DISTRIBUTED LEARNING METHODOLOGIES FOR AGRICULTURAL APPLICATIONS: CHALLENGES AND STRATEGIES FOR ENSURING PRIVACY AND RESILIENCE

Rudolf ERDEI [1,*], Daniela DELINSCHI [2], Emil PAȘCA [3], Laura ANDREICA [4], Oliviu MATEI [5]

Distributed Learning methods are becoming popular in agriculture as they minimise device expenses. However, they raise concerns about privacy, security and resource limits in various settings. In this paper, we study distributed learning systems' communication overhead, heterogeneity, and fault tolerance concerns and solutions, to find solutions for implementation in privacy-centric setups. Federated Learning helps with privacy, security, and heterogeneity. In agricultural applications, it can prove to be a good candidate to maximise their efficacy and efficiency.

Keywords: Distributed Systems, Distributed Learning, Machine Learning, Data Privacy, Low-Communications, Data Resilience

1. Introduction

Data is a fundamental element in today's digital world, with a rapidly growing volume and variety of it being generated every day. However, with the rise of *Big Data* comes the challenge of processing and analyzing it efficiently and effectively. This has led to the emergence of new data processing paradigms, such as *Distributed Learning* (DL), which enables the use of distributed computing resources to process large amounts of data in a scalable and efficient manner. DL has become increasingly important in recent years, especially with the rise of *Internet of Things* (IoT) applications, where massive amounts of sensor data need to be processed in real-time for timely decision-making. In this research article, we explore the importance of Data and the role of DL in extracting valuable insights from it, in a secure and privacy-preserving manner.

DL is a natural progression and mix between DL and *Distributed Systems/Distributed Computing*. Parallelization has been widely used for many years to overcome the hardware limitations imposed by costs or other factors. The

* Corresponding author
[1] Eng, Technical University of Cluj Napoca, Romania, e-mail: rudolf.erdei@campus.utcluj.ro
[2] Eng., Technical University of Cluj Napoca, Romania, e-mail: daniela.delinschi@campus.utcluj.ro
[3] Eng., Technical University of Cluj Napoca, Romania, e-mail: emil.pasca@campus.utcluj.ro
[4] Eng., Technical University of Cluj Napoca, Romania, e-mail: laura.andreica@campus.utcluj.ro
[5] Prof., Technical University of Cluj Napoca, Romania, e-mail: oliviu.matei@ieec.utcluj.ro

inclusion of *Graphical Processing Units* (GPUs), which have thousands of cores on a single chip, has also drastically improved training and exploitation execution times.

Concentrating all data on one computer or network raises unacceptable risks for both the supplier and the end-client. Thus, techniques that do not trust the communication medium are being developed, like Federated Learning (FL) which has gained popularity, improving privacy and security on platforms.

DL has opened a new direction of research and optimisations for many use-cases, some of which are less than obvious, like *agriculture*. Data may not need to be so private in this case, but other benefits of DL, like less verbose communication, can be exploited.

Past DL reviews include Briggs et al. [1], which focused on privacy, Quian et al. [2] which discuss the orchestration of model building in ML pipelines, and Li et al. [3] which discuss the challenges and methods of FL in special.

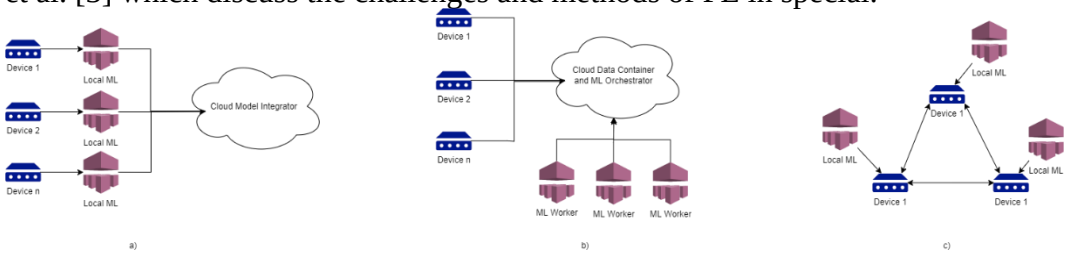

Fig. 1. Distributed Learning Paradigms

In Fig. 1, we can observe three paradigms of Distributed Learning:

a) **Federated Learning**: partial models are created within the edge nodes, while the overall model is created on the Cloud component;
b) **Classic Distributed Learning**, which uses training parallelisation: data is centralized on the Cloud and learning is done with workers, to distribute load and improve training time [4,5,6];
c) **Fog Learning**: nodes collaborate and exchange model information, contributing to an overall decentralized model. The coordination role is assumed by one of the nodes (a Leader node).

Each of these paradigms introduces improvements over the classical ML pipeline, optimizing training time, and communication within the system and solving privacy issues.

2. Research Motivation

The design of a new smart ML platform for large-scale agricultural deployment motivated this study, which enables yield and quality optimisation using ML models. Unlike personal data cases, this data was not secret, but the derived insights were sensitive. In addition, some agricultural systems use

execution hardware to control temperature, humidity, and watering. Thus, the system must act quickly to protect crops and is vulnerable to several attacks.

By integrating Artificial Intelligence (AI) into such a system, numerous opportunities arise, some noteworthy possibilities include:

- *Yield prediction*: By analysing historical data, farmers can get insights into the factors that affect crop yields and make more informed decisions;
- *Production Planning*: Yield prediction helps better production planning and distribution strategies, optimising resource use, and increasing profitability;
- *Predictive maintenance*: AI can help predict equipment maintenance, allowing for preventive measures to be taken before a breakdown occurs;
- *Precision farming*: Farmers optimise resource use, such as water, fertiliser, and pesticides. This can improve crop quality while reducing costs and their environmental impact.

3. Research Questions Formulation

DL offers a pathway to creating AI-based systems by leveraging lower-level hardware, enabling scalability, and enhancing flexibility. Consequently, several research questions must be addressed to design such a system, including:

- **RQ1**: What DL method allows for maximum flexibility, while also enabling *low communication* and improving *Data Safety*?
- **RQ2**: How can DL be used to facilitate the integration of IoT devices and other sensor data into agricultural systems, and what are the key challenges in doing so?
- **RQ3**: What are the most effective strategies for training and deploying DL models in resource-constrained environments, such as rural or remote agricultural settings?
- **RQ4**: What are the most important challenges that a DL system has to overcome, and what would be some strategies to do so?

Each of the above questions will offer context into the work needed for implementing a Distributed Learning Platform, that will operate flexibly.

The remainder of this article has 7 remaining sections, with the following structure: *Section 4* presents an overview of existing applications, *Section 5* discusses some frameworks for DL, *Section 6* analyses some algorithms for assuring data security and system resilience, *Section 7* presents some frameworks focused on system resilience, *Section 8* discuss frameworks and approaches focused on data privacy, *Section 9* discusses performance, *Section 10* provides answers to the research questions, and finally *Section 11* presents the conclusions.

4. Applications of Distributed Learning

Predd et al. [7] discuss a widely used approach in agricultural applications, which is *Wireless Sensor Networks* (WSN). Key features of the WSNs are low power usage requirements, low bandwidth as well as non-reliable/intermittent communication, so they pose a special challenge for classical DM methods, that require centralising all data. The article presents many approaches for generating and optimizing the algorithm for DL, all within the constraints of WSNs.

Deist et al. [8] propose a DL system that was implemented in five radiation oncology clinics in three European countries, for use in personalized radiation oncology treatment. The system is part of their *Decision Support System* and is implemented with a *Support Vector Machines* (SVM) algorithm.

5. Frameworks of Distributed Learning

SwitchML [9] is a framework for optimizing DL in a scalable way. However, the requirements for the bandwidth tend to be a little too high for specific applications like WSN. It features support for multiple workers and the optimization of the resulting model. One downside of this framework, however, is the need to centralize the data before it can be used in the ML pipeline. The article compares the framework with other options, with very good results.

Chamikara et al. [10] propose the DISTPAB system, which uses *Data Perturbation*-based FL, with very good results. The similarity of the resulting model with centralized learning is high, making this approach a worthwhile alternative. This approach, however, is used in conjunction with centralizing the data, albeit in a *Perturbed* way, so information leaks are still possible.

Xu et al. [11] are proposing the *GRACE* platform for DL. This platform uses multiple machines on the local network, to distribute the load on multiple computers. This platform has easy scalability, as more computers can be easily added. Still, disadvantages include the need to centralise data in the cloud, and lower platform security (as it usually runs inside its separate network).

Li [12] discusses two approaches for DL, namely *Parameter Server*, designed for efficient data communication, and *MXNet*, a library aimed at NN. Both are applications for *Distributed Computing* primarily, so privacy concerns are not taken into consideration strictly. The architecture includes a centralized training data pool, that feeds into a manager. This manager then provides chunks of data to worker nodes and creates the final model based on worker results.

A very different paradigm, namely *Fog Learning*, is proposed by Hosseinalipour et al. [13]. This paradigm distributes the effort of creating the final model to the edge nodes, intelligently, also accounting for network topology. Multiple layers of data transfer are considered, each providing dimensionality reduction, to limit upstream transfers. This approach has many benefits for large

numbers of edge nodes. For smaller systems, this approach introduces unnecessary complexity to the system, which can degrade quality but can prove useful when scaling applications.

Mai et al. [14] propose the *KungFu* system, which promises an adaptive approach to the DL paradigm. Their method centralises the user data and uses multiple systems, called workers, to compute new learning parameters.

Erdei et al. [15] propose a Security-Centric multi-paradigm platform that can handle both FL as well as Centralized Classical ML, depending on the user's option. This system, validated within an agricultural use-case, provides support for heterogeneity, enabling the system to act flexibly upon the edge nodes.

6. Algorithms for Processing and Optimisations in DL

Federated Optimization is a topic proposed by Konecny et al. [16], with the assumption that no device from an FL network has a correct overview of the behaviour in the system. Data points are distributed over a vast number of edge nodes, and they also can have a different distribution, making *Collaborative Data Mining* (CDM) harder. The proposed algorithm successfully integrates the available data sources, resulting in a quality model.

Model Averaging is a widely used and easy-to-implement method for FL. McMahan et al. [17,18] propose an efficient system for this on an application for *Long-Short Term Memory* (LSTM) networks, with some good results, yielding production-ready models. Communication is also addressed, and the system can reduce it effectively. Further work from McMahan et al. [19] has resulted in strategies for improving security inside the system, that can be applied singularly or combined. Also, in [17] they propose two *weighted average* algorithms (*FedSGD* and *FedAvg*) for computing weights or values of the model parameters and calculate weights by the number of data points that the node has.

Communication Optimization inside decentralized networks of nodes is discussed in detail by Elgabli et al. [20]. They discuss optimization possibilities like *gradient quantization* [21], *model parameter quantization* [22] and *model output exchange* [23] as alternatives, and present the *GADMM* algorithm. However, this communication optimization algorithm does not use partial models, so training is done with very verbose communication.

The problem of **Heterogeneity** in edge nodes is addressed by Hu et al. [24], which describes an algorithm (*ADSP*) which treats model updates as *commits*. These commits are forced to be sent only in specific intervals so that independent of the node training speed, model convergence would be achieved.

Damaskinos et al. [25] discuss the problem **model staleness** and **performance prediction**, proposing the *FLeet* online system for FL.

Heterogeneous devices are used, so that performance evaluation also includes hardware information. The proposed system performs better than classic FL.

7. Resilience in Distributed Learning

Resilience in distributed systems is a complex topic, that covers safety, security and the capacity to properly respond to byzantine errors within the system. As some DL systems are intrinsically more secure to data loss, research on this topic is currently limited, as more general protection strategies will work fine also in this case.

Chamikara et al. [10] proposed the *DISTPAB* system that can be very resilient to Known I/O (IO) attacks, Naive Inference (NI), and Independent Component Analysis (ICA).

Byzantine Attacks are addressed by Sohn et al. [26], which present an election-based system, mathematically proven to be resilient against byzantine attacks or failures.

Byzantine Resilience is another important topic and an important issue that most distributed systems must face. This topic is discussed in detail by Zhou et al. [27] and is also applied in their *PIRATE* system. Their approach is based on *Blockchain*, to safely store model updates. After an attack, new models can be generated by using historical updates from the Blockchain.

8. Data Privacy and Security in Distributed Learning

Data Privacy and *Data Security* are notions that gain importance and traction. Users are more aware of the data they are sharing while companies are more careful of their company secrets. This momentum can also be understood by analysing the *GDPR* law that the EU has enforced. The importance of this move will force research within the area of Data Analysis and ML.

A theoretical study of the implications and applications of data privacy is done by Balcan et al. [28], where they discuss privacy in the context of low-communicating systems. In the case of DL, especially FL, Data Privacy and Security can be a side-effect of the low-communication aim.

Data Perturbation is presented by Jayaraman et al. [35] as a solution to improving Data Privacy in ML. Multiple actors can collaboratively train a global model without exposing their own data and removing the risk of inferring it.

Froelicher et al. [29] propose a scalable system that does DL in a privacy-centric way, with good results, and propose the *SPINDLE* framework. Each node will compute a *gradient update* that will be sent to the central node and integrated into the final model. The same approach is taken by Huang et al. [30] and Jayaraman et al. [31].

Deist et al. [8] propose the *euroCAT* DL platform which is safe to use within the health industry. An *Alternating Direction Method of Multipliers* (ADMM) [32] method is used to train an SVM [33] model.

As evidenced in the presented study, Data Privacy and Data Security in DL are primarily determined by the platform's operational approach, rather than being a deliberate design decision, however, techniques such as Data Perturbation may improve the privacy of the ML system.

9. Performance of Distributed Learning Systems

Performance in DL can either be regarded as *processing speed* or *resulting model accuracy*. Both are equally important when comparing DL to Centralised Learning. When analysing *processing speed*, it is evident that the distribution process impact is marginal, compared to the training/inferring process. However, training parallelisation introduces vast benefits, with results that react 80% decrease in training time, as reported by [33].

Resulting model accuracy depends on the type of DL used. For FL, usually, the federated models are of lower accuracy [17], mainly due to the inferior capabilities of the federation process, compared to a model optimiser which has access to the entire dataset at once. Algorithms like *FedAVG* [17] aim to mitigate these aspects by weighting the parameters according to the row count they are based on, hence allowing significant potential for improvement.

10. Answers to the Research Questions

RQ1 - What DL method allows for maximum flexibility, while also enabling low communication and improving Data Safety?

Among all the DL methods, the one that stands out for Data Privacy is FL. FL operates by creating local partial models or model updates which do not carry any information about the entity that generated them. This feature makes FL an ideal choice for privacy-critical applications.

FL has several advantages. First, it reduces communication overhead and computational burden associated with large data transfers. Second, it allows the use of sensitive or private data without compromising data privacy or security. Finally, it promotes collaboration and knowledge sharing among multiple parties.

Despite many benefits, FL comes with some limitations. The quality of the local models varies based on the data quality available on each device or node. Additionally, device heterogeneity can pose a challenge when trying to aggregate the model.

RQ2 -How can distributed learning be used to facilitate the integration of IoT devices and other sensor data into agricultural systems, and what are the key challenges in doing so?

DL allows farmers to assess enormous volumes of data in real-time and make informed decisions about crop management, water consumption, and other aspects that impact yield and profitability, even in large-scale installations.

One of the key challenges is the need for robust and reliable communication infrastructure. To effectively process distributed data, it is essential to have a high-speed and low-latency network that can handle large volumes without interruptions. This can be especially challenging in rural areas, where connectivity may be limited or unreliable.

Privacy and security are particularly critical when integrating IoT devices and sensor data into agricultural systems. These devices may capture sensitive data on crops, land usage, workers' identities, and many other aspects, therefore it's crucial to secure it. Researchers are creating FL, homomorphic encryption, and differential privacy for safe and privacy-preserving distributed learning to meet this difficulty.

Finally, while creating distributed learning systems for agricultural IoT, user experience is crucial. Farmers and other stakeholders may need simple data access interfaces due to low expertise or resources.

RQ3 - What are the most effective strategies for training and deploying distributed learning models in resource-constrained environments, such as rural or remote agricultural settings?

Training and deploying DL models in resource-constrained environments, such as rural or remote agricultural settings, requires careful consideration of several factors, including limited computing resources, unreliable network connectivity, and limited access to specialized expertise.

In resource-constrained environments, it is important to **optimize the model architecture** and **training algorithms** to reduce the computational and memory requirements. Techniques such as *pruning*, *quantization*, and *distillation* can be used to reduce the size and complexity of the models while somewhat maintaining their accuracy.

Federated learning enables the training of models on local devices without uploading the raw data to a central server. It can also benefit from Transfer Learning, which involves leveraging pre-trained models and adapting them to a specific task or environment, before creating the initial model.

Collaborative Research Networks can facilitate the sharing of knowledge, expertise, and computational resources among researchers and practitioners in resource-constrained environments. This approach can be useful for developing and deploying distributed learning models in settings where specialized expertise or computational resources are limited.

RQ4 - What are the most important challenges that a Distributed Learning system must overcome, and what would be some strategies to do so?

DL systems face several challenges that need to be overcome to ensure their effectiveness and efficiency. Some of the most important challenges are:

- **Communication Overhead**: involves the exchange of large amounts of data between different devices;
- **Privacy and Security**: sharing data between different devices can pose significant privacy and security risks;
- **Heterogeneity**: Different devices in a distributed learning system may have different hardware capabilities and network connectivity;
- **Fault Tolerance:** Distributed learning systems may experience device failures or network outages, which can disrupt the learning process and result in data loss.

To overcome these challenges, several strategies can be employed, including:

- **Model Compression**: To reduce communication overhead, models can be compressed before they are transmitted between devices;
- **Encryption and Access Controls**: To ensure privacy and security, data can be encrypted, and access controls can be put in place to ensure that only authorized devices can access the data;
- **Resource Allocation**: To address heterogeneity, resources can be allocated dynamically based on the capabilities of each device;
- **Replication and Backup**: To ensure fault tolerance, models are replicated across multiple devices or backed up regularly to prevent data loss in the event of device failures or network outages.
- **Federated Learning**: this approach can help address communication overhead, privacy and security concerns, and heterogeneity challenges.

11. Conclusions

DL has come a long way, attempting to solve problems like parallelization and data privacy. Some frameworks implement useful DL algorithms into practical applications, bridging the gap to production environments.

Depending on application requirements and other constraints (like data privacy), several DL methodologies and frameworks exist, making it easier for platform developers to choose the more suitable option. Regarding Fig. 1, where we can observe the three DL paradigms, their utility can be summarized:

a) *Federated Learning*: used where the edge nodes do not trust the communication medium and/or Cloud Integrator with the data, or where bandwidth/data transfer is an issue;

b) *Classic Distributed Learning*: should be used only for non-critical/non-secret data, or within the company infrastructure. All the components trust each other and data transfer is verbose;
c) *Fog Learning*: could be used with systems that cannot have a central node. Lack of centralization will introduce a fair amount of complexity and communication overhead.

The data private options include *Federated Learning* and *Fog Learning*, as they tend not to exchange data points. Results can vary between partial models and gradient updates. For FL, several algorithms are provided that make updates easier to do.

Future research trends

Improving FL can be done in several directions, by designing more complex methodologies, incorporating and improving privacy-enhancing technologies, personalising the model to local environments, mitigating bias and improving resource efficiency.

Another research direction would be comprised of model importance and model drift metrics, where each model update is graded to be used accordingly, based on the data used for building the partial model and its current performance. This might involve quantity, quality and several other metrics like distribution, variety and so on.

Acknowledgements

This work was supported by a grant from the Romanian National Authority for Scientific Research and Innovation, CCCDI - UEFISCDI, project number EUROSTARS-4-E!4691-GenDeg 1/2024, within PNCDI IV.

This work was also supported by a grant from the Romanian National Authority for Scientific Research and Innovation, CCCDI - UEFISCDI, project number ERANET-CHISTERA-IV-TROCI 4/2024, within PNCDI IV.

R E F E R E N C E S

[1] Briggs, Christopher, Zhong Fan, and Peter Andras. "A review of privacy-preserving federated learning for the Internet-of-Things." Federated Learning Systems: Towards Next-Generation AI (2021): 21-50.
[2] YANG, R., ZOMAYA, A. Y., WANG, L., & RANJAN, R. (2019). Orchestrating development lifecycle of machine learning based IoT applications: A survey.
[3] Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). Federated learning: Challenges, methods, and future directions. IEEE signal processing magazine, 37(3), 50-60.
[4] Dean, J., Corrado, G., Monga, R., Chen, K., Devin, M., Mao, M., ... & Ng, A. (2012). Large scale distributed deep networks. Advances in neural information processing systems, 25.

[5] Chilimbi, T., Suzue, Y., Apacible, J., & Kalyanaraman, K. (2014). Project adam: Building an efficient and scalable deep learning training system. In 11th USENIX symposium on operating systems design and implementation (OSDI 14) (pp. 571-582).
[6] Yuan, B., Wolfe, C. R., Dun, C., Tang, Y., Kyrillidis, A., & Jermaine, C. M. (2019). Distributed learning of deep neural networks using independent subnet training. arXiv preprint arXiv:1910.02120.
[7] Predd, J. B., Kulkarni, S. B., & Poor, H. V. (2006). Distributed learning in wireless sensor networks. IEEE signal processing magazine, 23(4), 56-69.
[8] Deist, T. M., Jochems, A., van Soest, J., Nalbantov, G., Oberije, C., Walsh, S., ... & Lambin, P. (2017). Infrastructure and distributed learning methodology for privacy-preserving multi-centric rapid learning health care: euroCAT. Clinical and translational radiation oncology, 4, 24-31.
[9] Sapio, A., Canini, M., Ho, C. Y., Nelson, J., Kalnis, P., Kim, C., ... & Richtárik, P. (2021). Scaling distributed machine learning with In-Network aggregation. In 18th USENIX Symposium on Networked Systems Design and Implementation (NSDI 21) (pp. 785-808).
[10] Chamikara, M. A. P., Bertok, P., Khalil, I., Liu, D., & Camtepe, S. (2021). Privacy preserving distributed machine learning with federated learning. Computer Communications, 171, 112-125.
[11] Xu, H., Ho, C. Y., Abdelmoniem, A. M., Dutta, A., Bergou, E. H., Karatsenidis, K., ... & Kalnis, P. (2021, July). Grace: A compressed communication framework for distributed machine learning. In 2021 IEEE 41st international conference on distributed computing systems (ICDCS) (pp. 561-572). IEEE.
[12] Li, M. (2017). Scaling distributed machine learning with system and algorithm co-design. Santa Clara, CA, USA: Intel.
[13] Hosseinalipour, S., Brinton, C. G., Aggarwal, V., Dai, H., & Chiang, M. (2020). From federated to fog learning: Distributed machine learning over heterogeneous wireless networks. IEEE Communications Magazine, 58(12), 41-47.
[14] Mai, L., Li, G., Wagenländer, M., Fertakis, K., Brabete, A. O., & Pietzuch, P. (2020). {KungFu: Making training in distributed machine learning adaptive. In 14th USENIX Symposium on Operating Systems Design and Implementation (OSDI 20) (pp. 937-954).
[15] Erdei, R., Delinschi, D., & Matei, O. (2022, September). Security Centric Scalable Architecture for Distributed Learning and Knowledge Preservation. In International Workshop on Soft Computing Models in Industrial and Environmental Applications (pp. 655-665). Cham: Springer Nature Switzerland.
[16] Konečný, J., McMahan, H. B., Ramage, D., & Richtárik, P. (2016). Federated optimization: Distributed machine learning for on-device intelligence. arXiv preprint arXiv:1610.02527.
[17] McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017, April). Communication-efficient learning of deep networks from decentralized data. In Artificial intelligence and statistics (pp. 1273-1282). PMLR.
[18] McMahan, H. B., Yu, F. X., Richtarik, P., Suresh, A. T., & Bacon, D. (2016, December). Federated learning: Strategies for improving communication efficiency. In Proceedings of the 29th Conference on Neural Information Processing Systems (NIPS), Barcelona, Spain (pp. 5-10).
[19] Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., ... & Seth, K. (2016). Practical secure aggregation for federated learning on user-held data. arXiv preprint arXiv:1611.04482.
[20] Elgabli, A., Park, J., Bedi, A. S., Bennis, M., & Aggarwal, V. (2020). GADMM: Fast and communication efficient framework for distributed machine learning. Journal of Machine Learning Research, 21(76), 1-39.

[21] Suresh, A. T., Felix, X. Y., Kumar, S., & McMahan, H. B. (2017, July). Distributed mean estimation with limited communication. In International conference on machine learning (pp. 3329-3337). PMLR.
[22] Zhu, S., Hong, M., & Chen, B. (2016, March). Quantized consensus ADMM for multi-agent distributed optimization. In 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 4134-4138). IEEE.
[23] Jeong, E., Oh, S., Kim, H., Park, J., Bennis, M., & Kim, S. L. (2018). Communication-efficient on-device machine learning: Federated distillation and augmentation under non-iid private data. arXiv preprint arXiv:1811.11479.
[24] Hu, H., Wang, D., & Wu, C. (2020, April). Distributed machine learning through heterogeneous edge systems. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 34, No. 05, pp. 7179-7186).
[25] Damaskinos, G., Guerraoui, R., Kermarrec, A. M., Nitu, V., Patra, R., & Taiani, F. (2022). Fleet: Online federated learning via staleness awareness and performance prediction. ACM Transactions on Intelligent Systems and Technology (TIST), 13(5), 1-30.
[26] Sohn, J. Y., Han, D. J., Choi, B., & Moon, J. (2020). Election coding for distributed learning: Protecting signsgd against byzantine attacks. Advances in Neural Information Processing Systems, 33, 14615-14625.
[27] Zhou, S., Huang, H., Chen, W., Zhou, P., Zheng, Z., & Guo, S. (2020). Pirate: A blockchain-based secure framework of distributed machine learning in 5g networks. IEEE Network, 34(6), 84-91.
[28] Balcan, M. F., Blum, A., Fine, S., & Mansour, Y. (2012, June). Distributed learning, communication complexity and privacy. In Conference on Learning Theory (pp. 26-1). JMLR Workshop and Conference Proceedings.
[29] Froelicher, D., Troncoso-Pastoriza, J. R., Pyrgelis, A., Sav, S., Sousa, J. S., Bossuat, J. P., & Hubaux, J. P. (2020). Scalable privacy-preserving distributed learning. Proceedings on Privacy Enhancing Technologies 2021; 2021(2): 323–347.
[30] Huang, Z., Hu, R., Guo, Y., Chan-Tin, E., & Gong, Y. (2019). DP-ADMM: ADMM-based distributed learning with differential privacy. IEEE Transactions on Information Forensics and Security, 15, 1002-1012.
[31] Jayaraman, B., Wang, L., Evans, D., & Gu, Q. (2018). Distributed learning without distress: Privacy-preserving empirical risk minimization. Advances in Neural Information Processing Systems, 31.
[32] Wei, E., & Ozdaglar, A. (2012, December). Distributed alternating direction method of multipliers. In 2012 IEEE 51st IEEE Conference on Decision and Control (CDC) (pp. 5445-5450). IEEE.
[33] Ye, G. B., Chen, Y., & Xie, X. (2011, June). Efficient variable selection in support vector machines via the alternating direction method of multipliers. In Proceedings of the fourteenth international conference on artificial intelligence and statistics (pp. 832-840). JMLR Workshop and Conference Proceedings.
[34] Verbraeken, J., Wolting, M., Katzy, J., Kloppenburg, J., Verbelen, T., & Rellermeyer, J. S. (2020). A survey on distributed machine learning. Acm computing surveys (csur), 53(2), 1-33.
[35] Jayaraman, B., Wang, L., Evans, D., & Gu, Q. (2018). Distributed learning without distress: Privacy-preserving empirical risk minimization. Advances in Neural Information Processing Systems, 31.