FL can optimise the nature of complications discussed in section 2.1. Then about how the challenges discussed in section 2.1 can be better addressed using federated approach. Then the present report introduces a brief classification of federated models for anomaly detection, implemented till date based on the key advantages provided by FL. Further this section discusses three important lately developed models, providing their advantages.([1], [2], [3], [6], [5])

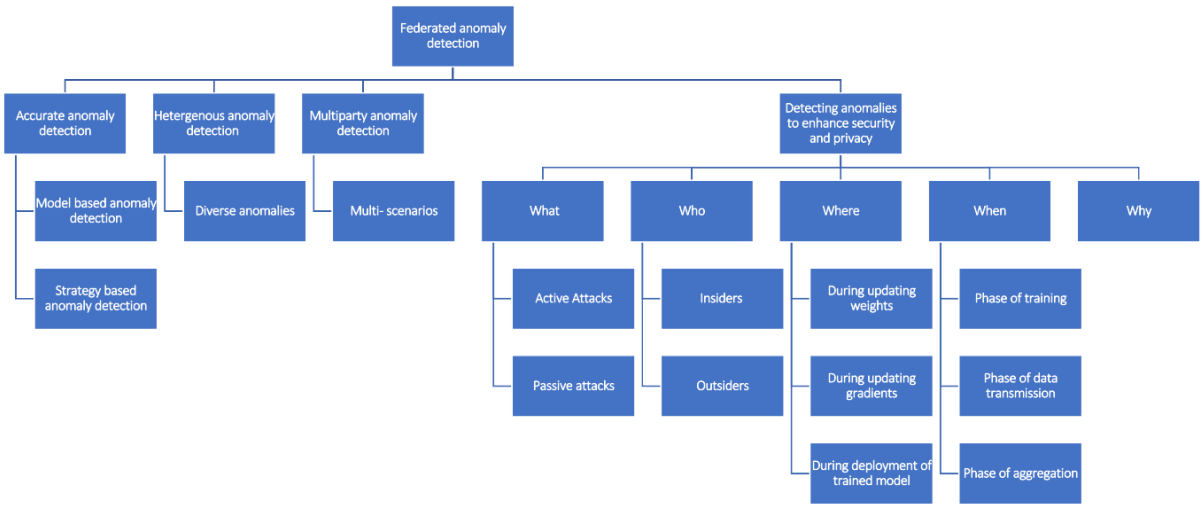


Figure 2: Taxonomy of Federated anomaly detection (A brief classification of federated models for anomaly detection implemented till date) [4], [10], [7], [1], [2], [3], [11], [6] (Fourth main category was analysed based on the taxonomy provided in [10])

Addressing the nature of complications discussed in section 2.1, initially when it comes to uncertainty, the anomalies can be easily detected because of decentralization in FL([1], [2]). Secondly, attributing to irregularity and imbalance, an optimized detection of anomalies in such scenarios can be performed because of distributed on-device training strategy implemented in FL ([1], [2]). Thirdly when it comes to diversity in anomalies, FL strategy can be used to benchmark models to detect multiple anomalies at the same time because of the presence of distributed systems. For example: a framework can be developed to do multiple tasks at the same time, similar to the work done in [9].

Attributing to the challenges discussed in section 2.1, recall rates can be enhanced because of frequent models updates that occur in FL [3]. Whereas, anomaly detection in higher dimensions, non-IID data sets and efficient learning of degree of normality and abnormality can also be enhanced because of distributed learning ([1], [2]). Also, detecting anomalies resilient to noise and complex anomaly detection becomes much more simpler because of local on-device training [3]. Further, due to the decentralization in FL, a higher accuracy of anomaly detection can be expected [1].

As we know that federated learning is one of the ongoing research in the field of anomaly detection, there exists a variety of key advantages that aids in improving the existing research models. These advantages also aims to provide deeper insights to come up with new strategies in future. Hence to have a detailed understanding of the federated anomaly detection models, the present report provides a taxonomic classification of the models proposed till date based on the key advantages of FL. The federated anomaly detection is divided into four main categories and these four main categories are in turn categorised into further sub categories. An overview of the classification is illustrated in figure 2. Basically the federated anomaly detection consists of four categories - Accurate anomaly detection, heterogeneous anomaly detection, multiparty anomaly detection and detecting anomalies to enhance privacy and security. These four categories are based on the advantages provided by FL. ([1], [3], [7], [6], [4], [2])

The first main category consists of accurate anomaly detection, which is due to the decentralized learning in FL. Although accuracy increases because of on-device training, it can further be made more accurate by integrating higher level architectures implemented in ML (subcategory - model based anomaly detection). For example: the work presented in [1] uses attention mechanism to integrate with convolutional neural networks and LSTM. In increasing accuracy, integrating advancements to fedAvg algorithm would be a research inclination for anomaly detection [12]. Another alternative to further increase accuracy is based on the learning strategies used (subcategory - strategy based anomaly detection). For example: the work presented in [2] used FDRL algorithm which helped them to accurately detect anomalies. Also, their framework can be further improved by further integrating deep Q-learning with it, this would be another research inclination for federated anomaly detection [13]. Another example for strategy based anomaly detection is the work presented in [3] where ensemble learning was used to increase accuracy of anomaly detection.

The second main category consists of heterogeneous anomaly detection, which is due to distributed on-site learning. It is because of this reason that enables researchers to come up with optimized models for heterogeneous anomaly detection. For example: when a model is implemented, it can be bench-marked to detect variety of anomalies [9] (as discussed above). Also when it comes to heterogeneity, we should consider noise resilient anomaly detection and uncertainties in anomalies that needs attention [4]. One research direction for such scenarios is to adopt the strategy of biased client selection similar to the work presented in [14] for detecting noise based anomalies. Another research direction maybe to utilize visual analysis technique similar to the work presented in [15] for detecting heterogeneous anomalies.

The third main category consists of multiparty anomaly detection. Here take an example of two banks. Assume some malicious user borrows a debt from the first bank, and uses that money to pay back the debt he had taken from the second bank. Such illegal practices will lead to a great fall in financial sector. Hence an FL-based framework can be developed that encrypts the local model updates and then de-crypts the model when the aggregated model is received from the cloud. Such anomalies can also be detected by developing a visual analysis similar to the work presented in [15]. Another strategy that can be applied for multiparty anomalies is to efficiently learn degree of normalities and abnormalities in a single framework, analyse them and provide feedback to the respective parties. Likewise, there exists various other innovative strategies for multiparty anomaly detection. Similar strategies can be applied to various domains, which will take our world to more advanced level. ([5], [6], [2], [7])

Fourth main category consists of detecting anomalies to enhance privacy and security. This category was analysed based on the taxonomy provided in [10]. Anomalies have to be analysed properly based on different types of questions: what, who, where, when and why. These questions can be easily answered when FL is used. When we look at the first type of question: what?, which means what kind of security attacks occurred. Basically as per [10] there are two types of attacks- subcategories: active attacks and passive attacks. Where in active attacks, the systems are attacked directly by content manipulation. In such attacks, victim is informed immediately. Whereas in passive attacks, the contents of the systems are observed or copied. In such attacks, victim is not informed. When we look at the second type of question: who?, which means who are the attackers. These attackers may include insiders or outsiders. Where insiders include venomous clients or servers who try to pass malicious model updates which may in-turn learn to security threats. For example: the model proposed in [16] lies under the category of insiders. In, contrast outsiders include customers or eavesdroppers who try to perform unknown attacks. For example: the attacks on the model presented in [2] includes insiders as well as outsiders. When we look at the third type of question: where?, which means where did the attack actually happen. Was it during model weights? was it during gradients update? or was it during deployment of trained model?. When we look at the fourth type of question: when?, which means at which point of time during the learning process. Was it in the training phase? was it in the data transmission phase? or was it in the aggregation phase?. When we look at the fifth type of question: why?, which means why did the attack occur?. Was it because of deducing the inputs and labels for training in the FedAvg algorithm? Was it because of various memberships in the FL model?, Was it because of class representatives in the models? or Was it because of deducing the privacy properties?. Overall, the fourth category is clearly addressed by the framework proposed in [2]. Many such frameworks can be developed, with further enhancements. Implementing such frameworks for anomaly detection with differential privacy can be a future research direction as proposed in [6]. A full taxonomy is as illustrated in figure 2, where the present report makes its best efforts to study FL models for anomaly detection and categorise them. ([10], [3], [2], [7])

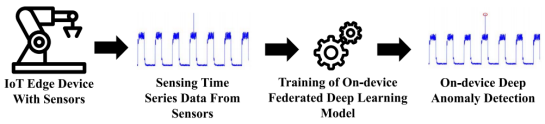
The taxonomy includes fundamental instincts about FL because it was provided by referring to several sources. Although it provides instincts about FL, deep models are always required to meet the expected pro-activeness [4]. As discussed in the taxonomy, all the FL based detection models fall under the illustrated categories. To have a deeper insights into these models, the present report further discusses three significant, on the other hand lately developed FL based frameworks, which fall under the proposed categories of taxonomy in figure 2. One of them being Model based anomaly , another being authorizing anomaly detection and yet another being strategy based anomaly detection (the sub categories from the taxonomy provided in figure 2).

2.2.1 Model Based anomaly detection

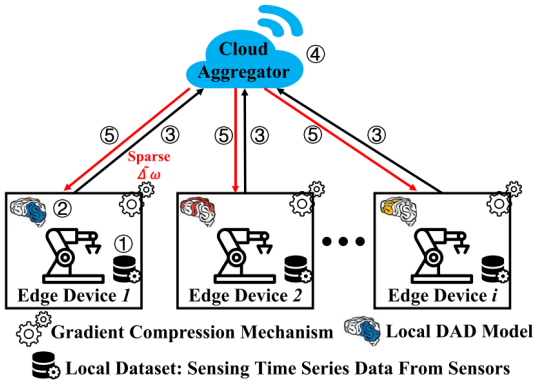
Framework presented in [1] is discussed in this section. It falls under the category of model based anomaly detection (sub-category) under the first main category in the taxonomy shown in figure 2. It was formally named as: Communication-efficient on-device FL based DAD framework. Its architecture and workflow is discussed below.

2.2.1.1 Architecture and workflow

In order to carry out DAD on-device within IIOT, a general scenario was setup in which DAD models were trained



(a) Workflow of the on-device DAD in IIoT. [1]



(b) Outline of the framework [1]

Figure 3: Communication efficient On-device FL-framework

with the contribution of the roles of edge devices and cloud aggregator(figure 3(a)). An on-device DAD framework was setup to ensure communication efficiency. The framework contains multiple edge devices to interactively train the model in IIOT as in figure 3(b). This framework has two components a cloud aggregator along with edge devices. Furthermore it has 2 mechanisms DAD (mechanism to detect anomalies) and GCM (mechanism to decrease communication overhead)[1].

When it comes to the components, the cloud aggregator is universally a powerful main server with great computing power and plentiful resources for computing. Two main functionalities of the cloud aggregator are to setup a global model to deploy it to all the edge devices and to aggregate the gradients that are uploaded via edge devices till the model gets converged. Next when it comes to the edge devices, they are the clients of main cloud. The edge devices use local data-sets for training a universal model that is sent by the cloud. The trained model is sent back to the cloud until convergence is achieved in the model. The local model that is contained in the edge devices is responsible for detecting anomalies. The local model was uniquely presented as "AMCNN-LSTM model" for accurate anomaly detection. This model is discussed further in the report.[1].

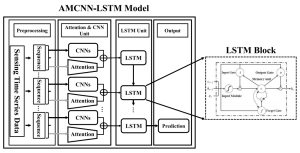
When it comes to the mechanisms, the DAD mechanism gets organised with the edge devices for detecting anomalies (inexpensive mechanism). Next when it comes to the GCM, it is the process that helps in mitigating the quantity of gradients transmitted among the cloud along with the edge devices, consequently resulting in reduced communication overhead [1].

Considering the components and mechanism, a workflow for the framework was established. The workflow goes on in this way: The sensitive time-series data aggregated from IIoT nodes, act as a local data-set used by edge devices. The local models (AMCNN-LSTM model) of these devices get trained based on their local data-set. The updated models (sparsed gradients) get uploaded to the cloud aggregator. All the received models are aggregated with the help of FEDAVG algorithm to obtain an advanced global model by the cloud aggregator. Finally, all edge devices in the network receive the updated global model from the cloud. This global model could be used for detecting anomalies timely and accurately. The above steps are iteratively executed until the global model reaches optimal convergence. The whole workflow is executed until the global model reaches its best state. Any decentralized devices could use this approach to carry out anomaly detection. Workflow is illustrated step by step in figure 3(b). ([1]).

2.2.1.2 Model: AMCNN-LSTM Model

AMCNN-LSTM Model is purely based on unsupervised approach. It contains an input layer, an AMCNN unit, LSTM unit together with an output layer as illutrated in figure 4(a). First the input layer uses the preprocessed-sensing time series data as its input. Second in the AMCNN unit, tiny features are captured by CNN units and a medium of attention to focus on the significant features captured by CNN unit. The output of AMCNN unit is used as the input for the third layer that is the LSTM unit, where future time series data is predicted. Finally anomaly detection scores were introduced which were used as ratings to detect anomalies.([1])

Steps of AMCNN-LSTM model in detail goes on this way: First Preprocessing, where the time series data aggregated by the edge devices are normalised into [0,1] to increase the speed of model convergence. Second AMCNN unit, it has 3 internal processes. Initially,an Attention mechanism was introduced in the CNN unit,to enhance the focus on the significant features. Attention mechanism was introduced to upgrade the potential of



(a) AMCNN-LSTM Model [1]

DETAILS OF FOUR REAL-WORLD DATA SETS

Datasets	Dimensions	X	X_n	X_a
Power Demand	1	1	45	6
Space Shuttle	1	3	20	8
ECG	1	1	215	1
Engine	12	30	240	152

(b) Real-world data sets [1]

Figure 4: A review of DAD

extracting features in numerous tasks like computer vision, speech processing. Hence the performance of the model was enhanced. Attention mechanism module performs feature aggregation and scale restoration. In the process of feature aggregation, convolution layers and pooling layers are piled up for extracting tiny features and a kernel matrix was convoluted with the data to obtain correlations among the extracted tiny features. In the process of Scale restoration, size of the output of a CNN module was made compatible with the key features, where the scaling of the key features are restored to (n × m), after which the values are constrained to [0, 1] using a sigmoid function. Then,CNN unit was introduced to extract fine grained features of time-series data. A single module of CNN was made by piling up multiple layers of 1-dimensional CNN units. Each layer has a layer of convolution, a layer to normalise the batch of data, and a non-linear layer. Corresponding CNN modules aggregate the input time series data samples with the help of pooling layers and thus creates various structures of hierarchies that successively extract important features by piling up convolutional layers. The CNN module outputs m feature sequences, of size n. Furthermore parallel branches are introduced in which attention mechanisms and CNN were combined to focus more on important features of time-series data. Then, the significant features computed by the attention mechanism and the sequential output features of the CNN module were multiplied. The product was used as the input of the following LSTM module. Third, One of the variants of RNN known as LSTM was used for accurate prediction of time series data and anomaly detection. LSTM contains 3 types of gates: input gate i_t, forget gate f_t and an output gate o_t. This gated structure is helpful in refining information .i.e. adding or deleting the information of the cell state. Fourth, Anomaly scores were used to detect anomalies. These scores predicted every point in a sequence of time series data as anomalous or normal.([1])

2.2.1.3 GCM

The communication among edge devices and the cloud takes place by interchanging gradients among each other. These gradients were mitigated for decreasing the communication overhead. Two approaches were adopted to mitigate gradients (in GCM algorithm [1]),they are:

(a) Momentum correction: This process converges the smaller gradients with the greater absolute values, consequently accelerating the speed of model convergence.[1]

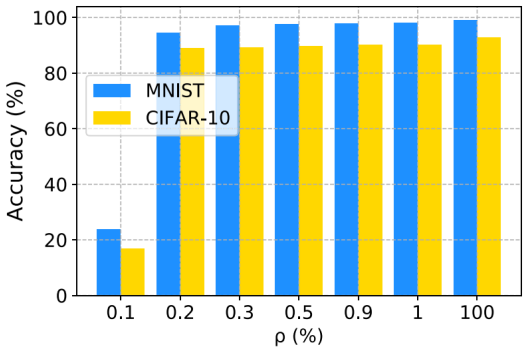
(b). Local gradient clipping: This process becomes helpful in reducing the gradient explosion problems.[1]

The implementation phases of GCM (gradient compression mechanism) as stated in [1], mitigates the gradients exchanged. The mitigation further contributes to increase in communication efficiency.

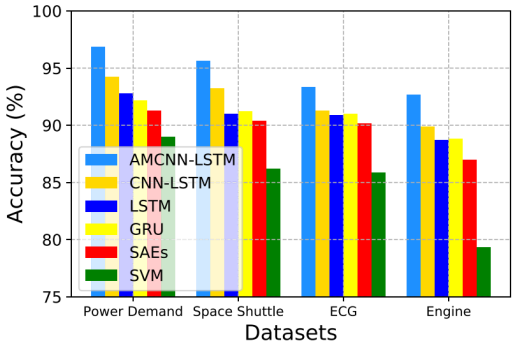
2.2.1.4 Investigation

The investigation was carried out on four data sets from the real world as in figure 4(b), X-original set, X_n-normal set, X_a-abnormal set. The data sets were specified in the ratio of 7:3 (training set:test set). Two open source libraries: Pytorch, PySyft were used. Data sets MNIST and CIFAR-10 datasets were normalised to [0,1]. Root mean square error (RMSE) was used to compute the performance of AMCNN-LSTM model. In the proposed deep-GCM,an appropriate threshold was selected to optimise the performance of the framework. Different values $\{0.1, 0.2, 0.3, 0.5, 0.9, 1, 100\}$ of threshold ρ were experimented to select a best value. There was a direct proportionality between the threshold values and the performance of the framework. The results were recorded as in figure 5(a). Also,the model reduces the gradient exchange upto 300 times, this reduction was immensely beneficial.([1])

To evaluate the performance of the framework, the AMCNN-LSTM model was compared with classical ML-based approaches. After simulations, two performance metrics accuracy and prediction error were compared. When considering accuracy, two parameters were considered: f_θ (accuracy), ς (recall rate). Different values of ς were used for different data sets. The experiment clearly indicated that AMCNN-LSTM model comparatively achieves highest values of accuracy. The results were recorded as in figure 5(b). The reason for this optimisation is the usage of FL based on device training, consequently increasing the robustness. Thus, edge devices can be



(a) Accuracy of the proposed framework with different thresholds on MNIST and CIFAR-10 data sets. [1]

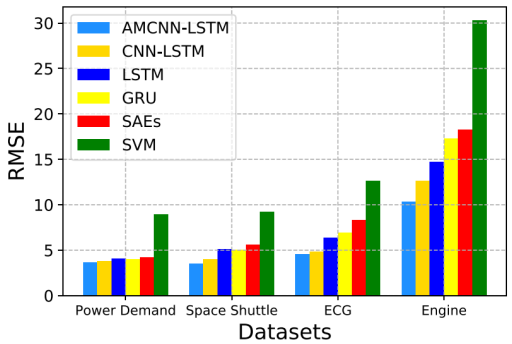


(b) Performance : detection accuracy comparison of proposed framework and other traditional frameworks on different datasets. [1]

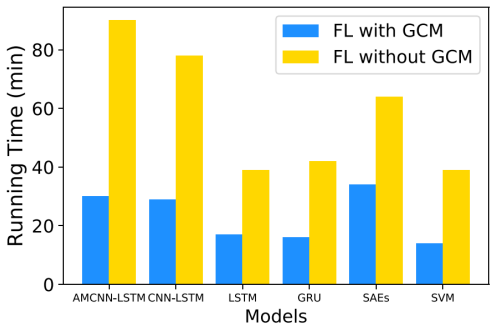
Figure 5: Performance

benefited to do timely updates. When considering prediction error, the experiment clearly indicated that AMCNN-LSTM model had comparatively less errors. The results of comparing prediction errors (RMSE-to calculate errors) were recorded as in 6(a). This optimization was due to the utilization of attention module, this module solves the gradient dispersion problems and memory loss problems by focusing on significant features. The proposed framework also was proven to be beneficial in predicting future time series accurately because of the retention of LSTM properties. Thus, the whole framework was overall proven to be accurate in anomaly detection and in time series prediction.([1])

Further, communication efficiency was evaluated. The proposed framework with GCM and classical methods in ML without GCM were simulated and compared. The figure clearly indicates that FL framework with GCM reduces the running time to almost 50% when compared to others. The results were as recorded in 6(b). This optimisation was because of GCM. Hence the framework was proved to be more practical and can be used in real time IIOT because of its optimizations in run time. Few advantages of this framework are: secure data access, optimised performance, accurate anomaly prediction, accurate time series prediction and minimised prediction error.([1])



(a) Performance : RMSE comparison of proposed framework and other traditional frameworks on different datasets. [1]



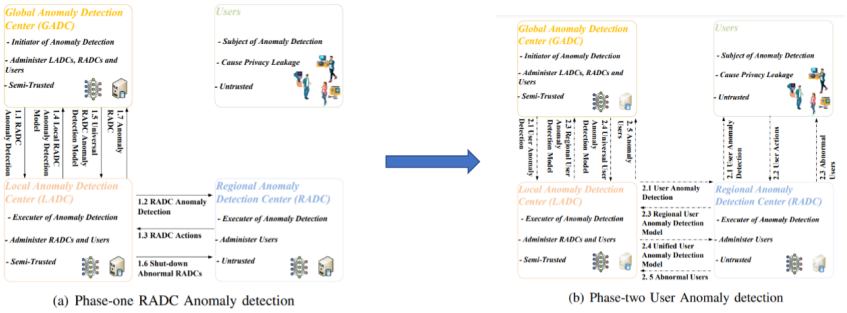
(b) Comparing communication efficiency between FL with GCM and FL without GCM with different models. [1]

Figure 6: Performance

2.2.2 Authorizing anomaly detection

The framework presented in [2] is discussed in this section. It addresses the fourth main category in the taxonomy shown in figure 2. It was formally named as : FLAD, where the anomaly detection was authorized based on the

hierarchy in Mobile edge computing (MEC) architecture. On the other hand MEC servers were used for operations in the network. Anomaly detection with preserving privacy and recognising abnormal ADCs (Anomaly detection centers) was performed. Few prerequisites to be kept in mind are: The framework FLAD was introduced with FDRL algorithm for anomaly detection based on variations (normality or abnormality) of ADCs. FDRL method was delineated based on privacy leakage degree to obtain a universal anomaly detection model for users with preserving privacy and security. Goal of the framework was to enhance anomaly detection accuracy and security aspects. An assumption was made that ADCs might be abnormal. In ADCs, anomalies were detected based on the internal or external actions. A mechanism of appeal was introduced for the normal users to reclaim their non-anomaly behavior (to enhance detection accuracy). The experiments were successful in demonstrating that the framework accurately detected abnormal users while maintaining a really high system throughput with a low latency. The architecture and workflow are discussed below. ([2])



(a) Phase-one RADC Anomaly detection

(b) Phase-two User Anomaly detection

Figure 7: FLAD Workflow [2]

2.2.2.1 Architecture and workflow

Lets address the framework in detail.The entities considered in the framework were: Global anomaly detection center(GADC),Local anomaly detection center (LADC),Regional anomaly detection center (RADC) and users. Entity whose actions exposed the sensitive information of other entities like positions, identities were considered as abnormal. ADCs use MEC servers (Mobile edge computing). MECs are interconnected. Since each end-user is connected to a specific region, anomaly detection on users could be performed by respective RADCs. Some RADCs might be incompetent of detecting anomalies, hence GADCs can be used in such cases. GADCs and LADCs might be semi-trusted, they maybe trustful but peculiar about user's sensitive information. RADCs might be treacherous, so abnormal RADCs can be detected by considering the differences among RADC actions. Differences in actions within RADC: between doubtful actions and regular actions is known as internal difference (Dominant factor for detection). Relation between RADCs-external difference (another dominant factor for detection). Models of normal RADCs were self trained, then transmitted to the GADC to construct a global model, then dispatched back to the RADC. FLAD framework is as illustrated in figure 7.([2])

FLAD is a 2 phase framework, its work flow consists of the following phases: [2]

Phase 1: Anomalous RADCs were detected by computing respective differences in their actions, then they were shutdown to confirm on highly accurate anomaly detection on users. GADCs performs anomaly detection on RADCs. Proficient LADCs perform training of local RADC models to detect anomalies and transmits it to GADC. GADC aggregates all these local models to construct a universal model and then deploys to every LADC. LADCs that are not proficient uses universal model for anomaly detection on RADCs. To perform this process FDRL strategy was adopted, DDPG algorithm was used to perform local RADC anomaly detection by LADCs. FL to perform global RADC anomaly detection. Anomalous RADCs were rated on a scale of (0 to 0.5). Actions were judged, anomalous RADCs were shutdown (offline) based on the scaling factor. Then proper actions and rewards were chosen based on online/offline state of RADCs. To increase detection accuracy, feeback mechanism was used to optimise the weights of the model. Optimization of reward function was the end point of the process.([2])

Phase 2: Anomalous users were detected by RADCs via privacy leakage using FL, this is totally different from detecting anomalies in RADCs. Focus of user anomaly detection was to avert from privacy leakage. Therefore privacy leakage degree was introduced into the model. Similar to phase 1,local user anomaly detection used DDPG algorithm, whereas global detection used FL strategy. A feedback mechanism was introduced to optimise the weights of the model. Normal and abnormal users were accordingly maintained by isolating the abnormal users, and if an RADC was shutdown at the same time, users were transferred to normal RADCs. Also to maintain

detection accuracy, an appeal mechanism was introduced where a user could prove its non-anomalous behaviour. Optimization of reward function was the end point of the process. Full workflow is illustrated in figure 7.([2])

2.2.2.2 Investigation

For investigation: a separate computer for the simulation of a GADC,10-computers for LADCs. RADCs were considered respectively. Performance was compared based System throughput, Latency and Anomaly detection accuracy. Throughput was varied based on RADC anomaly rate in phase 1 and user anomaly rate in phase 2. The proposed paradigm maintains the systems in the network in such a way that it detects anomalous users and anomalous RADCs very efficiently that the average value of a GADC throughput always remains high. Latency was varied with different user anomaly rates, RADC anomaly rates and the number of users. The latency was optimally reduced by FLAD, which resulted in increase in efficiency. For an accurate anomaly detection, FAR and MDR were taken into consideration, with the user anomaly rate and the RADC anomaly rate (with or without RADC anomaly detection). The proposed paradigm FLAD detects abnormalities so that FAR (false alarm rates) and MDR (missing detection rates) get sharply decreased. Few advantages of the framework are : Accurate anomaly detection, better privacy and security.([2])

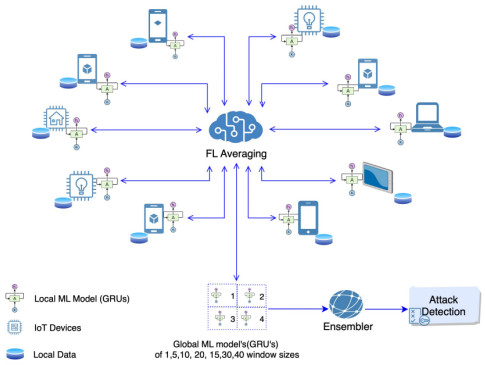


Figure 8: Ensembler-based FL framework [3]

2.2.3 Strategy based anomaly detection

The framework presented in [3] is discussed in this section that is based on ensemble learning. It falls under the category of strategy based anomaly detection under the first main category in the taxonomy shown in figure 2. It was formally named as- Detecting anomalies in the security attacks of IIOT. It is a framework which can recognise intrusion in IOT with a good extent of proactiveness in the networks of IIOT. An FL based model was used to perform federated training on GRUs keeping the data intact by only transmitting model weights. An ensembler was used to amalgamate the local model updates for optimising the accuracy of the model. This framework was compared with classical ML approaches and was proven to provide an optimal rate in detecting the attacks accurately. Using an ensembler was additionally more beneficial, which resulted in mitigating the error rates and false alarm rates when compared to classical ML models. The architecture of GRUs (Update gate and reset gate) are simpler when compared to that of LSTMs as implied by the source [3] in experimentation.([3])

2.2.3.1 Architecture and workflow

Lets address the framework in detail. The input sizes for each LSTMs/GRUs (in case of testing purposes) were varied with each window size (7 window sizes were considered). These broad sizes were significant to maintain an optimal performance of the ML model and a reduced training time. Window sizes in-turn rely on the type and size of data selected for bench-marking. Therefore overall setup of the architecture directly impacts the performance.([3])

The architecture includes a numerous virtual instances of IOT devices connected to a network, a DL model embedded in these instances, a component for averaging FL at the main server,a global model of DL for every window size and an ensembler which in turn consists of Random forest decision tree ensembler. ([3]) Steps in using the architecture are :([3])

The workflow goes on this way : Initially when it comes to virtual instances: Unique instances of IOT devices were cloned in the network using pysyft framework for flexibility in data sharing between the server and edge-devices. Modbus dataset was then segregated into n number of bundles and distributed over all the instances.

Then initialisation of data was captured: For initialising the data, network was captured in PCAP files and these files were converted into CSV files via CI-Cflowmeter tool. Using the CSV files, unwanted features were filtered out. Data was then segregated into n number of bundles and distributed across all the virtual instances. Then the virtual instances were trained using FL: An asynchronous training was performed on the available virtual instances with the local data-sets and weights were shared to the FL averaging component. To simulate the training, four GRUs were used with different varying layer sizes and window sizes. The training done on each device was known as an individual epoch (time periods). Algorithm presented in [3] describes the training process formally. Multiprocessors were introduced to enable a practical scenario realisation, where each virtual instance executes on a single processor. Then the learned model weights were synchronously shared with the server. $FL_{average}$ as in figure 8 serves as an aggregating agent, that collects all the model updates which arrive. Thus obtaining a global model for every window size after aggregation. Then a copy of this global model gets deployed to each end users. In the final step an ensembler was used: It uses the process of ensemble learning, which is beneficial for a higher value of accuracy rate and better efficiency. In this framework, Random forest decision tree classifier(rfc) was used for ensembling seven universal models representing different categories of attacks. Each universal model predicts a set of k probabilities. The probabilities were then combined by the rfc-ensembler to build a prediction function for ensembling. The function considers the predicted probabilities as votings for the data labels obtained from each model. The prediction function from the rfc ensembler always predicts the data labels with a high certainty as its output. Full architecture is illustrated in figure 8. ([3])

2.2.3.2 Investigation

To evaluate the performance the proposed model was compared with traditional ML models. Pysyft framework was used to generate virtual instances of IOT devices. Whereas, for traditional ML methods Pytorch framework was used. The performance of all the trained models were computed and compared. In the computations, the values for True positives(TP), True negatives(TN), False positives(FP) and False negativesO(FN) were computed. Where TN and TP indicates correctly predicted values, whereas FN and FP indicates the incorrectly predicted values. Accordingly training time was calculated separately for FL and non-FL approaches based on the factors used for experimental setup using accuracy, precision and recall. The proposed model with FL outperformed in evaluation. It acquires a higher accuracy by having a limited number of epochs. The overall time taken for every epoch in training was comparatively less due to parallel computing of virtual instances. Ensembler served as a great advantage because average detection accuracy value of every model was 99.5%, it resulted in minimising FAR (false alarm rates). Generally, average accuracy value was also better than non-FL strategies. Few advantages of the framework are : Accurate anomaly detection, better dynamicity and better security. ([3])

2.2.4 Analysis of the three models

This section includes the analysis of modelling factors and performance metrics from the above three models. A quick overview of comparison and analysis of the three models is presented in table 1. The present report makes its best efforts to analyze the above three models. Initially strengths are being analysed, then the weaknesses of the models are provided.

First, when it comes to the AMCNN-LSTM model [1], the authors used attention mechanism, CNN, LSTM for modelling and computed accuracy, prediction errors and run time [1]. Second, when it comes to the framework FLAD presented in [2], the authors used federated deep reinforcement learning and computed latency, system throughput and accuracy. Third when it comes to the ensembler based FL framework [1], they used an ensembler and computed variety of performance metrics. All the three models are critically important since they use innovative strategies to benchmark the models. The gradient boosting algorithm used by the framework presented in [1] serves as a great advantage since it minimizes the loss function and reduces the run time, thus contributing in reducing the communication efficiency [1]. The methods used to arrive at an optimized threshold for gradients, and to analyse the practicality of the model really inspires the future research inclinations [1]. The attention mechanism used in the same framework reduces the prediction error, further increasing the accuracy of anomaly detection, which is also inspiring. In addition, the innovative strategies used in the framework presented in [2] .i.e. FDRL algorithm and authorising anomaly detection inturn provides an inspiration for future research. Infact, the same model provides an incredible way of analysing network hierarchy to detect anomalies and performance by using reinforcement learning, which is another great inspiration. Additionally, the strategy used in the framework presented in [3] .i.e. ensemble learning also provides an inspiration. Overall, the unbiased insights used by all the models for implementation clearly shows that their strategies can be implemented in other models. For example: the models that are not yet migrated from ML to FL can utilize the strategies used by the the above three models because they are extremely relevant. ([1], [2], [3], [5], [7], [6])

Table 1: Analysis of the three models [1], [2], [3]

No.	Characteristic	AMCNN-LSTM model (FL based) [1]	FLAD, Privacy leakage degree [2]	Ensembler based , FL framework [3]
1.	Compared with traditional ML	Yes	No	Yes
2.	Compared with decentralized approaches	No	No	No
3.	Detection accuracy	99.08%	Variable according to network entity abnormalities	99.5%
4.	Prediction error	Yes	No	No
5.	False alarm rates	Not considered	Minimised	Minimised
6.	FP, FN,TN AND TP provided	No	No	Yes
7.	Communication efficiency considered	Yes	No	Yes
8.	Precision	No	No	Yes
9.	Speed	No	Yes	Yes
10.	Training time overhead	No	No	Optimised
11.	Recall rate	Yes	No	Yes
12.	Throughput	Not considered	Varied among other network entities	Not considered
13.	Latency	No	Yes	No
14.	Missing detection rate	No	Minimised	No
15.	Run time	Minimised	No	No
16.	Degree of abnormality	true to some extent	Yes	No

Although, the three models are extremely relevant, they possess their own limitations. Initially in the work presented in [1], an optimised performance was achieved but the model lacks defence mechanisms. There can be any unknown attacks on the edge devices at any point in time, hence integrating security and privacy aspects could be a future research inclination in this model (One hint: is to use differential privacy for added security) [6]. When it comes to the model presented in [2], it lacks evidences (for an optimised accuracy threshold for universal RADC model) and proofs for an optimised model convergence (in FL process). Anomaly detection accuracy in the same model still needs improvement, because the false alarm rates and missing detection rates can still be mitigated. One research direction in such a scenario can be to use deep Q learning in the same model [13]. Finally when it comes to the framework presented in [3], it lacks real time evidences for IOT. The same framework also does not specify, on what kind of attacks can the framework be used. One research direction could be to use differential privacy and investigate the provided model on real time industrial data sets. Overall, a diversity in performance metrics and modelling factors can be considered in future for the existing models, which will give deeper insights into the proposed models. ([1], [2], [3], [5], [7], [6])

3 Summary

To summarize, the report gives a good motivation to use FL in anomaly detection because of its decentralization and distributed architecture. Also, a brief overview of DAD is provided, where significance of DL is discussed in terms of anomaly detection with a taxonomic classification [4]. Since, FL optimises DL models, a taxonomic classification is provided based on the references being studied. Then to get deeper insights into the FL detection models, three relevant lately proposed FL models are discussed in detail. Where the initial model aggregates attention mechanism with CNN and LSTM to provide an accurate anomaly detection [1]. In addition to the AMCNN-LSTM model, the study [1] also proposes gradient compression mechanism (GCM) to increase the communication efficiency. The second model authorises anomaly detection locally, regionally and globally by using Federated deep reinforcement algorithm (FDRL) [2]. The third model uses ensemble learning, federated learning and GRUs to accurately detect anomalies [3]. Then an analysis of the three models is done. Then an outlook for future research is provided. Where a brief summary of the overall literature, thematic analysis, advantages, challenges and future directions were identified with possible solutions.

4 Conclusion

Extent of success in IIOT has been very large. Consequently, there exist lots of challenges due to the presence of large amount of un-noticed anomalies, advancements in the technology, evolution of interdisciplinary aspects and evolution of deep learning algorithms and models and much more to it. In the recent years, FL has served huge advantages in the field of ML. Hence, the present report, provides a brief overview of deep models in anomaly detection and a detailed discussion of FL models. It also gives its best efforts to provide a taxonomic classification

of the existing federated models in anomaly detection. An outlook is provided with the required details in brief for the future research. Where it is clear that more deep models, more evidences, more proofs and integration of advancements are required. Finally concluding, as said by Leonardo da Vinci "Learn how to see. Everything connects to everything else", hence all the studies were analysed, linked and a coherence was provided. It is expected that the depicted analysis will aid the future research in solving forthcoming challenges. Furthermore, a wide range of real time test-beds to aid future works are provided at : https://git.io/JTs93.

5 Outlook

This section provides a brief summary of the other relevant literature reviewed apart from the four studies that are used for discussion in detail for the previous sections, Table 2 provides a quick overview of the analysed literature. As discussed in previous sections, FL might have overcome the challenges in DL in detecting anomalies, but it has its own challenges. Hence few potentially identified challenges with the corresponding future directions are explained.

Out of all the details from table 2, few of the important aspects to be kept in mind are: When a framework is designed, it has basically two aspects Modelling and performance. Where in terms of modelling, future researchers can consider interconnecting higher level architectures like attention mechanism, CNN (Convolutional neural networks) [1], GNN (Graph neural networks) [17], KNN (Kervolutional neural networks) [18], GRUs [3], LSTMs [1], auto-encoders [19] and other architectures listed in table 2 used by respective studies. Also modelling can consider diversity in learning strategies, for example: in the work presented in [20] it was proven that split learning performs better than federated learning in case of imbalanced data distributions, whereas federated learning performs better than split learning in case of non-IID data distributions [20]. Such aspects can be taken into consideration, where a single model can be bench-marked to utilize different learning strategies, that is data specific or can be specific to any modelling factors to detect anomalies. As we previously discussed that complexity of DL in the field of anomaly detection does not match with its entire research area [4], more deep models can be developed so that complex anomalies can be easily detected. Another aspect that modelling could consider is diversity in anomalies, where the diversity includes patch wise anomalies, orientation of anomalies, noised anomalies etc [4]. The study [4] also says that bench-marking a single model to detect different kinds of anomalies is significant. Similar to the multitasking framework presented in [21], where a multitasking anomaly detection in the network was developed using FL. Many such frameworks can be developed. Also, in case of diverse anomalies, advancements done to overcome the disadvantages of CNN can be considered, for example: KNNs use patch wise kernel functions [18], which can be utilized to detect patch-wise anomalies easily [22]. Another example being capsule based convolutional neural networks which retains all the information and keeps track of data orientation, which can be used for detecting rare anomalies and for detecting anomalies that are uncertainly oriented [22]. For example: as the work presented in [22], detects malicious activities in social media networks by using both capsule based CNNs and capsule based KNNs [22]. Another example being the study presented in [23], in which they investigated the potential of KNNs in classification of time series data and anomaly detection (anomalies occuring in the time series data of helicopters) [23]. They also demonstrated that the amalgamating convolutional and kervolutional layers provides a better performance [23]. This mixture model was used as a bench-mark data set for the classification of time series [23]. They also proved that the mixture model provides a better anomaly detection in case of largely diverted anomalies [23]. Such advancements are less explored. Hence above listed examples are evident that advancements in CNN will be beneficial in potential classification and identification (example : combination of convolutional and kervolutional layers [23]), if utilized in IIOT , by integrating with FL for better results. Yet another advancement to CNN where Graph based CNNs were introduced which is a more generalised version of CNNs, they can be used to detect more generalised anomalies (eg: in graphically dependent non-IID data sets), similar to the work done in [17], where anomalies were detected in email communication networks and twitter data sets [17]. Such scenarios can be integrated with FL to obtain better results. This was an essential outlook on modelling.

When it comes to performance, future researchers should consider diversity in performance metrics, as the work presented in [3] considered diverse performance metrics, other studies can also consider diversity in performance metrics because it gives unbiased insights into the model being bench-marked. Also, performance can consider gradient boosting algorithms in ML, similar to the study presented in [1], where they used gradient compression mechanism, which contributed to higher anomaly detection accuracy and higher efficiency in communication [1]. Performance also depends on modelling, for example in the study [24], the anomaly detection accuracy was 94.23% and the prediction error was 57.7% lower than that of SVM model. Whereas, in the subsequent study of [1], the anomaly detection accuracy was 96.85% and the prediction error was 63.9% lower than that of SVM model. Where in the latter model, integration of attention mechanism improved the performance. Another

factor that affects performance is: learning strategies, for example: in the study of [13] it was proven that FL can be optimised with integrating deep Q learning. Where they proved that the number of communication rounds are reduced to almost 49% in case of MNIST data sets, 23% in case of fashion MNIST data sets and 42% in case of CIFAR-10 data sets [13]. Hence previous models that use similar data sets can be considered to integrate deep Q learning [13]. Multitasking modules of a framework improves performance as proven in the study of [21]. Hence multitasking can be taken into consideration for future research. For example: bench-marking a single model to detect diverse anomalies (as discussed above). Supervision of anomalies as stated by [4] affects performance. For example: anomalies can be supervised with domain based knowledge before detection [4]. Yet another factor that will provide deeper insights into performance is: diversity in analysis in a single model. An individual model or an algorithm can be bench-marked with different types of analysis (Quantitative analysis, qualitative analysis, predictive analysis, diagnostic analysis) for anomaly detection, which would be evident to provide deeper insights for future research as stated by [4]. Furthermore, anomaly detection requires an integration of differential privacy in case of issues related to security and privacy [6]. The source [6], provides insights on how differential privacy can be utilized. This was an essential outlook on performance.

The overall research topic of the present report was analysed thematically. Figure 9 provides a quick overview of thematic analysis. Where the present report makes its best efforts to analyse the research topic broadly into seven categories as illustrated in figure 10. One of the uniquely identified themes is visual analysis, which means analysing anomalies visually. For example: the study presented in [15] develops a visual analytics technique for horizontal federated learning (HFL).An inherent exploration of HFL was done for inspecting malicious client behaviors [15]. Where they visually analysed the relationships of clients in the HFL architecture, potentially significant anomalies were identified and the client contributions were assessed [15]. The visual technique for HFL was proven to provide a better understanding and diagnosis of the HFL architecture [15]. Such aspects could be considered for further research inclinations because visual analysis in the field of federated anomaly detection is less explored. Another uniquely identified theme was: possible data sets. Researchers from the study of [4] have provided possible challenging test beds on the site `https://git.io/JTs93` [4]. They also provide a detailed review of the algorithms of DL that requires further improvement [4]. One of the unique works found in anomaly detection was [25] (aging IOT), where it delineates statistical analysis of the abnormal process events in IOT devices. Applicability of such a strategy was proven to be superior in IIOT when compared to baseline methods in its native context. Many such models can be developed.

Challenges and future directions :

Few of the identified general challenges are: Initially, communication overhead which is very high due to the transmission of modelling parameters to and from the central server [7]. As a result the throughput of the federated network is decreased and the resultant central server becomes a congestion in FL [7]. Possible future directions could be to utilize binary neural networks, 5G and 6G communication protocols, gradient boosting mechanisms and asynchronous FL ([7], [26], [1], [4], [6]). Another, challenge would be that, the distributed architecture of FL makes it more complex to inculcate security measures [26]. Possible future directions could be to use blockchain technology with FL similar to the work done in [27] for anomaly detection. Another challenge could be network latency, which is due to the central server waiting for the local model updates [7]. Possible future directions for latency could be to utilize DL architectures which could reduce the latency [4]. Yet another challenge could be that resource management becomes difficult because of distributed learning. Possible future directions for this could be to use a learning rate scheduler as used in the study of [28]. Few other challenges could be that: some of the current FL models are still being bench-marked by comparing with classical ML models. Instead they could be compared with the existing FL models, which will aid future research. As discussed in the above sections, complexity of DL is less explored [4]. Backdoor attacks in FL is less explored [29], diversity in anomalies is less explored [4] and diversity in performance metrics is less explored. Integration of advancements to fedAvg algorithm and MEC architecture is less explored. Research similar to the works of [12],[14], [30] could be further carried out.

Some specific challenges are : In the study of [2], in the proposed two phase architecture, at the end of both the phases the universal anomaly detection model gets converged and optimization of reward function was considered as an end point [2]. But when and how the FL learning process gets converged is still left as an open challenge [2]. One research direction could be, to utilize the methods of clustering. Where clustering can be performed in different levels of the network hierarchy to optimise model weights [7]. This direction in-turn reduces the diversity of client updates. The reduction will lead to a conclusion on how and when the learning process would get converged. Another alternative could be to utilize biased or unbiased client selection ([14], [12]) in the fedAvg algorithm, which will in-turn lead to a faster conclusion on when and how the learning process gets converged. In the same study, there does not exist any data sets or evidence to prove if universal RADC detection model could satisfy specified accuracy threshold [2]. One research direction could be, to experiment with different threshold values to choose an optimized threshold, similar to the strategies adopted in [1].

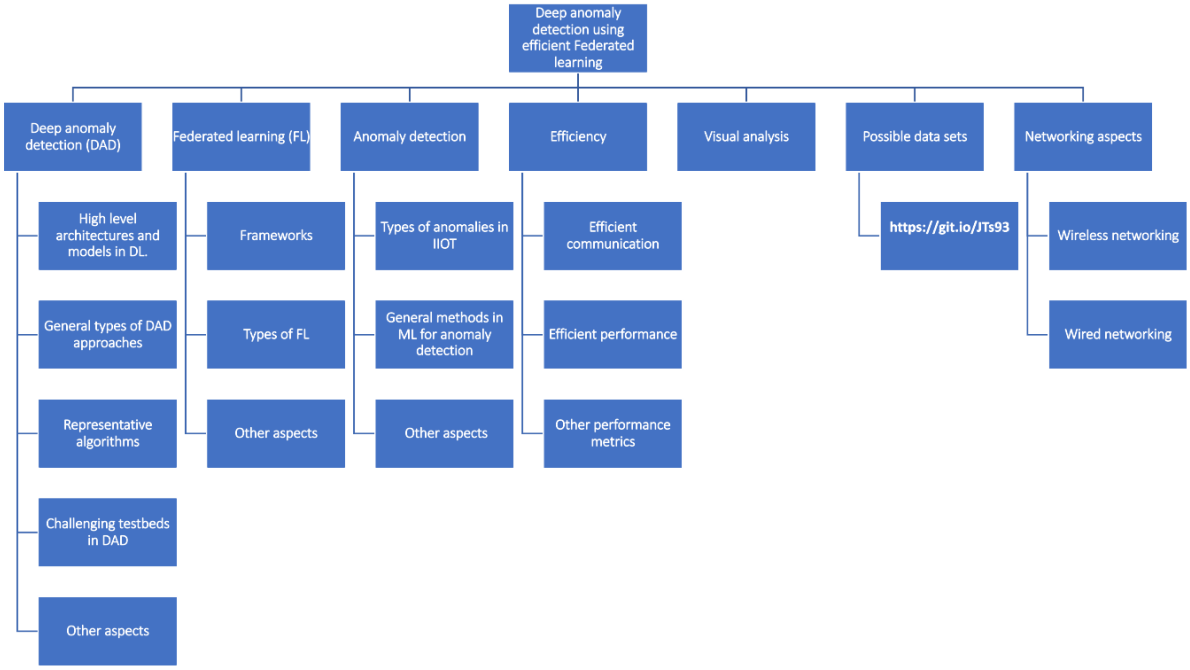


Figure 9: A thematic analysis of the research topic [1], [3], [2], [4], [15]

Whereas, in the study of [15], first a quantitative analysis of the visual analytics on HFL was not provided. Second, the attacking algorithms that lack distinctive performance are not detected by their approach [15]. A new visual analytical method could be introduced to detect such algorithms [15]. One of the examples of such algorithms is: Algorithms that detect morphing attacks. Third, focus of diversity in performance metrics could be increased based on the type of anomalies to be detected and the model used [15]. Fourth, Hardware and the client configuration did not possess privacy information [15].

Also, the framework presented in [3] could be investigated with real time IOT devices because currently it uses virtual instances of IOT. Testing with real time IOT will also mitigate cold start problems [3]. Since an ensembler is used, it can be tested with more challenging test beds because from its fundamental instincts its clear that the framework can be used in wide variety of domains [3]. The framework does not provide a classification of conspicuous and inconspicuous attacks. One research direction could be to monitor the actions done by the user, similar to the monitoring done in the framework presented in [2].

References

[1] Yi Liu, Sahil Garg, Jiangtian Nie, Yang Zhang, Zehui Xiong, Jiawen Kang, and M. Shamim Hossain. Deep anomaly detection for time-series data in industrial iot: a communication-efficient on-device federated learning approach. *IEEE Internet of Things Journal*, 8(8):6348–6358, April 2021.

[2] Xiaoding Wang, Sahil Garg, Hui Lin, Jia Hu, Georges Kaddoum, Md. Jalil Piran, and M. Shamim Hossain. Towards accurate anomaly detection in industrial internet-of-things using hierarchical federated learning. *IEEE Internet of Things Journal*, pages 1–1, 2021.

[3] Viraaji Mothukuri, Prachi Khare, Reza M. Parizi, Seyedamin Pouriyeh, Ali Dehghantanha, and Gautam Srivastava. Federated learning-based anomaly detection for iot security attacks. *IEEE Internet of Things Journal*, pages 1–1, 2021.

[4] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep Learning for Anomaly Detection: A Review. *ACM Computing Surveys*, 54(2):1–38, April 2021.

[5] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated Machine Learning: Concept and Applications. *ACM Transactions on Intelligent Systems and Technology*, 10(2):1–19, February 2019.

Table 2: A brief scientific summary of related literature reviewed

No.	Src	Delivery characteristic	Details	Details on performance
1.	[28]	FedIoT platform ,N-BaIoT dataset ,adaptive optimizer ,a rate scheduler for cross round learning.	Enhancement in algorithm and system design, Evaluated on both model and system performance.	Efficacy in FL for detecting huge range of attacks, Optimised end to end training time and memory cost
2.	[27]	Direct Acyclic Graph (DAG)-based blockchain (DAG-FL) framework	3 layer architecture, Two algorithms based on DAG-FL for controlling and updating	Better training efficiency and model accuracy
3.	[31]	TrustFed- a blockchain-based framework	To detect poisoning attacks of a model , to maintain fair training settings, and device reputation.	Lower model loss
4.	[32]	A classification model based on detecting intrustion in the networks	CNN-FL,nsl-kdd dataset(vector based-large), binary classification, multiple classifications, transmission of gradient related data that is encrypted	Higher performance and accuracy of classification
5.	[15]	System for visual analysis in horizontal federated learning (HFLens),	Good visual interpretation of clients, communication rounds, recognising anomalies and better diagnosis of HFL	Efficacy of the system,
6.	[33]	Fedcom , a robust byzantine based FL framework	Cryptography, Non-IID data (stochastic gradient-based), wasserstein distance to recognise poisoned attacks, MNIST datasets, HAR datasets.	Better global model accuracy
7.	[34]	Deepant, Time series predictor (deep convolutional neural network - DCNN),anomaly detector (tag-normal/abnormal)	Small dataset,Good capability to generalise, detects wide range of anomalies	Better precision, detection accuracy, recall rate and F- scores as compared to previous approaches.
8.	[35]	DIOT (Autonomous framework- self learning)	Detects compromised IOT devices(detecting of intrusion), device specific communication profile is built automatically , used Mirai malware	95.6% detection rate, 257 ms speed, no false alarms for smart homes
9.	[24]	FL with DAD,CNN-LSTM model, GCM	CNN-LSTM model for accurate anomaly detection, GCM to increase communication efficiency	94.23% detection accuracy, Prediction error:57.7% lower than classical classical methods Better communication overhead
10.	[25]	Framework to detect physical decay of IIOT devices using Hotelling's T2 statistics and univariate cumulative sum to detect abnormalities	For recognising Aging IIOT , by considering physical dimension as an aging factor, systems performance reduce with age.(stealth attacks)	Performance on continuous stirred-tank reactor (CSTR) modelwas superior
11.	[19]	Composite-autoencoder model	Detecting anomalies based on distribution of errors, SWaT dataset,	88.5% recall rate, 87.0% F1-score
12.	[36]	Decision tree based solution for anomaly detection	3 decision trees, smart grids,Investigates the intrusion detection problem, NSL-KDD dataset, Cyber attacks	Better performance than IOT based SG
13.	[37]	Approach based on LSTM and Gaussian Bayes method(GNB)	For detecting anomalies , LSTM-constructs data of normal time-series, abnormalities are detected by using predictive error by using GNB model, good classification	Better accuracy, precision, recall rates and F1 scores
14.	[8]	Survey on DAD models	Research of DAD models for anomaly detection, Challenges, complexities, limitations, advantages	survey
15.	[4]	Review on research studies of DAD	Intuitions, objective function, assumptions , Future opportunities	Review
16.	[38]	DeepFed , FL based intrusion detection framework, Pailier cyptosystem	CNNs with GRUs for deetcting threats, Pailier cryptosystem to maintain security of all models on the basis of secure communication protocols	Accuracy above 99%, Precision above 98%, recall rate above 96%, F1 scores above 97%
17.	[39]	FedAGRU algorithm	FL based attention GRU,to recognise poisoning attacks, Elimination of minimal contribution updates 3 datasets - optimal results,	8% improvement in detection accuracy than centralized approaches, communication costs 70% lesser than FL algorithms
18.	[40]	FL based wireless intrusion detection(WID), awid dataset	Evaluation on AWID dataset	Effective classification accuracy, computation and communication cost
19.	[41]	TensorFlow federated framework	FL based IDS, NSL-KDD dataset	Better accuracy than previous approaches
20.	[42]	TensorFlow DL framework(centralized)	6 LSTMs of unique layers used for detecting threats, cyber attacks	Efficient model, 99% accuracy in detecting cyber attacks.
21.	[43]	Basic AI neural network classification and logistic regression applied.(can be used to prevent attacks,smart manufacturing, IOT solutions and smart devices)	For identifying attacks and their patterns, unique attributes used for anomaly detection in smart homes	2 cases : with featured datasets 99.4% accuracy andwithout features 99.9% accuracy
22.	[9]	Multitask learning framework Stacked LSTM Fl model based on privacy with design	Twice the speed of convergence when compared with centralized LSTM and State of art performance compatible with classification as well as regression tasks	2 cases : Mitigates overall training costs without decreasing prediction performance.
23.	[21]	Multi-Tasking Deep Neural Networks framework based on FL	Multiple tasks performed: anomaly detection in networks, traffic recognition, classification task	Training time overhead was decreased, better than classical methods in ML.

[6] Edited by: Peter Kairouz and H. Brendan McMahan. Advances and Open Problems in Federated Learning. *Foundations and Trends® in Machine Learning*, 14(1), 2021.

[7] Shaashwat Agrawal, Sagnik Sarkar, Ons Aouedi, Gokul Yenduri, Kandaraj Piamrat, Sweta Bhattacharya, Praveen Kumar Reddy Maddikunta, and Thippa Reddy Gadekallu. Federated learning for intrusion detection system: Concepts, challenges and future directions. *arXiv preprint arXiv:2106.09527*, 2021.

[8] R.Chalapathy and S Chawla. DEEP LEARNING FOR ANOMALY DETECTION: A SURVEY. *arXiv:1901.03407 [cs.LG]*, 2019.

[9] A. Ben Hamza Raed Abdel Sater. A Federated Learning Approach to Anomaly Detection in Smart Buildings. *ACM*, May 2021.

[10] Xuefei Yin, Yanming Zhu, and Jiankun Hu. A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions. *ACM Computing Surveys (CSUR)*, 54(6):1–36, 2021.

[11] Latif U Khan, Walid Saad, Zhu Han, Ekram Hossain, and Choong Seon Hong. Federated learning for internet of things: Recent advances, taxonomy, and open challenges. *IEEE Communications Surveys & Tutorials*, 2021.

[12] Takayuki Nishio and Ryo Yonetani. Client selection for federated learning with heterogeneous resources in mobile edge. 04 2018.

[13] Hao Wang, Zakhary Kaplan, Di Niu, and Baochun Li. Optimizing federated learning on non-iid data with reinforcement learning. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*, pages 1698–1707. IEEE, 2020.

[14] Yae Jee Cho, Jianyu Wang, and Gauri Joshi. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies. *arXiv preprint arXiv:2010.01243*, 2020.

[15] Quan Li, Xiguang Wei, Huanbin Lin, Yang Liu, Tianjian Chen, and Xiaojuan Ma. Inspecting the running process of horizontal federated learning via visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–1, 2021.

[16] Zhipin Gu and Yuexiang Yang. Detecting malicious model updates from federated learning on conditional variational autoencoder. In *2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 671–680. IEEE, 2021.

[17] Anshika Chaudhary, Himangi Mittal, and Anuja Arora. Anomaly detection using graph neural networks. In *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, pages 346–350, 2019.

[18] Chen Wang, Jianfei Yang, Lihua Xie, and Junsong Yuan. Kervolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 31–40, 2019.

[19] Chao Wang, Bailing Wang, Hongri Liu, and Haikuo Qu. Anomaly detection for industrial control system based on autoencoder neural network. *Wireless Communications and Mobile Computing*, 2020:1–10, 08 2020.

[20] Yansong Gao, Minki Kim, Sharif Abuadbba, Yeonjae Kim, Chandra Thapa, Kyuyeon Kim, Seyit A. Camtep, Hyoungshick Kim, and Surya Nepal. End-to-end evaluation of federated learning and split learning for internet of things. In *2020 International Symposium on Reliable Distributed Systems (SRDS)*, pages 91–100, 2020.

[21] Ying Zhao, Junjun Chen, Di Wu, Jian Teng, and Shui Yu. Multi-task network anomaly detection using federated learning. In *Proceedings of the Tenth International Symposium on Information and Communication Technology*, SoICT 2019, page 273–279, New York, NY, USA, 2019. Association for Computing Machinery.

[22] Chanchal Suman, Ayush Raj, Sriparna Saha, and Pushpak Bhattacharyya. Authorship attribution of microtext using capsule networks. *IEEE Transactions on Computational Social Systems*, 2021.

[23] Oliver Ammann, Gabriel Michau, and Olga Fink. Anomaly detection and classification in time series with kervolutional neural networks. *arXiv preprint arXiv:2005.07078*, 2020.

[24] Yi Liu, Neeraj Kumar, Zehui Xiong, Wei Yang Bryan Lim, Jiawen Kang, and Dusit Niyato. Communication-efficient federated learning for anomaly detection in industrial internet of things. In *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, pages 1–6, 2020.

[25] Béla Genge, Piroska Haller, and Călin Enăchescu. Anomaly detection in aging industrial internet of things. *IEEE Access*, 7:74217–74230, 2019.

[26] Priyanka Mary Mammen. Federated Learning: Opportunities and Challenges. *arXiv:2101.05428v1 [cs.LG] 14 Jan 2021*, January 2021. arXiv:2101.05428v1 [cs.LG] 14 Jan 2021.

[27] Mingrui Cao, Long Zhang, and Bin Cao. Towards on-device federated learning: A direct acyclic graph-based blockchain approach. *conference paper*, 2021.

[28] Tuo Zhang, Chaoyang He, Tianhao Ma, Mark Ma, and Salman Avestimehr. Federated learning for internet of things: A federated learning framework for on-device anomaly data detection. *arXiv preprint arXiv:2106.07976*, 2021.

[29] Zhuosheng Zhang, Jiarui Li, Shucheng Yu, and Christian Makaya. Safelearning: Enable backdoor detectability in federated learning with secure aggregation. *arXiv preprint arXiv:2102.02402*, 2021.

[30] Chandra Thapa, M.A.P. Chamikara, and Seyit Camtepe. Splitfed: When federated learning meets split learning. *arXiv:2004.12088v1 [cs.LG]*, 04 25 April,2020.

[31] Muhammad Habib ur Rehman, Ahmed Mukhtar Dirir, Khaled Salah, Ernesto Damiani, and Davor Svetinovic Center. Trustfed: A framework for fair and trustworthy cross-device federated learning in iiot. *IEEE Transactions on Industrial Informatics*, 2021.

[32] ZHAO Ying, WANG LiBao, CHEN JunJun, and TENG Jian. Network anomaly detection based on federated learning. *Journal of Beijing University of Chemical Technology*, 48(2):92.

[33] Bo Zhao, Peng Sun, Liming Fang, Tao Wang, and Keyu Jiang. Fedcom: A byzantine-robust local model aggregation rule using data commitment for federated learning. *arXiv preprint arXiv:2104.08020*, 2021.

[34] Mohsin Munir, Shoaib Ahmed Siddiqui, Andreas Dengel, and Sheraz Ahmed. DeepAnT: A Deep Learning Approach for Unsupervised Anomaly Detection in Time Series. *IEEE Access*, 7, 2019.

[35] Thien Duc Nguyen, Samuel Marchal, Markus Miettinen, Hossein Fereidooni, N Asokan, and Ahmad-Reza Sadeghi. Dïot: A federated self-learning anomaly detection system for iot. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pages 756–767. IEEE, 2019.

[36] Seyedeh Mahsan Taghavinejad, Mehran Taghavinejad, Lida Shahmiri, Mohammad Zavvar, and Mohammad Hossein Zavvar. Intrusion detection in iot-based smart grid using hybrid decision tree. In *2020 6th International Conference on Web Research (ICWR)*, pages 152–156, 2020.

[37] Di Wu, Zhongkai Jiang, Xiaofeng Xie, Xuetao Wei, Weiren Yu, and Renfa Li. Lstm learning with bayesian and gaussian processing for anomaly detection in industrial iot. *IEEE Transactions on Industrial Informatics*, 16(8):5244–5253, 2020.

[38] Beibei Li, Yuhao Wu, Jiarui Song, Rongxing Lu, Tao Li, and Liang Zhao. Deepfed: Federated deep learning for intrusion detection in industrial cyber–physical systems. *IEEE Transactions on Industrial Informatics*, 17(8):5615–5624, 2021.

[39] Zhuo Chen, Na Lv, Pengfei Liu, Yu Fang, Kun Chen, and Wu Pan. Intrusion detection for wireless edge networks based on federated learning. *IEEE Access*, 8:217463–217472, 2020.

[40] Burak Cetin, Alina Lazar, Jinoh Kim, Alex Sim, and Kesheng Wu. Federated wireless network intrusion detection. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 6004–6006, 2019.

[41] Sawsan Abdulrahman, Hanine Tout, Chamseddine Talhi, and Azzam Mourad. Internet of things intrusion detection: Centralized, on-device, or federated learning? *IEEE Network*, PP:1–8, 09 2020.

[42] Mahdis Saharkhizan, Amin Azmoodeh, Ali Dehghantanha, Kim-Kwang Raymond Choo, and Reza M. Parizi. An ensemble of deep recurrent neural networks for detecting iot cyber attacks using network traffic. *IEEE Internet of Things Journal*, 7(9):8852–8859, 2020.

[43] Nilesh Kumar Sahu and Indrajit Mukherjee. Machine learning based anomaly detection for iot network: (anomaly detection in iot network). In *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, pages 787–794, 2020.