Adaptive Secure Threat Intelligence Infrastructure for AI and the Edge

Tulasi Kasuba
Research Scholar
Dept. of Computer Science and Engineering
Annamalai University
Annamalainagar – 608002
Email: ktulasi2907@gmail.com

Dr.S.Saravanan
Assistant Professor
Dept. of Computer Science and Engineering
Annamalai University
Annamalainagar – 608002.
Email: ssaravau@gmail.com

Dr.V.V.S.S.S.Balaram
Professor and Head
Sreenidhi Institute of Science and Technology
Yamnampet, Ghatkesar,
Hyderabad – 501301.
Email: vbalaram@sreenidhi.edu.in

Article Info
Page Number: 1459-1474
Publication Issue:
Vol. 71 No. 4 (2022)

Abstract—Cyber security warfare is actively fought on both sides by AI and machine learning, which allows both adversaries and defenders to engage at unprecedented speeds and scales. For intrusion detection system intelligence, artificial intelligence and machine learning are essential for managing the huge amount of information and assuring the credibility of such data. Dealing with massive amounts of data transitions can lead to major issues termed security challenges. A new learning strategy called Federated Learning (FL) facilitates deep neural network training between numerous dispersed edges nodes without the need for transmitting data thus addressing privacy concerns. Using a federated forest algorithm, a safer cross regional deep learning system is put forward that enables training to be collectively supervised over edge nodes in various regions locations using the same node sample set and yet different feature sets, collecting and analyzing data retained in every one of them while not transferring their original information.

Keywords: - Cloud Computing, Edge Computing, Serverless Computing, Threat detection, Machine learning, Deep learning techniques, Federated Learning.

Article History
Article Received: 25 March 2022
Revised: 30 April 2022
Accepted: 15 June 2022
Publication: 19 August 2022

I. INTRODUCTION

Artificial Intelligence (AI) or machine learning is the emerging jargon being used by the cyber security field to upgrade and develop security measures. Machine learning is considered necessary since attacks are becoming more technologically advanced. These two fields have a long tradition of innovation and their skills cover a wide range of application domains. These are not related specifically to cyber security. By exploiting these skills in other areas, a lot can be learned about how to apply similarly to threat intelligence. For instance, we can learn about information security responses from developments in autonomous bots reactions for support websites. Machine learning and Natural Language Processing (NLP) are both necessary due to the variety of user expressions and the diverse results users hope to achieve [1][2]. Such a system is in many ways comparable to how a cyber resilience response should behave in response to various threat patterns, actor motivations and defensive consequences anticipated in the security field.

Corporate executives are boosting digitization plans quickly in 2022 to support entrepreneurial ventures and expansion plans. Organization's need for real time information and analytics is a substantial growth catalyst to edge devices and computing. It is anticipated that global expenditure on edge device computing to hit 176 billion dollars by 2022 and will increase to 274 billion dollars in 2025 [3][36]. By 2023, more than 50 percent of information generated and recorded by corporations would be processed outside information centres or on the cloud, up from less than 10 percent in 2019.

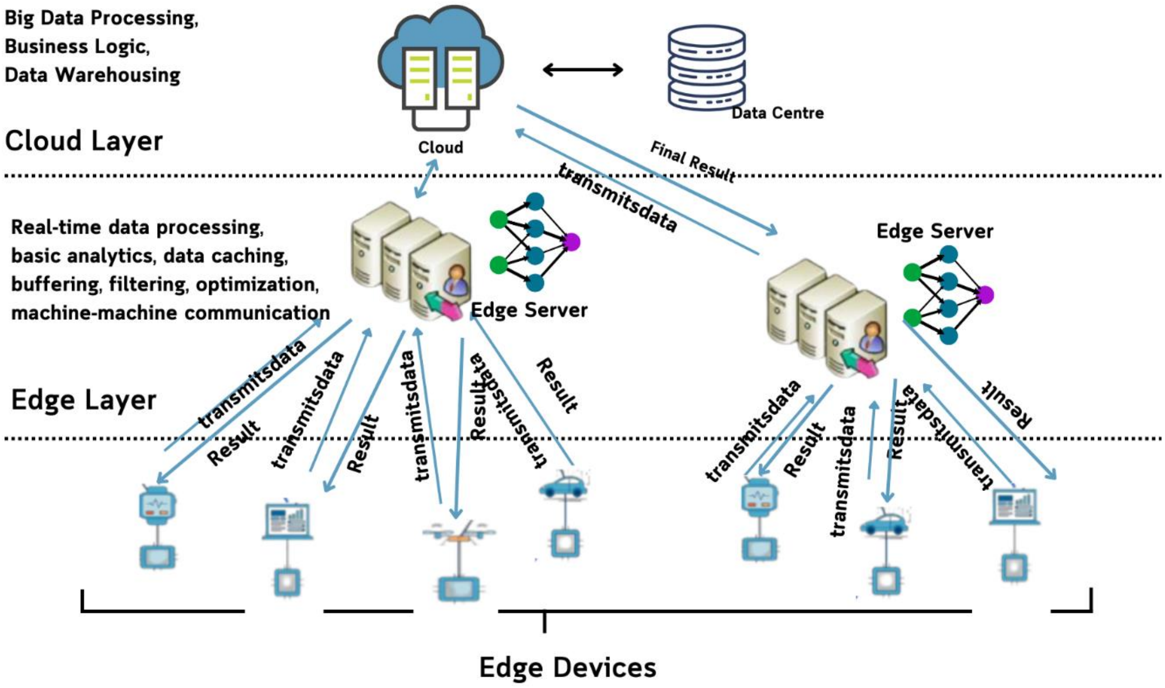


Figure 1 AI data processor and Edge node

Rather than keeping data in the cloud or a centralized information repository, IT personnel can deliver information processing capability at the network edge [37]. It has so been regarded as an appropriate choice for various applications like Internet of Things (IoT) and manufacturing. They can gather data streams using edge devices which would avert device flaws. They do not even play the role of cloud computing, though, in terms of global administration. Applications that analyze and maintain information entirely through centralized cloud computing environments is vulnerable to delay and unavailability if web access is sluggish or regularly disturbed. It can take a long time to communicate requests to clouds, get them evaluated and report the results. These shortcomings of cloud computing paved the way to edge computing, where it relocates computational resources to the actual place of information generation or the internet's so-called edge. The claimed benefits include real-time velocities and significantly improved system reliability, flexibility, robustness and consistency [4][38].

Rapid innovations in AI capabilities increase uptake of Internet of Things (IoT) systems, as well as the potential of edge computing, which have combined to unveil the prospects of edge AI. In ways that facilitate our personal lives, at the office, in academia and in transit, edge technologies are propelling us to the next phase in AI. Edge AI refers to implanting AI software on equipment all across the real world [39][40]. The rationale is named “Edge AI” because, as opposed to being conducted centrally in the public cloud or in an in-house data centre, the AI data processing is handled closer to the consumer of the network edge, close to where the information is stored as shown in Figure 1. The edges of the internet can refer to any area as the online world is accessible from everywhere. It might be a department shop, industry, medical centre or one of the devices we see often, such as traffic signals, bots or mobiles.

The effectiveness of deploying AI at the edges is a consequence of three major developments:

- Neural networks: Have ultimately reached a stage of maturity that makes generalized ML feasible, along with accompanying AI architecture. Organizations are gaining knowledge on how to effectively educate artificial intelligence system models and use them in manufacturing at the edge.
- New developments in computing infrastructure: To operate intelligence at the edge, significant computational power is needed. Neural network processing is now possible because of recent developments in massively parallel graphical processing units.
- Connected devices acceptance: The growth of big data has been propelled by the broad use of IoT. Researchers now have the data-driven insights to operate intelligent algorithms at the edge thanks to the rapid ability to gather information from every part of an enterprise including sensing devices, intelligent cameras, robotics and much more. In addition, 5G is enhancing connected communication with quicker, more reliable and secure messaging.

In every industry, including production, health, financial sectors, aviation and energy, edge AI has been creating great business outcomes.

A. Motivation

Intelligence at edge sought to overcome several of the serious challenges that cloud intelligence is currently running into as mentioned below [5][6][41]:

- With edge computing, there is no longer a requirement to retain data in the cloud; instead, only a small amount of data is transferred to the cloud for processing [42].
- With Edge AI, the content will in fact be examined at the network edge, greatly reducing the risk of identity theft.
- Data is a valuable asset of an enterprise and nobody wants it to leave its boundaries. Instead of transmitting raw data to the cloud, Edge AI now starts assessments locally.
- Due to the ability to perform interpretation in fractions of a second, Edge AI sensors are able to overcome latency, a limitation in cloud technology that prevents the creation of sustainable efficient real time insights.

The following upper hands make it more appealing to adopt threat intelligence in edge devices [7].

- Intelligence: In comparison to standard programs, which can only react to inputs that the designer has anticipated, intelligent systems are often more efficient and versatile. An AI neural net, on the other hand, is trained to respond to a certain type of question rather than a particular one, even if the query indeed is innovative. Applications would be unable to process information as varied as text, oral sounds or multimedia lacking intelligence.

- Real-time perceptions: Edge technology reacts to user requirements instantly since it examines information directly rather than in a distant cloud where it is delayed by long distance interconnection.
- Cost savings: Applications require reduced bandwidth requirements as a consequence of moving computing capabilities near the edge, which significantly lowers connectivity expenditures.
- Enhanced confidentiality: Since AI could evaluate actual data all without revealing it to an individual, it dramatically enhances confidentiality for anybody whose visage, speech, medical image or other private details are to be evaluated. By keeping the data on edge devices and only transmitting the deep insights towards the cloud, Edge AI substantially improves confidentiality. Even though some of the data is transmitted for learning, user identities could still be safeguarded.
- High availability: Since data processing can be done without internet access thanks to decentralization and disconnected functionalities of edge AI. As a consequence, manufacturing intelligent systems that are mission critical are more available and reliable.

Most training and learning algorithms work well with datasets that contain a few couples of features or rows. On the other hand, a disorganized dataset, like the one inferred from IoT systems (significant growth in the number of intelligent sensors, like android smart phones, smart watches, intelligent home gadgets and other connected devices produces massive amounts of information from the real world), holds so many aspects that make this strategy ineffective. Deep learning technique's efficiency to examine massive volumes of features whilst engaging with disorganized input makes it extremely important. In order to facilitate the identification, categorization and forecasts of future actions, machine learning models are frequently constructed from the acquired data. It is frequently unfeasible to transport all the data to a single centralized location point because of limitations in connectivity, memory space and security considerations. The challenge of a learning set of parameters from data dispersed over several edge nodes is examined in this work without the need to transport unprocessed data to a centralized location. Other technological and paradigm changes in edge computing have either explicitly or implicitly impacted the acceptance and evolution of federated learning frameworks.

B. Contributions of this work

The work has produced the following outcomes:

- We demonstrated an advanced threat intelligence framework at edge nodes, including the rationale behind the transition to adaptive edge intelligence. The threat intelligence provided by the prediction model (Federated Forest) and the relevant advice given by the defensive strategies serve as the data support and conceptual foundation for defence throughout the suggested strategy. The speed and resilience of threat intelligence are considerably increased by synchronizing preventative experience and implementing AI tactics.
- Furthermore, the state-of-the-art survey has been performed for obtaining deep insight into the prevailing technology advancement while over viewing the characteristics and the present-day use in brief.

The following section is laid out as follows. Section II discusses the background and state-of-the-art level survey for Edge AI threat model, Section III elaborates on the proposed framework. Finally, Section IV concludes the work.

II. BACKGROUND AND RELATED WORK

The proactive mitigation of growing cyber attacks has been made possible by the exchange of Cyber Threat Intelligence (CTI), a fresh weapon in the armoury cyber warriors. Researchers and practitioners now face significant complications as a result of automating CTI transmission and even the most fundamental consumption [8]. Cyber Threat Intelligence (CTI) is a field of cyberspace that emphasizes gathering and analyzing data regarding existing and potential assaults that pose a risk to an organization’s security or the security of its valuables. It is organized into four different categories to enable the consumption of threat intelligence. Particularly, they are technical threat intelligence, operational threat intelligence, tactical threat intelligence, and strategic threat intelligence. Regarding information gathering, representations, and intelligence ingestion, these four types of threat intelligence diverge [9].

Wide information on cyber overall security, threats, the financial effects of various cyber attacks, attack patterns, and the consequences of senior-level corporate decisions are provided by strategic threat intelligence. High-level managers and the organization’s management, including the CISO and IT management, ingest this information. It aids management in identifying present cyber threats, unidentified future hazards, risk squads, and breach accountability. The information gathered offers a risk-based reading that largely concentrates on high-level concepts of hazards and their likelihood [10-12].

The development of mitigation and detection strategies by security departments using tactical threat intelligence includes changing security products with known indicators, replacing susceptible processes, etc. It aids cyber intelligence officials in comprehending how attackers are anticipated to attack the setup, detecting information leaks from the corporation, attackers’ technical skills and objectives, as well as attack methods [14]. Operational threat intelligence provides information about potential dangers to the organization as a whole. It offers factual information about security incidents and events that aids defenders in disclosing potential risks, providing deeper insight into fugitive tactics, establishing prior suspicious attacks, and conducting investigations into fraudulent attacks in a way that is significantly more cost-effective [16]. It aids cyber intelligence officials in comprehending how attackers are anticipated to attack the setup, detecting information leaks from the corporation, attackers’ technical skills and objectives, as well as attack methods. Operational threat intelligence provides information about potential dangers to the organization as a whole. It offers factual information about security incidents and events that aids defenders in disclosing potential risks, providing deeper insight into fugitive tactics, establishing prior suspicious attacks, and conducting investigations into fraudulent attacks in a way that is significantly more cost-effective.

Federated learning at Edge

Federated Learning (FL) is a machine learning paradigm in which a model is trained by a set of stakeholders working collaboratively under the control of a centralized server whilst training information is kept offline [13]. By enabling AI training at dispersed edge devices even without the requirement for information exchange, Federated Learning (FL) has evolved as a decentralized participatory AI strategy that can facilitate numerous intelligent edge computing applications. Models are consolidated at a centralized computer in FL after being trained locally. After numerous attribute or gradient accumulation upgrades, a generic solution is obtained. Contrary to decentralized machine learning, FL’s central server is not exposed to the data of the edge nodes. The way that the knowledge is dispersed among the edge nodes can either be independent and identical or non-independent and identical. Horizontal FL (HFL), Vertical FL (VFL) and Federated Transfer Learning (FTL) are the three basic forms of FL (FTL).

Preliminaries: A single unified neural net is maintained in a central server in this processing approach. The inputs being used for training the network are typically disparate and are kept independently over several edge nodes. Here, it is considered that there are many nodes $N_1, N_2, \ldots, N_n$. The confidential data set ζi is maintained on the edge node end N_i. If we presume the loss function is *f(.)*, in single convergence, node N_i evaluates the upgraded weight, at the current weight at time t: w^i_t, step size at time t: γ_t, individual data set: ζi. Then

$$w_{t+1}^{i} = w_t - \gamma_t . \frac{\delta f(w_t,\zeta_t)}{\delta w} \text{ (i=1, 2,….n)}$$

Consider the fact that this localized upgrade may execute one or more cycles. The weights transmitted by each node are collected on the server side. To update the weights for the subsequent round, the server converges all the transmitted weights using an aggregation method *A(.)*. The updated weights at time *t+1* are:

$$w_t + 1 = A(w_t^1, w_t^2, \ldots, w_t^n)$$

Typically, aggregation of the transmitted weights is performed and upgraded to the network model using an average function. The model can be duplicated among all edge nodes and local predictions can be derived as [14] necessary. It is not demanded that all nodes take part in a particular synchronization due to the heterogeneity in federated learning. The computation will only be carried out by a small number of the nodes, chosen at random.

E. Related Works

Using the block chain incentive mechanism and privacy preserving techniques, Weng et. al. [15] introduced Deep Chain, achieving confidentiality of data and computational authenticity. It is clear from the integration of distributed ledger technology and confidentiality strategies that fully distributed learning improves the shared computing ecosystem’s ability to maintain trust. Since individual nodes must regularly contact a central server while using federated computing, lowering transmission costs is a significant limitation. Hence, it is crucial to figure out how to increase communication efficiency without compromising the precision of the combined assessment. Findings stipulated that sparse matrix [16] and model compression can dramatically lower transmission costs with little to no deterioration in model correctness [17].

According to our investigation, we observed that several of the research methodologies employ their own created or emulated datasets. For instance, [18] combined an FL technique with fog computing, wherein fog network nodes worked together to detect DDoS events. The authors employ Fed Avg as the aggregation method and Gated Recursive Units (GRUs) [19] as the ML technique for this. A threshold based technique to lessen connectivity across various data centres is proposed in [20], which studies decentralized deep learning across many data centres in various global zones. The approach in [20] focuses on peer-to-peer interconnected data centres, which is distinctive from the federated learning paradigm which is not peer-to-peer, even if it is relevant to the adaption of synchronization frequency and asset concerns. Additionally, it permits data centre node asynchronism, which is not possible with federated learning.

If an IDS were to be installed on a specific network, several target devices might have traffic connected to various assaults (such as port scanning or DoS attacks), whereas other devices might simply have traffic necessary for their intended use. In recent years, there has been an increase in the context of FL enabled IDS approaches in the framework of IoT networks [21][22]. Even so, the majority of the ways that have been suggested are either based on binary classification methodologies, wherein data analysis is simply labelled as an attack or harmless [23]. The work by Bonawitz et al. is based on Tensor Flow [24]. The main contribution of their work is the provision of an established systems approach for the deployment of federated applications by developers. They address several crucial issues, including Device accessibility, Resource management and durability. In order to increase the poisoning attack stealth, Bhagoji et al. [25] proposed a better cope strategic plan in federated learning that estimates the local upgrades of the superficial participants, attempting to make the visual justifications of model judgments impossible to distinguish among relatively harmless and attack models. Bagdasaryan et al. [26] concentrated on the backdoor attack, in which an adversarial agent can leverage the model positioning technique to insert hidden vulnerabilities into the federated model.

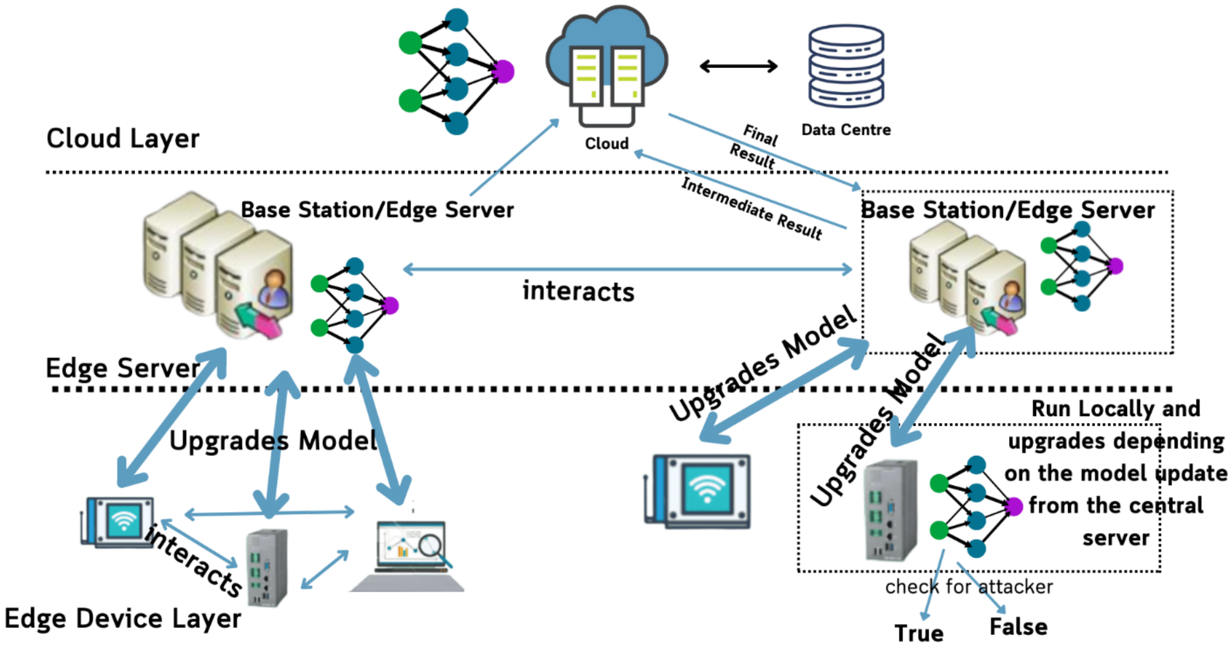


Figure 2 Overall Architecture- Integrated Threat Intelligence in Edge

III. PROPOSED FL INTEGRATED EDGE AI THREAT INTELLIGENCE FRAMEWORK

Edge computing reduces privacy and security hazards by enabling information to be evaluated close to the point of origin, possibly by a locally trustworthy edge server. This avoids using the public network. Integrating vast, complex DNNs enabling real time situations on edge endpoints is still challenging because of limited resources (such as the battery, computing devices and storage). Therefore, it makes sense to think about outsourcing DNN data processing from end systems to much more prominent nodes, like edge device servers or the central cloud. The overall framework of integrating threat intelligence in edge is represented in Figure 2. With intelligence integrated privacy protection client entity in each edge node and control unit on the edge network server, the system combines data and network intelligence services. Three phases, including Data and Behaviour Auditing, Training and Predicting, make up the FL execution's phase structure. At each stage of FL execution, the model encounters a series of privacy and security issues.

A. Data and Behaviour Auditing

Each edge node data is transparent and accessible in FL. Individual nodes have complete control over their data. It is challenging to audit the accuracy of the data and the previous conduct of all local nodes due to this restriction. Consequently, a malevolent node has the ability to quietly alter the training data in order to affect the ultimate global model. Furthermore, problems with data quality, like insufficient, unclear and unlabeled data, might arise while the collection of data, transportation and interpretation. These could have a substantial effect on the use of data in decision making [27].

B. Training Phase

Edge nodes work with the central server to update the global model throughout each round of the training process, which lasts until a particular number of iterations have been completed or a predetermined level of accuracy has been attained. The primary actions in each training round include [28][29]:

- A selection of nodes is chosen by the central server. Different factors can be taken into account for this purpose; for instance, in an Internet of Things (IoT) environment, machine's computational resources might be used to choose the trustworthy edge nodes to take part in the training session.
- The central server sends to the chosen nodes the global model's parameters and weights.
- By employing Stochastic Gradient Descent (SGD) [30] and unique local data in a training process, the various nodes modify the parameters and weights of the global model. In the case of an IDS system, it is made to carry out the training utilizing each node's local internet traffic. The learning rate in this situation reflects the local training cycles performed by a node using its private data set prior to upgrading the global model.
- The nodes then submit to the central server the revised model's parameters and weights. The server combines all the parameters and weights into an updated global model, which will be utilized in the subsequent training phase, depending on the aggregation technique being used. A round is a name given to the process in which nodes train their models, modify the global model and transmit the results to the centralized server for aggregation. A round is a name given to the process in which clients train their models, update the global model and send the results to the server for aggregation. There are other different algorithms that can be taken

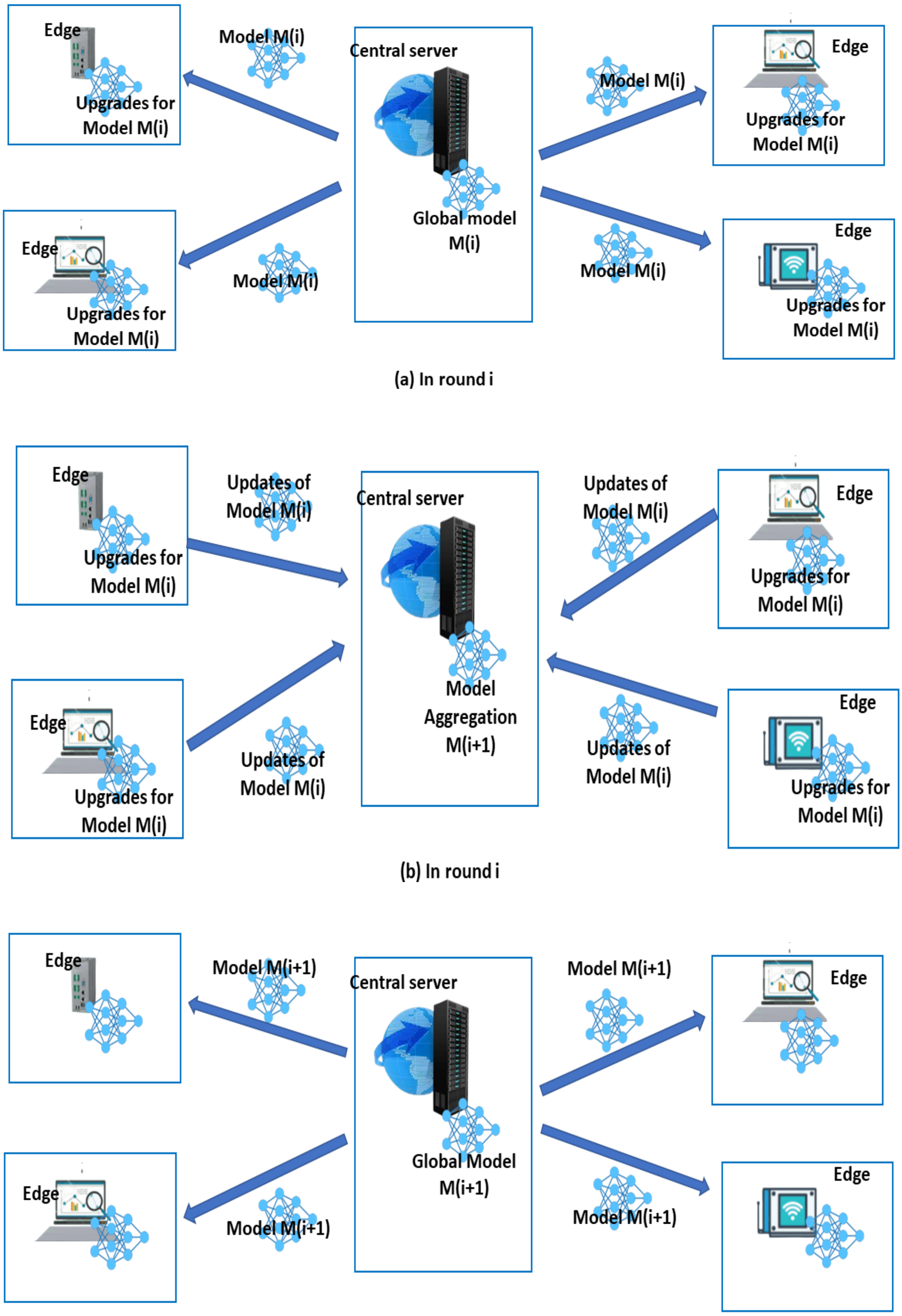
into consideration for this procedure, including Fed Prox [31] and the more current Fed+ [32] of which Fed Avg is the most popular aggregation method.

C. Prediction Phase

The robustness of a learning system under attack should be investigated by a classifier because machine learning techniques are not intrinsically adversary aware. Because of very slow pace of currently offered works based on cryptography for privacy preserving ML techniques, [33] first demonstrated that RF could be naturally applicable in a fully distributed architecture, and then developed protocols for RF to empower overall and productive distributed privacy preserving data analysis.

1) Random Forest (RF): The ensemble supervised machine learning method known as Random Forest (RF) is frequently used for classification and regression problems [34]. The decision tree is typically used by RF as the base classifier and many decision trees are generated to produce predictions [24], where the randomness is offered in 2 distinct ways: bagging technique and random selection of input characteristics. It is built on the idea of ensemble methods, which is a technique of integrating various classifiers to address difficult issues and enhance model performance. For an improved Random forest classifier, there should be some actual values in the dataset for the dataset's feature variable to predict true outcomes rather than a speculated result and each tree's predictions must have extremely low correlations. Random Forest works as two phases. First, N decision trees are combined to generate the random forest and then predictions are made for each tree that was produced in the first phase. The work flow can be enumerated as:

- Pick K data points at random from the training set.
- Construct the decision trees linked to the chosen datasets (Subsets).
- Select N for the size of the decision trees that are intended to be constructed.
- Find each decision tree's recommendations for any latest data values and then place them in the group that receives the most votes.



(c) In round i+1 and continues…

Figure 3 Model iterations

2) Federated Forest: The conventional method of prediction necessitates several rounds of interaction between the master (central server) and clients (edge nodes). The communication needs for predicting will become a significant burden when the number of nodes, the maximum tree height and the sample is big. To address this, federated forest prediction algorithm [35] is followed. Each client starts by forecasting samples using the individually stored model. Each sample enters the binary tree for the tree T_i on the i^{th} client from the parent node and eventually lands in one or more leaf nodes. Whenever the sample set passes over each node, if the network retains the partition information there, the split threshold is checked to determine whether this sample enters the left or right sub tree. The sample enters both the left and right sub trees concurrently if the model lacks split information at this node. Second, until every test falls into one or many leaf nodes, the route of the node in the tree is decided iteratively. Each leaf node of the forest T_i on the edge node will retain a bunch of observations once this operation is complete. Third, the central server computes the outcome after taking the convergence for each leaf. The samples that each node in the tree on the full tree T is then already linked to the outcomes.

Thus, each device will have a local copy of centralized machine learning program that users can utilize as necessary. The model will now progressively pick up knowledge and train on the data entered by the user, becoming periodically smarter. The training results from the machine learning app's local copy are then permitted to be transmitted from the edge devices back to the main computer. Multiple devices with a local copy of the app experience this same process simultaneously. The findings will once again be combined in the central server excluding user information. Based on the combined training results, the central cloud server now upgrades its predictive model, which is much more effective than the prior iteration. Nodes update the app with the wiser model made out of the own information, and the developer upgrades the model to a revamped model. The model iteration is represented in Figure 3.

D. Results and Discussion

The proposed architecture is validated in the real time cloud deployed in Azure single node instance. Further, a secured perimeter is created with multiple IDPS system. The entire technology stack relies on Elastic Search, Logstash and Kibana for data management. Figure 4 shows the hyper parameter results used in the proposed architecture. It is clear that the training, testing (Refer Figure 5) and validation score are as expected.

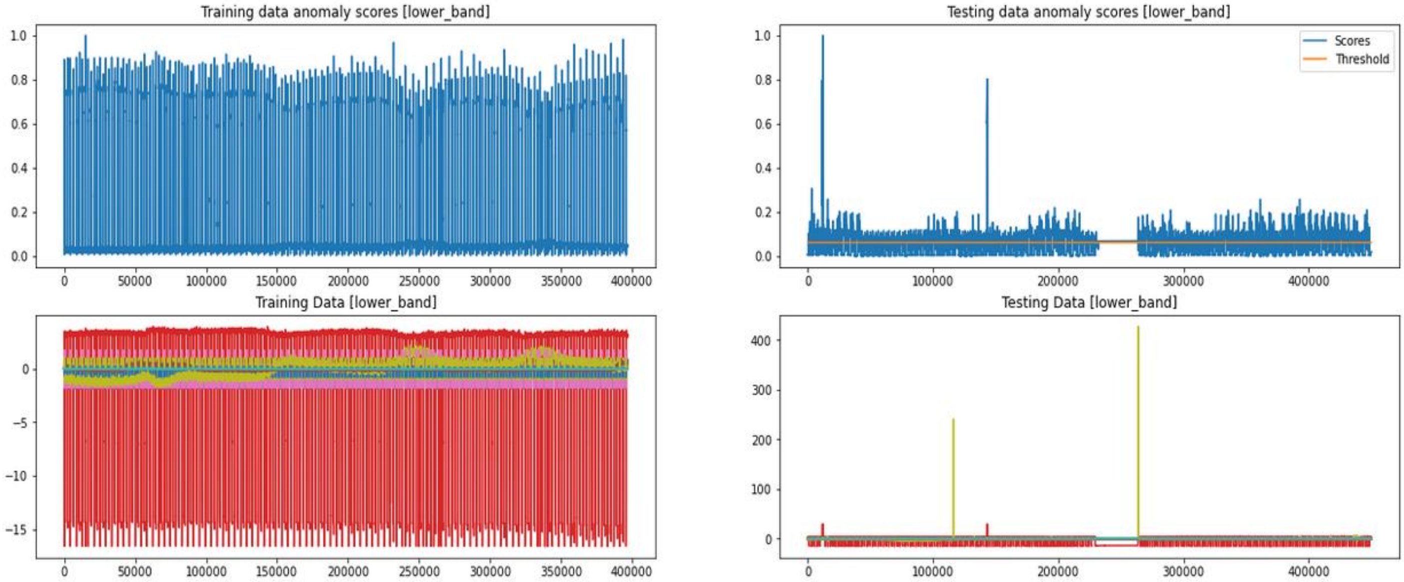


Figure 4 Hyper parameter results

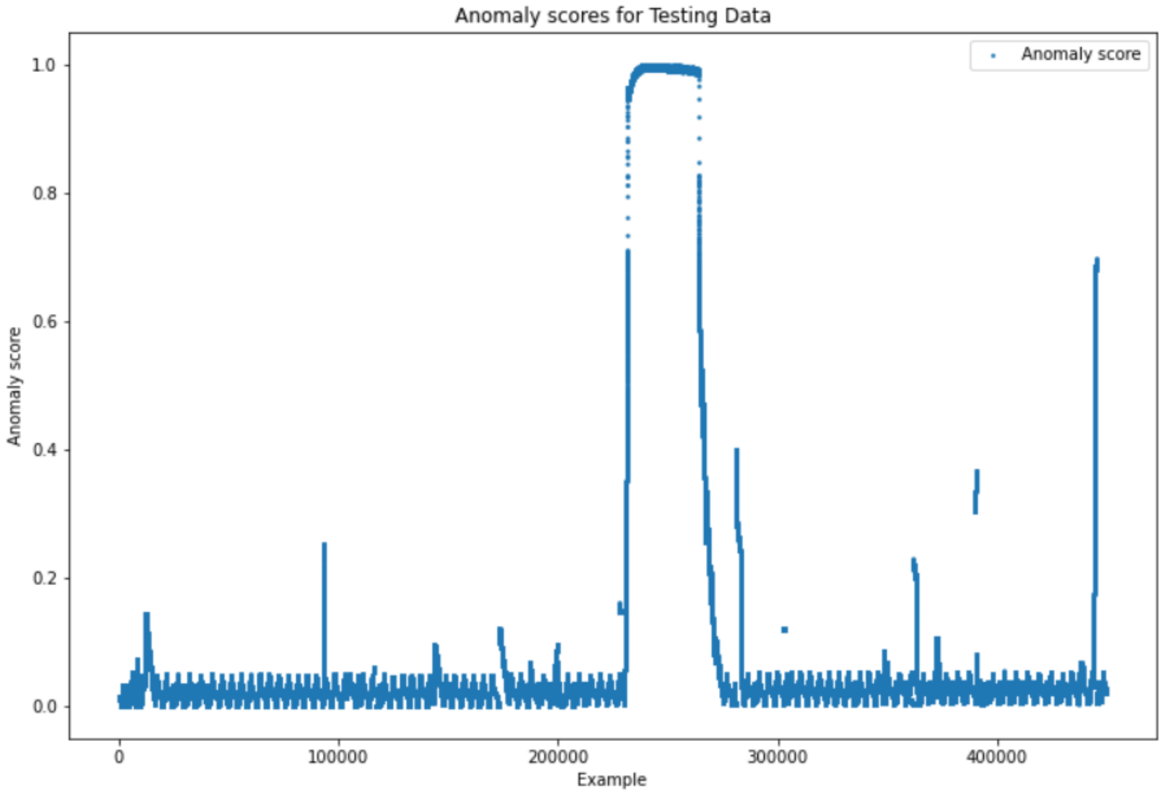


Figure 5 Threat Intelligence – Anomaly score

E. Advantages of the proposed model

The following benefits are provided by the incorporation of threat intelligence and analysis into the edge nodes:

- Accelerated Interpretation: Analyzing data flow at the edge servers produces fast results for services that use pre-trained deep learning models to generate categories or suggestions. The main cause of this is the reduction of data transmission delay between the edge system and the cloud.
- Information Locality: Information rarely leaves the edge device since it conducts the majority of data visualization and interpretation. Maintaining data location is essential for preserving secrecy in applications like healthcare monitoring, spatial localization and others.
- Data privacy: Applications like medical devices run a confidentiality risk when uploading sensitive data to the cloud. A breach of privacy in these circumstances could literally mean life or death. Thus, keeping data local helps protect end user's privacy.
- Learning in a group is simpler: Federated learning significantly reduces the amount of time and effort required for the data gathering and labelling process, as opposed to requiring the collection of one enormous data set to train a machine learning model.

IV. CONCLUSION

An intelligence security management system can benefit from the many facets of machine learning, which can improve several different components of the system. Due to their benefits over centralized information learning approaches, FL methodologies have received a great deal of interest in recent years. Throughout this work, we dispensed an outline of the current works for integrating FL in the creation of IDS methods for edge devices. We considered the potential challenges that traditional edge computing technologies may face and propounded a federated learning framework for threat intelligence in edge computing. As a matter of fact, as compared to cloud AI, edge AI that is based on deep learning is more efficient, effective, resilient and up to date. Almost always, whether in terms of cost, effectiveness or convenience, ensuring security requires some kind of compromise. While some trade-offs are expensive and have very little impact, others are very efficient and cheap. With a strong design approach, privacy must be prioritized over other design criteria and determined on an application-by-application basis. It will be necessary to look at performance, effectiveness and reliability metrics to optimize the efficiency of the aforementioned framework.

REFERENCES

[1] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman and A. Galstyan, “A survey on bias and fairness in machine learning”, *ACM Computing Surveys (CSUR)*, vol. 54, no. 6, pp. 1–35, 2021.

[2] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto and L. Zdeborova, “Machine learning and the physical´ sciences”, *Reviews of Modern Physics*, vol. 91, no. 4, p. 045002, 2019.

[3] S. Sibi Chakkaravarthy, D. Sangeetha and V. Vaidehi, “A Survey on malware analysis and mitigation techniques”, Computer Science Review, Vol. 32, 1-23, May 2019, Elsevier.

[4] B. Gill, “Gartner,” 2021. [Online]. Available: https://blogs.gartner.com/bob-gill/2021/10/05/notes-from-the-edgeis-edge-computing-all-hype/

[5] K. Cao, Y. Liu, G. Meng and Q. Sun, “An overview on edge computing research”, *IEEE Access*, vol. 8, pp. 85714–85728, 2020.

[6] E. Li, L. Zeng, Z. Zhou and X. Chen, “Edge AI: On-demand accelerating deep neural network inference via edge computing”, *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 447–457, 2020.

[7] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen and M. Chen, “In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning”, *IEEE Network*, vol. 33, no. 5, pp. 156–165, 2019.

[8] D. Arivudainambi, K.A. Varun Kumar, S. S Chakkaravarthy and P. Visu, “Malware traffic classification using principal component analysis and artificial neural network for extreme surveillance”, Computer Communications, Vol.147, November, 2019, pp.50-57, Elsevier.

[9] M. Conti, T. Dargahi and A. Dehghantanha, “Cyber threat intelligence: challenges and opportunities”, *Cyber Threat Intelligence*, pp. 1–6, 2018.

[10] T. D. Wagner, K. Mahbub, E. Palomar and A. E. Abdallah, “Cyber threat intelligence sharing: Survey and research directions”, *Computers & Security*, vol. 87, p. 101589, 2019.

[11] P. Ranaweera, A. D. Jurcut and M. Liyanage, “Survey on multi-access edge computing security and privacy”, *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 1078–1124, 2021.

[12] K. Cao, Y. Liu, G. Meng and Q. Sun, “An overview on edge computing research”, *IEEE access*, vol. 8, pp. 85714–85728, 2020.

[13] M. S. Ansari, S. H. Alsamhi, Y. Qiao, Y. Ye, and B. Lee, “Security of distributed intelligence in edge computing: Threats and countermeasures”, in *the cloud-to-thing continuum*. Palgrave Macmillan, Cham, 2020, pp. 95–122.

[14] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li and H. Vincent Poor, “Federated learning for internet of things: A comprehensive survey”, *IEEE Communications Surveys Tutorials*, vol. 23, no. 3, pp. 1622–1658, 2021.

[15] Q. Xia, W. Ye, Z. Tao, J. Wu and Q. Li, “A survey of federated learning for edge computing: Research problems and solutions”, *High-Confidence Computing*, vol. 1, no. 1, p. 100008, 2021. [Online]. Available:
https://www.sciencedirect.com/science/article/pii/S266729522100009X.

[16] J. Weng, J. Weng, J. Zhang, M. Li, Y. Zhang and W. Luo, “Deep chain: Auditable and privacy-preserving deep learning with block chain-based incentive”, *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 5, pp. 2438–2455, 2019.

[17] J. Konecnˇ y, H. B. McMahan, F. X. Yu, P. Richt` arik, A. T. Suresh and D. Bacon, “Federated learning: Strategies for improving communication efficiency”, *arXiv preprint arXiv:1610.05492*, 2016.

[18] S. Sibi Chakkaravarthy, V. Vaidehi and Steven Walczak, “Cyber Attacks on Healthcare Devices Using Unmanned Aerial Vehicles", Journal of Medical Systems, Vol.44, Article 29

[19] C.-Y. Chen, J. Choi, D. Brand, A. Agrawal, W. Zhang and K. Gopalakrishnan, “Adacomp: Adaptive residual gradient compression for data parallel distributed training”, in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[20] J. Li, L. Lyu, X. Liu, X. Zhang and X. Lyu, "Fleam: A federated learning empowered architecture to mitigate DDoS in industrial IoT", *IEEE Transactions on Industrial Informatics*, vol. 18, no. 6, pp. 4059–4068, 2021.

[21] R. Dey and F. M. Salem, "Gate-variants of Gated Recurrent Unit (GRU) neural networks", in *2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS)*. IEEE, 2017, pp. 1597–1600.

[22] K. Hsieh, A. Harlap, N. Vijaykumar, D. Konomis, G. R. Ganger, P. B. Gibbons and O. Mutlu, "Gaia:{Geo-Distributed} machine learning approaching {LAN} speeds", in *14th USENIX Symposium on Networked Systems Design and Implementation (NSDI 17)*, 2017, pp. 629–647.

[23] T. D. Nguyen, S. Marchal, M. Miettinen, H. Fereidooni, N. Asokan and A.-R. Sadeghi, "D¨IoT: A federated self-learning anomaly detection system for IoT", in *2019 IEEE 39th International conference on distributed computing systems (ICDCS)*. IEEE, 2019, pp. 756–767.

[24] N. A. A.-A. Al-Marri, B. S. Ciftler and M. M. Abdallah, "Federated mimic learning for privacy preserving intrusion detection", in *2020 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*. IEEE, 2020, pp. 1–6.

[25] V. Rey, P. M. S. Sanchez, A. H. Celdr an and G. Bovet, "Federated learning for malware detection in IoT devices", *Computer Networks*, vol. 204, p. 108693, 2022.

[26] S. Sibi Chakkaravarthy, D. Sangeetha and V. Vaidehi,"Intrusion Detection System to detect Wireless attacks in IEEE 802.11 networks", IET networks, July 2019, Volume 8, Issue 4, pp. 219- 232, IET.

[27] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konecn y, S. Mazzocchi, B. McMahan` *et al.*, "Towards federated learning at scale: System design", *Proceedings of Machine Learning and Systems*, vol. 1, pp. 374–388, 2019.

[28] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning", in *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, 2015, pp. 1310–1321.

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

[30] G. Jiang, W. Wang, Y. Qian and J. Liang, "A unified sample selection framework for output noise filtering: An error-bound perspective", *J. Mach. Learn. Res.*, vol. 22, pp. 18–1, 2021.

[31] Dedipyaman Das, S.Sibi Chakkaravarthy and Suresh Chandra Satapathy, "A Decentralized Open Web Cryptographic Standard", Computers and Electrical Engineering, Elsevier, Volume 99, 107751, April, 2022.

[32] B. McMahan, E. Moore, D. Ramage, S. Hampson and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data", in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273– 1282.

[33] E. M. Campos, P. F. Saura, A. Gonzalez-Vidal, J. L. Hern andez-Ramos, J. B. Bernabe, G. Baldini and A. Skarmeta, "Evaluating federated learning for intrusion detection in Internet of Things: Review and challenges", *Computer Networks*, vol. 203, p. 108661, 2022. [Online]. Available:
https://www.sciencedirect.com/science/article/pii/S1389128621005405.

[34] J. Goetz, K. Malik, D. Bui, S. Moon, H. Liu and A. Kumar, "Active federated learning", *arXiv preprint arXiv:1909.12641*, 2019.

[35] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar and V. Smith, "Federated optimization in heterogeneous networks", *Proceedings of Machine Learning and Systems*, vol. 2, pp. 429–450, 2020.

[36] S. Sibi Chakkaravarthy, D. Sangeetha, Meenalosini Vimal Cruz, V. Vaidehi and Vaidehi V, "Design of Intrusion Detection Honeypot using Social Leopard Algorithm to detect IoT ransomware attacks", IEEE Access, IEEE, vol. 8, pp. 169944-169956, 2020

[37] P. Yu, L. Wynter, and S. H. Lim, "Fed+: A family of fusion algorithms for federated learning," *arXiv preprint arXiv:2009.06303*, 2020.

[38] J. Vaidya, B. Shafiq, W. Fan, D. Mehmood and D. Lorenzi, "A random decision tree framework for privacy-preserving data mining", *IEEE transactions on dependable and secure computing*, vol. 11, no. 5, pp. 399–411, 2013.

[39] N. Zimmerman, A. A. Presto, S. P. Kumar, J. Gu, A. Hauryliuk, E. S. Robinson, A. L. Robinson and R. Subramanian, "A machine learning calibration model using random forests to improve sensor performance for lower-cost air quality monitoring", *Atmospheric Measurement Techniques*, vol. 11, no. 1, pp. 291–313, 2018.

[40] Y. Liu, Y. Liu, Z. Liu, Y. Liang, C. Meng, J. Zhang and Y. Zheng, "Federated forest", *IEEE Transactions on Big Data*, 2020.

[41] D. Sangeetha, S. Sibi Chakkaravarthy, Suresh Chandra Satapathy, Vaidehi V and Meenaloshini Vimal Cruz, "Multi Keyword Searchable Attribute Based Encryption for efficient retrieval of Health Records in Cloud", Multimedia Tools and Applications, Springer, 2021.

[42] Akshay T, S. Sibi Chakkaravarthy , D. Sangeetha, M. Venkata Rathnam and V. Vaidehi, "Role Based Policy to Maintain Privacy of Patient Health Records in Cloud", Journal of Super Computing, Vol.75, Issue 9, June 2019, pp.5866–5881, Springer.