

Article

Not peer-reviewed version

Architecture for Edge-Based Predictive Maintenance of Machines Using Federated Learning and Multi Sensor Platforms

[Gustavo Laydner de Melo Rosa](#)*, [Praveen Mohanram](#)*, [Andre Gilerson](#)*, [Robert Heinrich Schmitt](#)*

Posted Date: 23 May 2023

doi: 10.20944/preprints202305.1563.v1

Keywords: artificial intelligence, federated learning, predictive maintenance, 5G, smart sensors, manufacturing, data security, data silos



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

Architecture for Edge-Based Predictive Maintenance of Machines Using Federated Learning and Multi Sensor Platforms

Gustavo Laydner de Melo Rosa ^{1,*†}, Praveen Mohanram ^{1,†}, Andre Gilerson ^{1,†} and Robert H. Schmitt ^{1,2}

¹ Fraunhofer Institute for Production Technology IPT, 52074 Aachen, Germany

² Laboratory for Machine Tools and Production Engineering (WZL), RWTH Aachen University, 52062 Aachen, Germany

* Correspondence: gustavo.laydner.de.melo.rosa@ipt.fraunhofer.de

† These authors contributed equally to this work.

Abstract: Artificial Intelligence (AI) models are expected to have a great impact in the manufacturing industry, optimizing time and resource cost by enabling applications such as predictive maintenance (PM) of production machines. A necessary condition for this is the availability of high quality data collected as close as possible to the process in question. With the advent of 5G equipped multi sensor platforms (MSPs), high sampling rate data can be collected and transmitted for processing in real time. This poses a data security challenge, since this data may give valuable insight into confidential business information of companies. Federated learning (FL) enables the training of AI models with data from multiple sources without it leaving the shop floor, by utilizing distributed computing resources available on premise. This paper introduces an architecture of FL based on data collected from 5G MSPs for enabling PM in industrial environments and discusses its potential benefits and challenges.

Keywords: artificial intelligence; federated learning; predictive maintenance; 5G; smart sensors; manufacturing; data security; data silos

1. Introduction

The current progress of the Industry 4.0 revolution, generates an unprecedented amount of data from industrial processes. According to [1], data came to be measured by Terabytes in 2005, Petabytes in 2010, Exabytes in 2015 and Zettabytes in 2020, meaning roughly a billion times more data within these 15 years. New advances in Artificial Intelligence (AI) promise to make beneficial use of this data, by training models for applications in e.g., quality control, processes parameter optimisation and process forecasting [2]. One important use case is PM according to [3], since the condition of the manufacturing equipment plays a vital role in the company's competitiveness. Finding the correct schedule is difficult however, since maintenance always introduces downtime and cost [4]. "Perfect" maintenance approaches, where machine is restored to a nearly brand new state, are often expensive and time consuming [22]. On the other hand, "minimal" maintenance approaches, despite cheaper and faster, do not decrease the overall failure rate of the machine and can represent risk to personal safety, product quality and machine (i.e. production) running time [22]. PM assisted by AI models enables businesses to find the optimal maintenance schedule, by predicting when and which equipment require maintenance [5].

The performance of an AI model is highly depended on the quality of data used during training [6]. It is, therefore, necessary to equip the machinery with sensor devices, that are placed as physically close as possible to the manufacturing process. Sensors are normally attached to mobile, or rotating parts, therefore the use of wireless technologies in MSPs is often required [23]. Previously, this would prohibit high frequency measurements, because wireless technologies were not capable of transmitting the large amounts of data with the necessary low latency. The introduction of 5G technology in

industrial applications however, allows for far greater data rates with lower latency, enabling the measurement of e.g. vibrations or acoustic emission [24]. Training AI models requires a large amount of computing resources [7]. Cloud computing currently works as a catalyst for development and deployment of scalable Internet of Things (IoT) applications, since it provides a platform for storage, computation and communication of edge-generated data [25]. Businesses are then required to store their data on third party servers and transmit them over the public network. This introduces additional challenges, because public infrastructure is not capable of supporting mass-scale, high-frequency communication [9], and moving and storing data off premise for cloud computing increases the risk of data breaches [10].

Federated learning (FL) was first introduced in 2016 [14] for mobile applications in the private sector and since then has been considered the promising AI technology in terms of privacy [30]. For its capacity of utilising the distributed computing resources while communicating machine learning (ML), i.e. deep learning, pipelines through these computing resources, data does not have to move off premise [31]. Additionally, it enables novel ML models to be trained on a large amount of data from multiple businesses without them having to share it with third parties, because it is processed by each business separately. In multiple areas FL applications have been discussed. One of its most promising applications is its fusion with industrial IoT [26]. In the automotive industry, FL-enabled IoT can check for anomalies in production [27], intelligent transportation systems [28] can be developed and can provide optimal IoT-device scheduling regarding energy efficiency [29].

Finally, deploying ML models on resource-constrained end-devices poses multiple challenges [32]. Those devices are mostly powered by microcontrollers. Although microcontroller performance has made impressive strides, their computational resources and storage capacities are still limited [33]. Additionally, microcontroller-powered devices are often run without any operating system, requiring specialised bare metal software. Their main benefits however, are low latency and low power consumption [34]. It is therefore desirable in some applications that ML or data pre-processing models (e.g. self-calibration) are deployed on those end-devices, because data can either be processed immediately at the source without any added communications delay [35].

This paper presents a novel architecture utilising FL with 5G-enabled MSPs for PM, that was developed as part of the European project Celtic-Next: AI-NET-ANIARA. Subsection 1.1 and 1.2 presents a concise background on FL and PM. Section 2 introduces the infrastructure, including the 5G MSPs. Section 3 presents the developed architecture of FL-enabled MSPs for PM of machines. Section 4 concludes with a discussion on the benefits and the potential challenges of this setup.

Prior to proceeding to the subsequent section, it is imperative to provide one concise explanation of the FL and PM techniques. It is suggested that those familiar with these techniques may skip this exposition and proceed directly to section 2.

1.1. Federated learning

From the technical point of view, FL is a decentralized ML technique that delivers ML models without centralization of data [14]. In other words, instead of centralizing data in a single location for training, FL is able to train ML models by transmitting the model and training instructions to the scattered edge infrastructures where data is located. The main objective of this technique is to eliminate the necessity of sharing (sensible) data for developing ML models with neither data parties nor cloud infrastructure providers [11]. Therefore, this technique can significantly increase data security and represents one of the state of the art techniques for privacy-preserving ML.

The training process of FL is different from a conventional ML training. Since data is not centralized in a central server, the ML model is the one that must be transmitted to the clients. Once the clients are connected to a central server, one round of the FL training follows these four steps:

1. A central server sends the initial version of a ML model to participating clients. An example ML model: a neural network with its architecture defining the amount of nodes, activation functions, amount of hidden-layers, etc.

2. Each client fits the ML model to its own local data for a predefined number of times (i.e., epochs). In this step, fitting means to that the ML model recalculates its parameters aiming to minimize the difference between its predicted outputs and the true labels using a loss function (e.g., cross-entropy) and an optimization function (e.g., stochastic gradient descent).
3. Each client sends back only the updated model parameters (not the actual data) and the central server keeps track of each client's response, awaiting for the necessary quorum.
4. Once the quorum is achieved, the central server aggregates the updated parameters from each client using a fusion algorithm and uses this aggregated updated parameters to improve the ML model.

A round (or communication round) refers to a single iteration of the aforementioned four steps of the FL training process. The entire FL training process can be composed of multiple rounds and each one with multiple epochs per client.

1.2. Predictive Maintenance

An overview of the maintenance concept evolution in [12] presents that in the late 1940s the focus was on fixing equipment after it broke down. This approach, known as corrective maintenance (or breakdown maintenance), is often reactive and costly, as it involves repairing equipment without planned downtime. The primary goal is to get the equipment up and running again as quickly as possible to minimize equipment downtime [45]. As industrialization progressed, equipment downtime became more expensive and problematic, making corrective maintenance less practical. In response, maintenance practices evolved to focus on preventing breakdowns from occurring in the first place. This approach, known as preventive maintenance, involves regularly inspecting and maintaining equipment to identify and fix potential problems before they lead to breakdowns [46]. Preventive maintenance can be scheduled based on time, usage, or a combination of the two. Although combinations of corrective and preventive maintenance has been studied [43], they are still considered economically inefficient approaches, since production equipment would be stopped to constantly exchange parts even if there still is equipment remaining useful life (RUL) left. In recent years, technological advancements have enabled maintenance practices that early predict failures [42]. These practices have become meaningful and imperative to industrial environments [44].

PM can be described as a maintenance scheduling-support technique that is based on monitoring the current operating condition of the equipment according to [16]. That is, according to the actual operating status, the equipment's RUL can be forecasted. In general, RUL forecasting can be performed by a handful of algorithms, but essentially the ML-based algorithms have shown significant performance advantage [47]. In PM, ML-based RUL forecasting is trained on historical data acquired directly on the operating status. Integrated sensors in the various areas gather critical data on the operating status can be used to sense failure precursors [48]. As soon as enough data is available, a ML model is trained, extracting information out of this database. Based on this, maintenance scheduling-support is performed with information on the exact component and exact time, when maintenance is required. This is highly beneficial for industries for its cost (i.e. maintenance costs and production costs) reduction potential. Furthermore, it enhances the efficiency by utilizing the machinery to the full intended life time.

Training ML models for enabling PM applications with good performance isn't yet trivial. A whole digital infrastructure - networked sensors and machines, communication protocols, storage and computational resources - is required, before these models can be trained. A common obstacle most companies approaching PM face is the necessity to store this enormous amount of data in a centralized location for ML training [13]. Although the actual approach of using cloud solutions as an infrastructure provider for enabling data storage and ML training, cloud environments have more than once failed to provide essential data security assurances [17]. The lack of security of cloud services leads cloud service contractors to be held accountable for violations of the General Data Protection Regulation (GDPR) for data leaks. According to the GDPR [19], severe violations imply a fine of up

to 20 million euros or up to 4 % of the company's total global turnover of the preceding fiscal year, whichever is higher.

According to [18], efficient bearing failure detection through vibration data of a rotating equipment and an autoencoder-based FL approach could be achieved, while efficiently using network capacity. Therefore, this paper assumes high potential of utilizing operational data from industrial equipment for supporting maintenance scheduling through a PM application. From our perspective the RUL forecasting of rotatory equipment bearing requires a higher degree of data-privacy (and therefore an FL approach [52]), since confidential process configurations can be derived from the operational data acquired from the equipment (e.g., current, vibration, and temperature) through model inversion [49], membership inference [50] and other data-privacy attacks [51]. These threats represent a risk, specially for industrials, because unauthorized data access not only can allow competitive parties to reverse engineer equipment parameters, processes and products [49] but can also trigger legal actions against businesses for violating such data security regulations (e.g. GDPR) [52].

2. Materials and methods

2.1. Materials

The architecture system described was developed using an open-source framework, namely IBM Federated Learning (IBMFL) [20], as core module for the FL platform. From the hardware side, the MSP [21] described is developed using the STM32H7 series microcontroller. The programming language is Embedded C using the STM32 cube IDE. The 5G modules used are the commercially available Quectel RM500Q modules. Additionally, an NVIDIA Jetson Xavier NX is coupled to the MSP, providing GPU resources, storage capacity and remote access through tunneling services. The (edge) cloud infrastructure, i.e. Fraunhofer Edge Cloud (FEC), is implemented within Fraunhofer IPT's digital infrastructure and provides storing, computing and communication resources, as well as management and scalability features. Within the FEC infrastructure, a cluster with four virtual machines and VPN access is utilized.

The FEC is a cloud computing platform designed to provide computing resources and services for Internet of Things (IoT) devices and applications at the network edge, which allows faster data processing, lower latency, and reduced bandwidth consumption. The FEC is based on a distributed architecture, where computing resources are located closer to the sensors that generate data. This allows for processing of large amounts of data in real-time and enables real-time decision-making capabilities for IoT applications. The FEC platform provides a range of services, including data analytics, ML, security, and resource management.

The FL server, that performs aggregation and hosts the web application (mentioned in section 3.2), is deployed in a cluster of virtual machines inside the FEC. External parties or clients can access the web application and the FL services through a private VPN access to the cluster. The resources, that are available for aggregation services, can be scaled and managed through a resource management service provided by the FEC alongside the implementation of load balancing algorithms of container orchestration systems.

2.2. Methods

The development of the architecture was carried out using a mixed-method approach: a desk research and a survey with stakeholders from an automotive manufacturer. The desk research is composed in two main elements: a literature review on existing FL-based industrial IoT concepts and a research of existing tools for technical development of the architecture. The survey with stakeholders from an automotive manufacturer was performed as an one-to-one interview and involved mainly the definition of requirements from the industrial domain.

The literature review was performed using the Webster & Watson's method [37]. After an initial exploratory search, a search query was defined as: ("industrial Federated Learning architecture" OR

"industrial IoT architectures" OR "microcontroller-based federated learning architecture"). Mainly Google Scholar was queried. The search retrieved 131 articles. Of those, a total of 32 were screened in detail and 5 concepts were selected.

Index	Architecture	Description
1	[39]	Industrial Internet Reference Architecture (IIRA)
2	[36]	A Multi-Agent Approach for optimizing Federated Learning in Distributed Industrial IoT
3	[38]	Industrial Federated Learning (IFL) system
4	[40]	Architecture for federated analysis and learning in the healthcare domain
5	[41]	Scalable production system for Federated Learning in the domain of mobile devices

With regards to the elements of a FL system identified in the literature review, multiple existing tools were identified. Each tool, ML framework and library were analysed and their individual technical capabilities identified. Further on, these tools were used to develop the FL platform.

The survey with stakeholders took place in August 2022 and provided end-user requirements for the architecture. These end-user requirements were recorded and can be derived in the following classes:

- Data security: data storage on-premise
- Required communication protocols: LoRa Network, 5G public network and WiFi connectivity.
- Sampling requirements: vibration sampling of minimum 200Hz
- Feedback signal: maximum feedback velocity
- Services: remote and VPN access to the MSP
- Storage capacity: storing capacity for a minimum of 6 months data
- Battery capacity: battery capacity for a minimum of 1 week MSP-operation

3. Architecture Setup

In this section, the architecture overview will be explained, alongside its components and interfaces. This architecture for PM application through embedding the FL in the MSPs is represented in the Figure 1. The envisioned multi-agent environment consists of several MSPs connected to a central on-premise server. The MSPs communicate data to the on-premise central server, where data remains stored and the local training services can be run, within the FL party servers. Further on, the FL party servers deployed on the on-premise central server are longly connected with a cloud environment, where the FL server is deployed.

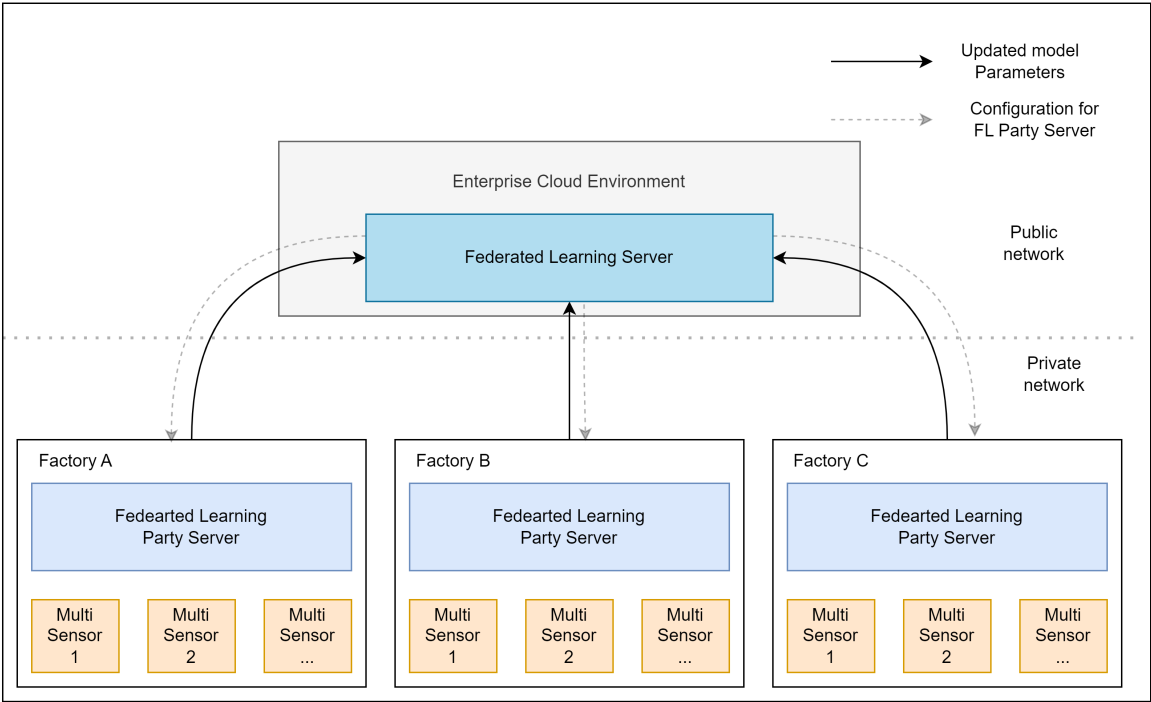


Figure 1. Multi sensor and FL platform training architecture

In this architecture, the FEC is the cloud environment hosting the FL server, the MSP is the environment hosting data acquisition services and the coupled NVIDIA Jetson Xavier NX hosts the FL party server. The later runs on premise in the manufacturing environments from vehicle paint-shop area in an automobile industry. In this scenario, the connection of the on-premise server and the cloud infrastructure is performed through the VPN access.

MSPs are deployed in bearings of air exchange systems acquiring, pre-processing, communicating and storing vibration data. These MSPs are connected to the public network through a 5G module, allowing remote connection, VPN connection to the FEC and, therefore, embedded FL services at any time.

This architecture, where high frequency vibration data is locally stored and where only ML model update parameters are transmitted, is expected to deliver less traffic load to the public network as well as enhance data privacy of parties (i.e. clients).

3.1. Multi sensor platform

Traditionally sensors are integrated into the application that requires monitoring, and the analog data from these sensors are cabled out to external data acquisition systems. Cabled sensors tend to obstruct rotating parts and impair data acquisition and the actual process. Therefore, integrating the sensor and cables into the equipment requires much planning and cost. Specific applications, such as ML-based PM, require additional sensors integrated into the equipment to extend the feature space of the dataset.

A wireless solution seems more practical for this application. The approach with a wireless solution provides another challenge: battery run-time. Since the application typically needs to run for a long time, like months, the MSP must run for the whole duration before being removed for battery change. With the increased complexity of additional sensors and data processing for demanding applications, an energy-efficient MSP is needed.

The energy-efficient MSP [8] was designed to consist of multiple sensor interfaces and complemented with hybrid communication connectivity. The MSP has a LoRa module for low-power transmission and a 5G module for enhanced connectivity with fast communication. The necessary sensors are integrated using the sensor interface, depending on the application. The core of the MSP, the microcontroller, has a dual-core processor, which can be selected on the run depending on the performance/battery run time requirements. The MSP's software achieves energy efficiency by optimizing the application with scheduled data acquisition, data processing, analysis, and data transmission through appropriate communication channels.

Furthermore, the coupled NVIDIA Jetson was setup with four *systemctl* commands for running *systemd* services. One for automatically connect and monitor internet connectivity through the 5G module. A second for starting a tunneling service, enabling remote access to the NVIDIA Jetson and the MSP. A third for connecting to the FEC through VPN. A fourth for managing the interface and data stream between the MSP and NVIDIA Jetson.

3.2. Federated Learning Platform

This technical concept mainly joined principles from the IIRA [36], IFL system design [38], the a three-layer collaborative and layered databus architectures [39]. Further more, the FL platform was developed on top of one of the most advanced open-source frameworks: the IBM Federated Learning (IBMFL) [20]. This constantly developing framework is currently in enabling multiple ML methods and fusion algorithms in federated fashion. In addition, IBMFL framework can perform state-of-the-art encryption methods, e.g., homomorphic encryption using secure multi-party protocols.

In Figure 2, the schema of the FL platform, it's components, core-services and interactions is presented.

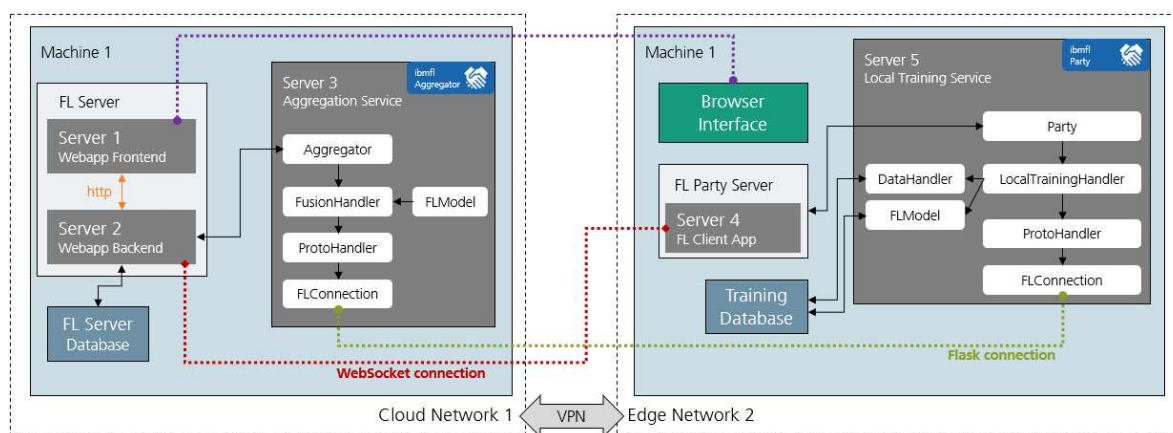


Figure 2. FL platform and cross-network interactions

The FL server is the element responsible for two main activities: the project management of decentralized projects and the execution of aggregation services.

Project management is performed through an user-friendly web application deployed in a cloud environment reachable for the project partners. It is designed for collaborative project development with data-owner partners and data scientist partners. For this activity two interfaces are foreseen: one with data-owner parties - through which local training settings can be provided - and one with the data scientist responsible for developing the ML pipelines. The web application maintains a WebSocket connection with FL party servers permitting exchange of command line input (CLI) commands between the FL server and the FL party server, what eases the interaction of data-owners with the IBMFL framework.

Aggregation services are responsible for distribution of training settings, monitoring the server-party connection and executing model fusion algorithms iteratively, the vital algorithm of

FL. Aggregation services are executed when a new training round is triggered in the project and these services first task is to transmit a non-transitory computer-readable medium that stores instructions about training settings to the respective FL party servers involved in the project. These decentralized FL party servers then execute a local training service and register automatically through the CLI commands to the aggregation service according to these instructions. After configuring the local training service with the aggregation service, a continuous flask connection allows mutual model parameters exchange as well as IBMFL commands (e.g. TRAIN, SAVE, SYNC, EVAL).

The FL party servers are responsible for three main activities: the configuration of local training services with respective aggregation services, the execution of local training services and the iterative exchange of the partial training results for updated global parameters with the aggregation services.

The configurations arrive at the FL party server immediately after triggering of aggregation service in the web application. These settings define CLI commands for the local training service register to the aggregation service, the ML model, the data handling pipelines, the local and global hyperparameters of each single decentralized project.

After executed, the local training service is responsible for the processing of local data during model training and exchanging partial training results with the aggregation service. This processing of local data takes place on the party infrastructure, where the data is safely stored and previously set on the project management web application by the data-owner. The processing may also be set by the data-owner to a chosen computational resource available in his network, which must have been also previously defined in the project management web application. Conclusively, settings for the location of the data and available computational resource are flexible for data-owner partners to designate.

The iterative exchange of partial training results occurs through the flask connection between the the local training service and aggregation service in the FL server after the training phase is triggered. The partial training results of all local training services arrive at the aggregation service one-by-one as the local processing of the iteration is completed. The lack of parties synchronicity does not affect the quality of the global model, but does affect the total training time. Aggregation services are allowed to abandon parties for an iteration round in case of a non-responsive FL party server, ensuring the rightful use of data-owner resources involved in the project.

4. Conclusions

The adoption of the architecture setup with edge FL in MSPs network for PM offer numerous advantages to production companies. However, its successful implementation is likely to encounter several challenges that need to be overcome for seamless operation.

4.1. Adoption advantages

Cutting-edge maintenance strategy

The use of PM approaches itself represent a cost reduction potential for the manufacturing company. Through the monitoring of rotating equipment components, the maintenance procedure can be scheduled according to the component condition. On one hand, this approach supports maintenance intervention by optimizing the life time of the component, what than reduces maintenance costs. On the other hand, it supports scheduling of the intervention according to the RUL forecast of the ML model. Further on, the RUL forecast enables the intervention to adhere to production planning requirements, therefore reducing production downtime costs.

Data privacy and governance

Without the transmission of vibration data to a cloud environment, confidential data is kept safe on-premise. Nevertheless, the application of FL for training models capable of forecasting RUL of equipment components, permits to still exploit the potential of operational data without risking, both data privacy and governance [52].

Federated Learning Operations

The introduction of a FL platform enables a first step to conduction decentralized learning projects in manufacturing environments [53]. It centralizes the management of a multi-agent service (i.e. the FL services) and provides user-friendly features for managing several decentralized projects at the same time. The FL platform also simplifies the configuration of the local training services for the clients, by introducing the Websocket connection enabling third parties to configure them as well. Triggering commands to the FL core framework is also possible through the web application.

4.2. Implementation challenges

Data heterogeneity

Once decentralized data is being acquired by the MSP, the different local requirements of each data-owner party are expected to result in heterogeneous datasets. Each industrial environment presents different behaviors and they are expected to be fingerprinted in the data through its distribution, quality and sampling frequency. Training FL models ideally requires similar datasets in all parties for achieving high model performance and for that achieving comparability between heterogeneous datasets is most important. An innovative solution for overcoming this challenge is enabling personalized data preprocessing pipelines on the individual datasets through the FL platform. By this means it's expected to enable comparability between heterogeneous datasets and increase performance model prospects after training.

Network congestion

One of the main challenges of using FL is network congestion. This can occur when many edge nodes are trying to communicate with the central server at the same time, resulting in a bottleneck that slows down the training process. Additionally, if some edge nodes have limited bandwidth, they may be unable to transmit their data to the server quickly, further contributing to congestion. This can make it difficult to train the model effectively and can also lead to increased latency, which can impact the user experience. To address these challenges, FL systems need to be designed to handle network congestion and optimize the communication between edge nodes and the central server.

Network congestion can still be an issue in a data silo FL system, i.e. where the data is stored on-premise. Due to the limited computing and storage resources available to each edge node, it is likely that a central on-premise data storage system is employed.

Hardware acceleration

For an application involving AI, including FL, the platform should meet specific hardware requirements such as computational performance, graphics processing unit, memory bandwidth, data storage, etc. With the edge computation paradigm, these requirements cannot be met with the standard system on chip (SoC) or mini PCs and hence needs acceleration. These hardware accelerators can be in terms of external/onboard co-processors, GPUs, ASICs, or even FPGAs. For example, the new Nvidia Jetson series System on Chip modules has a CPU and a discrete GPU developed extensively for Edge AI applications. These modules can make use of the GPU to enhance the AI inference with GPU-accelerated models without overloading the CPU, making it more real-time and efficient.

Hardware energy consumption

One of the main challenges in developing the wireless MSP for PM and FL is power consumption. Since the MSP is typically plug-and-play, where it is mostly deployed in environments where access to a power supply is a challenge. Hence it has to run on battery for a longer time i.e. at least one week on a single charge (see 2.2). Ideally, a microcontroller with a good amount of peripherals and optimum processing power, capable of handling simple AI loads while consuming low power (typically less than

500mW to 1W) is preferred. The communication also follows with ideally using low-power technology such as LoRa, WiFi and 5G. The software architecture also plays a significant role in determining the run time. The application has to be programmed taking into the available run time, and processing requirements and accordingly scheduling the task for the application. With the local FL platform, the second challenge the MSP faces is to communicate with the aggregation server. A generalized communication protocol has to be predefined and communicated with all the MSPs so that the data between sensors and the FL platform can be communicated without data loss. This could be either sophisticated protocols such as TCP, LwM2M or simple message query protocols such as MQTT or CoAP with data redundancy feature included.

References

1. Duan, ., Da Xu, . Data Analytics in Industry 4.0: A Survey. Inf Syst Front (2021). <https://doi.org/10.1007/s10796-021-10190-0>
2. Krauß, J., Dorißen, J., Mende, H., Frye, M., Schmitt, R.H. (2019). Machine Learning and Artificial Intelligence in Production: Application Areas and Publicly Available Data Sets. In: Wulfsberg, J.P., Hintze, W., Behrens, BA. (eds) Production at the leading edge of technology. Springer Vieweg, Berlin, Heidelberg. <https://doi.org/10.1007/978-3-662-60417-5-49>
3. Cardoso, D.; Ferreira, L. Application of PredictiveMaintenance Concepts Using Artificial Intelligence Tools. Appl. Sci. 2021, 11, 18. <https://dx.doi.org/10.3390/app11010018> arXiv:2207.14529.
4. Tam, A. S. B., Chan, W. M., & Price, J. W. H. (2006). Optimal maintenance intervals for a multi-component system. Production Planning and Control, 17(8), 769-779.
5. Errandonea, I., Beltrán, S., & Arrizabalaga, S. (2020). Digital Twin for maintenance: A literature review. Computers in Industry, 123, 103316. doi:10.1016/j.compind.2020.103316
6. Budach, L., Feuerpfeil, M., Ihde, N., Nathansen, A., Noack, N., Patzlaff, H., ... & Naumann, F. (2022). The Effects of Data Quality on Machine Learning Performance. arXiv preprint
7. J. Chen and X. Ran, "Deep Learning With Edge Computing: A Review," in Proceedings of the IEEE, vol. 107, no. 8, pp. 1655-1674, Aug. 2019, doi: 10.1109/JPROC.2019.2921977.
8. S. S. Schmitt, P. Mohanram, R. Padovani, N. König, S. Jung and R. H. Schmitt, "Meeting the Requirements of Industrial Production with a Versatile Multi-Sensor Platform Based on 5G Communication," 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, 2020, pp. 1-5, doi: 10.1109/PIMRC48278.2020.9217230.
9. Deutsche IHK: Digitalisierungsumfrage 2021. Zeit für den digitalen Aufbruch. Die IHK-Umfrage zur Digitalisierung. Deutscher Industrie- und Handelskammertag e. V., 2021.
10. Sotto, L., Treacy, B., & McLellan, M. (2010). Privacy and data security risks in cloud computing.. World Communications Regulation Report, 5(2), 38.
11. Ludwig, H., & Baracaldo, N. (2022). Introduction to Federated Learning. In H. Ludwig & N. Baracaldo (Eds.), Federated Learning: A Comprehensive Overview of Methods and Applications (pp. 1–23). doi:10.1007/978-3-030-96896-0-1
12. Manzini, R., Regattieri, A., Pham, H., & Ferrari, E. (2010). Maintenance for industrial systems (pp. 409-432). London: Springer.
13. Brandner, M.; Fritz, T.: Predictive Maintenance. Basics, Strategies, Models. DLG Expert report 5/2019, 2019.
14. Konečný, J., McMahan, H. B., Ramage, D., & Richtárik, P. (2016). Federated optimization: Distributed machine learning for on-device intelligence. arXiv preprint arXiv:1610.02527.
15. Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019. Federated Machine Learning: Concept and Applications. ACM Trans. Intell. Syst. Technol. 10, 2, Article 12 (March 2019), 19 pages. <https://doi.org/10.1145/3298981>
16. Popescu, T.D., Aiordachioaie, D. & Culea-Florescu, A. Basic tools for vibration analysis with applications to predictive maintenance of rotating machines: an overview. Int J Adv Manuf Technol 118, 2883–2899 (2022). <https://doi.org/10.1007/s00170-021-07703-1>
17. Tianfield, H. (2012). Security issues in cloud computing. 2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 1082-1089.

18. Becker, S., Styp-Rekowski, K., Stoll, O. V. L., & Kao, O. (2022). Federated Learning for Autoencoder-based Condition Monitoring in the Industrial Internet of Things. arXiv preprint arXiv:2211.07619.
19. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) [2016] OJ L 119/1
20. Ludwig, H., Baracaldo, N., Thomas, G., Zhou, Y., Anwar, A., Rajamoni, S., ... Others. (2020). IBM Federated Learning: an Enterprise Framework White Paper V0. 1. ArXiv Preprint ArXiv:2007. 10987.
21. Mohanram, P.; Passarella, A.; Zattoni, E.; Padovani, R.; König, N.; Schmitt, R.H. 5G-Based Multi-Sensor Platform for Monitoring of Workpieces and Machines: Prototype Hardware Design and Firmware. *Electronics* 2022, 11, 1619. <https://doi.org/10.3390/electronics11101619>
22. Pham, H., & Wang, H. (1996). Imperfect maintenance. *European journal of operational research*, 94(3), 425-438.
23. A. A. Kumar S., K. Ovsthus and L. M. Kristensen., "An Industrial Perspective on Wireless Sensor Networks — A Survey of Requirements, Protocols, and Challenges," in *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1391-1412, Third Quarter 2014, doi: 10.1109/SURV.2014.012114.00058.
24. Attaran, M. The impact of 5G on the evolution of intelligent automation and industry digitization. *J Ambient Intell Human Comput* (2021). <https://doi.org/10.1007/s12652-020-02521-x>
25. Lee, J., Singh, J., & Azamfar, M. (2019). Industrial artificial intelligence. arXiv preprint arXiv:1908.02150.
26. Pham, Q. V., Dev, K., Maddikunta, P. K. R., Gadekallu, T. R., & Huynh-The, T. (2021). Fusion of federated learning and industrial internet of things: a survey. arXiv preprint arXiv:2101.00798.
27. A. Husakovic, E. Pfann, and M. Huemer, "Robust machine learning based acoustic classification of a material transport process," in 2018 14th Symposium on Neural Networks and Applications (NEUREL). IEEE, 2018, pp. 1–4.
28. A. Ferdowsi, U. Challita, and W. Saad, "Deep learning for reliable mobile edge analytics in intelligent transportation systems: An overview," *IEEE vehicular technology magazine*, vol. 14, no. 1, pp. 62–70, 2019.
29. Y. Cui, K. Cao and T. Wei, "Reinforcement Learning-Based Device Scheduling for Renewable Energy-Powered Federated Learning," in *IEEE Transactions on Industrial Informatics*, 2022, doi: 10.1109/TII.2022.3210008.
30. R. C. Geyer, T. Klein, and M. Nabi, "Differentially private federated learning: A client level perspective," arXiv preprint arXiv:1712.07557, 2017.
31. J. Konecny, H. B. McMahan, D. Ramage, and P. Richtarik, "Federated optimization: Distributed machine learning for on-device intelligence" arXiv preprint arXiv:1610.02527, 2016.
32. M. G. Sarwar Murshed, Christopher Murphy, Daqing Hou, Nazar Khan, Ganesh Ananthanarayanan, and Faraz Hussain. 2021. Machine Learning at the Network Edge: A Survey. *ACM Comput. Surv.* 54, 8, Article 170 (November 2022), 37 pages. <https://doi.org/10.1145/3469029>
33. S. S. Saha, S. S. Sandha and M. Srivastava, "Machine Learning for Microcontroller-Class Hardware: A Review," in *IEEE Sensors Journal*, vol. 22, no. 22, pp. 21362-21390, 15 Nov.15, 2022, doi: 10.1109/JSEN.2022.3210773.
34. A. Burrello, M. Scherer, M. Zanghieri, F. Conti and L. Benini, "A Microcontroller is All You Need: Enabling Transformer Execution on Low-Power IoT Endnodes," 2021 IEEE International Conference on Omni-Layer Intelligent Systems (COINS), Barcelona, Spain, 2021, pp. 1-6, doi: 10.1109/COINS51742.2021.9524173.
35. M. Brandalero et al., "AITIA: Embedded AI Techniques for Embedded Industrial Applications," 2020 International Conference on Omni-layer Intelligent Systems (COINS), Barcelona, Spain, 2020, pp. 5, doi: 10.1109/COINS49042.2020.9191672.
36. Zhang, W., Yang, D., Wu, W., Peng, H., Zhang, N., Zhang, H., & Shen, X. (2021). Optimizing federated learning in distributed industrial IoT: A multi-agent approach. *IEEE Journal on Selected Areas in Communications*, 39(12), 3688-3703.
37. Jane Webster and Richard Watson. 2002. Analyzing the Past to Prepare for the Future: Writing a Literature Review. *MIS Q.* 26, 2 (2002), xiii–xxiii. DOI:<https://doi.org/10.1.1.104.6570>
38. Hiessl, T., Schall, D., Kemnitz, J., & Schulte, S. (2020, July). Industrial federated learning—requirements and system design. In *Highlights in Practical Applications of Agents, Multi-Agent Systems, and Trust-worthiness. The PAAMS Collection: International Workshops of PAAMS 2020, L'Aquila, Italy, October 7–9, 2020, Proceedings* (pp. 42-53). Cham: Springer International Publishing.

39. S.-W. Lin, B. Miller, J. Durand, G. Bleakley, A. Chigani, R. Martin, B. Murphy, M. Crawford, The industrial internet of things volume g1: reference architecture, Indus. I Internet Consortium 10 (2017) 10–46.
40. Antunes, R. S., André da Costa, C., Küderle, A., Yari, I. A., & Eskofier, B. (2022). Federated learning for healthcare: Systematic review and architecture proposal. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 13(4), 1-23.
41. Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., ... & Roselander, J. (2019). Towards federated learning at scale: System design. *Proceedings of machine learning and systems*, 1, 374-388.
42. Roda, I., & Macchi, M. (2021). Maintenance concepts evolution: a comparative review towards Advanced Maintenance conceptualization. *Computers in Industry*, 133, 103531. doi:10.1016/j.compind.2021.103531
43. Stenström, C., Norrbin, P., Parida, A., & Kumar, U. (2016). Preventive and corrective maintenance – cost comparison and cost-benefit analysis. *Structure and Infrastructure Engineering*, 12(5), 603–617. doi:10.1080/15732479.2015.1032983
44. P. Henriquez, J. B. Alonso, M. A. Ferrer and C. M. Travieso, "Review of Automatic Fault Diagnosis Systems Using Audio and Vibration Signals," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 5, pp. 642-652, May 2014, doi: 10.1109/TSMCC.2013.2257752.
45. Wang, Y., Deng, C., Wu, J., Wang, Y., & Xiong, Y. (2014). A corrective maintenance scheme for engineering equipment. *Engineering Failure Analysis*, 36, 269–283. doi:10.1016/j.engfailanal.2013.10.006
46. de Faria, H., Costa, J. G. S., & Olivas, J. L. M. (2015). A review of monitoring methods for predictive maintenance of electric power transformers based on dissolved gas analysis. *Renewable and Sustainable Energy Reviews*, 46, 201–209. doi:10.1016/j.rser.2015.02.052
47. Sateesh Babu, G., Zhao, P., & Li, X. L. (2016). Deep convolutional neural network based regression approach for estimation of remaining useful life. In *Database Systems for Advanced Applications: 21st International Conference, DASFAA 2016, Dallas, TX, USA, April 16-19, 2016, Proceedings, Part I* 21 (pp. 214-228). Springer International Publishing.
48. H. M. Hashemian and W. C. Bean, "State-of-the-Art Predictive Maintenance Techniques*," in *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 10, pp. 3480-3492, Oct. 2011, doi: 10.1109/TIM.2009.2036347.
49. Oh, H., & Lee, Y. (2019, June). Exploring image reconstruction attack in deep learning computation offloading. In *The 3rd International Workshop on Deep Learning for Mobile Systems and Applications* (pp. 19-24).
50. R. Shokri, M. Stronati, C. Song and V. Shmatikov, "Membership Inference Attacks Against Machine Learning Models," 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 2017, pp. 3-18, doi: 10.1109/SP.2017.41.
51. Mireshghallah, F., Taram, M., Vepakomma, P., Singh, A., Raskar, R., & Esmailzadeh, H. (2020). Privacy in Deep Learning: A Survey. *ArXiv [Cs.LG]*. Retrieved from <http://arxiv.org/abs/2004.12254>
52. Brauneck, A., Schmalhorst, L., Kazemi Majdabadi, M.M. et al. Federated machine learning in data-protection-compliant research. *Nat Mach Intell* 5, 2–4 (2023). <https://doi.org/10.1038/s42256-022-00601-5>
53. Q. Cheng and G. Long, "Federated Learning Operations (FLOps): Challenges, Lifecycle and Approaches," 2022 International Conference on Technologies and Applications of Artificial Intelligence (TAAI), Tainan, Taiwan, 2022, pp. 12-17, doi: 10.1109/TAAI57707.2022.00012.