Secure and Robust Federated Learning for Predictive Maintenance in Optical Networks

Khouloud Abdelli[1,2], Joo Yeon Cho[1], and Stephan Pachnicke[2]

[1]ADVA Optical Networking SE, Fraunhoferstr. 9a, 82152 Munich/Martinsried, Germany
[2] Christian-Albrechts-Universität zu Kiel, Kaiserstr. 2, 24143 Kiel, Germany
{KAbdelli, JCho}@adva.com, stephan.pachnicke@tf.uni-kiel.de

Abstract

Machine learning (ML) has recently emerged as a powerful tool to enhance the proactive optical network maintenance and thereby improves network reliability and reduces unplanned downtime and maintenance costs. However, it is challenging to develop an accurate and reliable ML model for solving predictive maintenance tasks (e.g., anomaly detection, fault diagnosis, remaining useful prediction etc) mainly due to the unavailability of a sufficient amount of training data since the device failure does not occur often in optical networks. Federated learning (FL) is a promising candidate to tackle the aforementioned challenge by enabling the development of a global ML model using datasets owned by many vendors without revealing their business-confidential data. While FL greatly enhances the data privacy, it is vulnerable to various model inversion and poisoning attacks. In this paper, we propose a robust collaborative learning framework for predictive maintenance in a cross-vendor setting, whereby the defensive mechanisms to protect against the aforementioned attacks are implemented. The multi-party computation (MPC)-based secure aggregation is adopted to defend against the model inversion attacks whereas a trained autoencoder based anomaly detection model is used to recognize the model poisoning attacks launched by compromised vendors. The proposed framework is applied to the semiconductor laser degradation prediction use case. We conduct experiments on semiconductor laser reliability data obtained from different laser manufacturers under various attack scenarios to evaluate the attack defense and detection capabilities of the proposed approach. Our experiments confirm that a global ML model can be accurately built with sensitive datasets in federated learning even when a subset of vendors is compromised.

Introduction

Optical fiber networks compose the core of the telecommunication infrastructure today due to their high capacity of data transmission. Optical networks rely on fully functional hardware components that run under optimal conditions. In order to reduce the risk of unplanned network interruption and service outage, it is important to predict the degradation of hardware network components correctly using analyzing tools and techniques, by which the maintenance budget and resources are allocated efficiently and timely. Due to the great benefits in industry, the global predictive maintenance market is expected to reach more than 13 billion US dollars by 2026 (ReportLinker. (2021)).

Machine learning (ML) based techniques have emerged as efficient tools to improve the accuracy of predictive maintenance in the manufacturing industry and communication networks. An ML model is trained by the historical data of hardware failure and then the upcoming maintenance is predicted by real-time data gathered through measurement at the edge. ML techniques can be useful, if a sufficiently large, diverse, and realistic set of training data exists. Since an ML model relies so heavily on good training data, the availability of such datasets is a crucial requirement for this approach.

However, it is challenging to develop a high-precision ML model for predictive maintenance mainly due to the lack of training data. The hardware failures or maintenance events do not occur frequently so that it takes time until good and meaningful training data are collected through the network. Hence, an ML model is often trained using the accelerated aging test results (e.g., a life cycle under the extreme temperature or the over-powered condition) that are conducted by hardware manufacturers. Since the components of network equipment are usually produced by small and medium-sized companies, such an ML model is trained based on the limited amount of data that are owned by each manufacturer. This situation can be relieved, if the training dataset can be aggregated from multiple vendors and consolidated in a central location. Since collaborative learning allows to train a model on larger datasets rather than the dataset available in

a single vendor, a higher quality and more accurate ML model can be built. However, such collaboration is not straightforward in reality since vendors are not willing to share their training data with external companies. Aging test data are often company-confidential and trade secret. Moreover, sharing data with foreign companies may be prohibited by privacy protection regulations in their home countries. To overcome such data-privacy concerns, federated learning (FL) (i.e., collaborative learning) has been proposed by enabling many vendors to collaboratively train a global ML model without sharing their local private data with others. However, FL is susceptible to various attacks such as inversion model attacks aiming to compromise the data's confidentiality and poisoning attacks preventing the global model from converging and thereby adversely impacts its performance.

In this paper, we propose a secure and robust collaborative learning framework incorporating defensive mechanisms to defend against above attacks, using cross-vendor datasets for predictive maintenance in optical networks. We apply our approach to the use case of predicting the degradation of semiconductor laser devices deployed in optical networks. The experiments are performed using laser reliability data from different laser manufacturers under various attack scenarios to test the efficiency of our defensive mechanisms in protecting against attacks launched by compromised vendors.

The rest of this paper is structured as follows: Section 2 gives some background information and related work. Section 3 presents the proposed framework as well as the defending mechanisms involved in the framework. Section 4 describes the validation of the presented framework using experimental data. Conclusions are drawn in Section 5.

Background and related work

Federated Learning

Federated Learning (FL) is a framework of enabling distributed parties to work together to train machine learning models without sharing the underlying data or trusting any of the individual participants (Bonawitz, et al., 2017). FL can be used to build an ML model from various companies for the purpose of predicting the failures, repairs, or maintenance of network systems. With the FL technique, the training data is not required to be centralized, but can instead remain with the data owners. Each vendor trains an ML model on their private data and using their own hardware. These models are then aggregated by a central server (e.g., a network operator) to build a unified global model that has learned from the private data of every vendor without ever directly accessing it. Hence, confidential training data (e.g., aging test results of products) are not visible to a server, nor other competitive vendors. An important challenge in FL is to prevent a server or other vendors from reconstructing the private data of any vendor while collaborating at any circumstances. While a secure aggregation protocol in FL addresses the strong privacy of the data of the vendors, the FL framework creates a new attack surface during the model training process. Since the vendors have full control over local training processes, they may submit arbitrary updates to change the global model without being detected. Among the broad range of attacks on FL, the following attacks are the most relevant to our use case:

Model inversion attack

An attacker can intercept the updated local models and extract the private training data from the models. For example, in (Fredrikson, Jha, & Ristenpart, 2015), the authors demonstrated a model inversion attack that could extract images from a face recognition system, which look suspiciously similar to images from the underlying training data.

Local model poisoning attack

This attack injects poisoned instances into the training data, or directly manipulates model updates during the aggregation protocol. An attacker can compromise some vendors and thereby he may upload the poisoned local models, which are highly deviating from the global model. As a result, the attacker can tamper with the weights of the global model or inject a backdoor into it, misclassifying specific inputs into the target class as intended by the attacker.

Secure aggregation

Secure aggregation in FL is a cryptographic protocol that enables each vendor to submit a local model securely and a server learns nothing but the sum of the local models. A secure aggregation method for mobile networks was presented in (Bonawitz, et al., 2017) and (Bell, Bonawitz, Gascón, Lepoint, & Raykova, 2020). This method relies on a pairwise secret exchange and Shamir's *t*-out-of-*n* secret sharing scheme, focusing on the setting of mobile devices where communication is extremely expensive, and dropouts are common.

There is a rich literature exploring secure aggregation in both the single-server setting (via additive masking (Bonawitz K. A., et al., 2016), via threshold homomorphic encryption (HE) (Halevi, Lindell, & Pinkas, 2011), and via generic secure multi-party computation (MPC) (Burkhart, Strasser, Many, & Dimitropoulos, 2010) as well as in the multiple non-colluding servers setting (Corrigan-Gibbs & Boneh, 2017). For instance, one can perform all computations using a fully homomorphic encryption scheme resulting in low communication but very high computation or using classical MPC techniques with more communication but less computation. Other works use a hybrid of both and thus enjoy further improvement in performance (Mishra, Lehmkuhl, Srinivasan, Zheng, & Popa, 2020) (Juvekar, Vaikuntanathan, & Chandrakasan, 2018). Nevertheless, it is

still an open question how to construct a secure and robust aggregation protocol that addresses all the challenges.

Autoencoder

An autoencoder (AE) is a type of artificial neural network seeking to learn a compressed representation of an input in an unsupervised manner (Kramer, 1991). An AE is composed of two sub-models namely the encoder and the decoder, whereby the former is used to compress an input $\boldsymbol{x}$ into lower-dimensional latent-space representation $\boldsymbol{z}$ through a non-linear transformation, and the latter maps the encoded representation back into the estimated vector $\hat{\boldsymbol{x}}$ of the original input vector as follows:

$$\boldsymbol{z} = f(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}), \quad (1)$$
$$\hat{\boldsymbol{x}} = g(\boldsymbol{W}'\boldsymbol{z} + \boldsymbol{b}'), \quad (2)$$

where f and g represent the activation functions of the encoder and the decoder respectively. The weight matrix $\boldsymbol{W}$ (resp. $\boldsymbol{W}'$) and bias vector $\boldsymbol{b}$ (resp. $\boldsymbol{b}'$) are the learnable parameters for the encoder (resp. decoder).

The training objective of the autoencoder is to minimize the reconstruction error between the output $\hat{\boldsymbol{x}}$ and the input $\boldsymbol{x}$, referred as the loss function $\mathcal{L}(\theta)$, typically the mean square error (MSE), expressed as:

$$\mathcal{L}(\theta) = \sum \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^2 \quad (3)$$

where $\theta = \{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{W}', \boldsymbol{b}'\}$ denotes the set of the parameters to be optimized.

AE has been widely used for anomaly detection by adopting the reconstruction error as anomaly score. It is trained with only normal data representing the normal behavior. After training, AE will reconstruct the normal instances very well, while it will fail to reproduce the anomalous observations by yielding high reconstruction errors. The process of the classification of an instance as anomalous/normal is shown in Alg. 1.

Algorithm 1: Autoencoder based anomaly detection

Input: Normal dataset $\boldsymbol{x}$, anomalous dataset $\boldsymbol{x}^i \;\; i = 1, \dots, N,$ threshold θ
Output: reconstruction error $\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|$

```
train an autoencoder given the normal data x
for i = 1 to N do
    reconstruction error (i) = ||x^i − g(f(x^i))||
    if reconstruction error (i) > θ then
        x^i is anomalous
    else
        x^i is normal
    end if
end for
```

Gated Recurrent Unit

The Gated Recurrent Unit (GRU) recently proposed by (Cho, et al., 2014) to solve the gradient vanishing problem, is an improved version of standard recurrent neural networks (RNNs), used to process sequential data and to capture long-term dependencies. The typical structure of GRU contains two gates namely reset and update gates, controlling the flow of the information. The update gate regulates the information that flows into the memory, while the reset gate controls the information flowing out the memory.

The GRU cell is updated at each time step t by applying the following equations:

$$\boldsymbol{z}_t = \sigma(\boldsymbol{W}_z x_t + \boldsymbol{W}_z \boldsymbol{h}_{t-1} + \boldsymbol{b}_z) \quad (4)$$
$$\boldsymbol{r}_t = \sigma(\boldsymbol{W}_r x_t + \boldsymbol{W}_r \boldsymbol{h}_{t-1} + \boldsymbol{b}_r) \quad (5)$$
$$\widetilde{\boldsymbol{h}_t} = \tanh(\boldsymbol{W}_h x_t + \boldsymbol{W}_h(\boldsymbol{r}_t \circ \boldsymbol{h}_{t-1}) + \boldsymbol{b}_h) \quad (6)$$
$$\boldsymbol{h}_t = \boldsymbol{z}_t \circ \boldsymbol{h}_{t-1} + (1 - \boldsymbol{z}_t) \circ \widetilde{\boldsymbol{h}_t} \quad (7)$$

where $\boldsymbol{z}$ denotes the update gate, $\boldsymbol{r}$ represents the reset gate, $\boldsymbol{x}$ is the input vector, $\boldsymbol{h}$ is the output vector, $\boldsymbol{W}$ and $\boldsymbol{b}$ represent the weight matrix and the bias vector respectively. σ is the gate activation function and tanh represents the output activation function. The "$\circ$" operator represents the Hadamard product.

Related work

In (Bonawitz K. , et al., 2017), a practical secure aggregation technique in an FL setting was proposed over large mobile networks. Such a framework does not fit for our use case due to multiple reasons. Firstly, in our use case, a global model is not shared with data owners (vendors). Each vendor gets a benefit by receiving an individual maintenance result (e.g., the difference between the prediction and the real failure) after the global model is deployed and hardware degradation is predicted. Secondly, the scalability is not important since the number of vendors is not very large and dropouts are expected to be rare. On the other hand, secure aggregation is critical since the disclosure of the private training dataset may give negative impact on the data owner's business.

Another interesting work on collaborative predictive maintenance was presented in (Mohr, Becker, Möller, & Richter, 2020), where a combination of blockchain and federated learning techniques was applied. We apply a multi-party computation technique for data privacy since it is more suitable for our use case. More recently, in (Zheng, et al., 2021), an end-to-end platform for collaborative learning using MPC is proposed. Though it is an interesting approach, it is unlikely that this platform can be applied to our use case since the collaborative learning through the use of release policies and auditing is not preferable to the predictive maintenance.

Proposed Framework

Figure 1 illustrates the proposed secure collaborative learning framework for predictive maintenance in optical networks. We consider a FL approach that assumes *N* vendors for collaborative training of a global ML model under the control of an aggregator server hosted by an optical network operator, while keeping every client's data private. Each vendor builds a local model using its own training dataset and uploads it to the server. The private dataset remains in the vendor's domain and is never exposed to other companies. The local model updates are sent securely to the server. At the server side, an anomaly detection method adopted to defend against the local model poisoning attacks is used to firstly recognize the abnormal local model updates sent by potentially compromised vendors, which are discarded. Afterwards, a server builds a global ML model by aggregating only normal local ML models iteratively and averaging them to form an updated global model proportional to the size of dataset. An MPC-based secure aggregation defending against the model inversion attack is adopted. In our

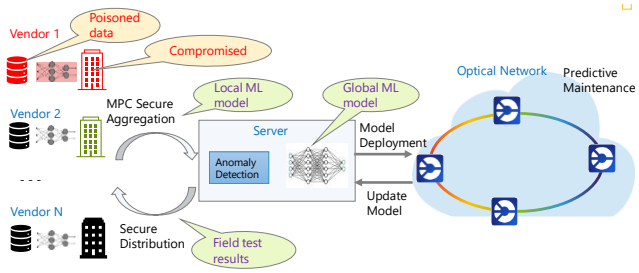

Fig. 1. ML-based predictive maintenance process in a dishonest setting.

framework, a secure aggregation protocol is tolerant to the malicious behavior of participants in an honest-majority model; that is, a server and majority of vendors are assumed to be honest, yet some may be malicious or unreliable. Using the global model, the potential risk of hardware failure or degradation and corresponding maintenance events are predicted, and the necessary resources are proactively prepared to run optical networks without disruption. Compared to the original FL, the local models are not many, and the dropouts are very rare in our framework. Furthermore, an updated global model is not shared with vendors. The reason is that, while a global model is a valuable asset to the network management, it is not really beneficial to the vendors. Instead, each vendor receives the personalized maintenance report which contains the discrepancy between its local model and the global model, which is useful to improve the quality of products in the future.

MPC-based Secure aggregation

Suppose that the server and vendors (clients) behave honestly, but curiously (semi-honest model). That is, all participants follow the protocol exactly as instructed, but also try to retrieve the private data of other vendors, if possible. Under this scenario, a simple *n*-out-of-*n* additive secret sharing scheme can be used to prevent the model inversion attack as well as keep the privacy of local models.

Suppose N is the number of clients, and each client has its own local model f_i where $1 \leq i \leq N$. The client i generates a random linear mask s_i and sends $f_i + s_i$ to the server.

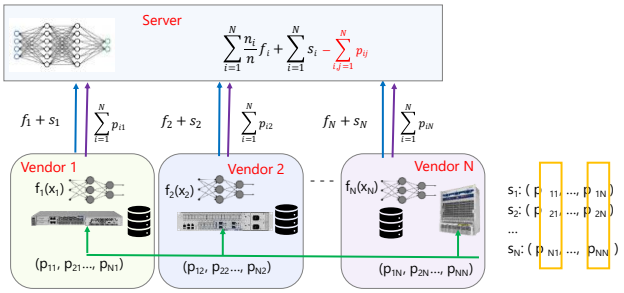

Fig. 2. Secure collaborative learning using Secret Sharing in FL.

Also, the client i divides s_i into N additive shares $\{p_{i1},\ldots,p_{iN}\}$ in such a way that $s_i = \sum_{j=1}^{N} p_{ij}$.

Note the size of s_i is similar to those of shares. These N shares are distributed to other clients in such a way that each client receives a unique share out of N shares. In result, the client i receives $\{p_{1i}, \ldots, p_{Ni}\}$.

Finally, the client i sends the sum of the shares $\sum_{j=1}^{N} p_{ij}$ to the server. This process is repeated for all clients. By aggregating one-time padded local models and the sum of the shares, the server can calculate the sum of the local models as follows:

$$\sum_{i=1}^{N}\left(\frac{n_i}{n} f_i + s_i\right) - \sum_{i,j=1}^{N} p_{ij} = \sum_{i=1}^{N} \frac{n_i}{n} f_i + \sum_{i=1}^{N}\left(s_i - \sum_{j=1}^{N} p_{ij}\right) = \sum_{i=1}^{N} \frac{n_i}{n} f_i$$

where n_i denotes the size of the data of the client i. n represents the size of the aggregated data of all the clients ($n = \sum_1^N n_i$)

An overview of the secure collaborative learning procedure is shown in Fig. 2.

Autoencoder based anomaly detection

An autoencoder based anomaly detection method is adopted to detect and exclude malicious local models updates from the aggregation process. It is used to compute the reconstruction errors of the local model updates. If the reconstruction errors are high, the model updates are considered malicious and thereby removed.

The autoencoder is trained with a dataset $D = \{w_1, w_2 \ldots w_N\}$ incorporating the local model updates (i.e.,

model weights) sent by trusted clients (i.e., normal weights) under no attack setting and stored at the server. The dimensionality of the model weight w_k is reduced to produce a low-dimensional input in order to reduce the computational complexity due to the high dimension of the model weight. The generated input is fed then to the autoencoder for training, whereby the encoder compresses the input into a lower-dimensional latent vector which is then reconstructed by the decoder.

After the training phase, the autoencoder is able to recognize the normal weights and mark any weight that deviates from the data seen during the training as an anomaly. The reconstruction error between the input weight and the reconstructed weight is used as an anomaly score. If the anomaly score exceeds a pre-defined threshold, the weight is recognized as anomalous potentially sent by a malicious client, and thereby it is removed and not considered for the update of the global model. The threshold is optimized in order to improve the detection capability of the autoencoder for different poisoning model attacks.

Validation of the Proposed Framework

Use case: Optical Transmitter Degradation Prediction

Semiconductor lasers are considered as one of the most commonly used optical transmitters for optical communication system thanks to their high efficiency, low cost, and compactness. They have been rapidly evolved to meet the requirements of the next generation optical network in terms of speed, power consumption, etc. However, during operation, the performance of the laser can be adversely impacted by several factors such as contamination, crystal defects, facet oxidation etc. Such factors are hard to predict, and their interaction can lead to complex degradation mechanisms which are hard to model. The semiconductor laser degradation occurs in three different modes: rapid, catastrophic, and gradual. Each degradation mode is characterized by its own signature depending on the laser's architecture and composition. Among the degradation models, a catastrophic mode is considered as the most challenging and hazardous ons as it appears as a quick and sudden failure after a normal operation of the device. Therefore, it is hard to predict such degradation leading to the end of the life of the laser, and thereby resulting in optical network disruption and high maintenance costs. Therefore, it is highly beneficial to predict the degradation of the semiconductor laser device after its deployment in optical communication system in order to enhance the system reliability and minimize the downtime costs.

ML techniques could provide a great potential to tackle the laser degradation prediction problem (Abdelli, Griesser, & Pachnicke, 2020). The development of such prognostic methods requires the availability of run-to-failure data sets modelling both the normal operation behavior and the degradation process under different operating conditions. However, such data is often unavailable due the scarcity of the failures during the system operation and the long time required to monitor the device up failing and then generating the reliability data. That is why accelerated aging tests are often adopted to collect run-to-failure data in shorter time by speeding up the device degradation by applying accelerated stress conditions resulting in the same degradation process leading to failure.

However, the burn-in aging tests are carried out just for few devices due to the high costs of performing such tests. Hence, the amount of the run-to-failure data that can be derived from such tests, might be small, which can adversely affect the performance of ML model (Abdelli, Griesser, & Pachnicke, A Hybrid CNN-LSTM Approach for Laser Remaining Useful Life Prediction, 2021). Therefore, an FL approach is considered as a promising candidate to address the aforementioned problem, whereby different semiconductor laser manufacturers (i.e vendors) collaborate with their small local dataset, stored at their premise, in order to build an accurate and reliable global laser degradation prediction model with good generalization and robustness capabilities.

Note that the semiconductor laser manufacturers might have different types of laser devices with various characteristics leading to different degradation trends, and that the data owned by each vendor is derived from aging tests conducted under different operating conditions. State that the global model is running on a server hosted by an optical network operator owning the infrastructure in which the semiconductor lasers manufactured by the different vendors are deployed.

We consider an FL system composed of a server and N clients (i.e., vendors) that collaboratively train a global model to predict the semiconductor laser degradation using the FedAvg algorithm (McMahan, Moore, Ramage, Hampson, & Arcas, 2017).

The clients securely send the local model weight updates to the server using MPC. A GRU based model is used as global model to solve the task of semiconductor laser degradation prediction. A convolutional autoencoder implemented at the server is adopted as an anomaly detection method to detect the anomalous weights sent by the malicious clients.

Experimental data

To evaluate our FL framework, we adopt different datasets obtained from semiconductor laser manufacturers. The datasets represent the reliability data of two different types of semiconductor lasers namely vertical-cavity surface-emitting laser (VCSEL) and tunable distributed feedback (DFB) laser. VCSEL and DFB lasers differ in semiconductor ma-

terials and resonator structures, and are characterized by different degradation trends. Each dataset is derived from various accelerated aging tests performed according to Telcordia GR-468 CORE requirements for multiple devices with various characteristics (e.g., VCSELs with different oxide aperture sizes…) operating under several operating conditions and carried out under high temperature T (50°C $\leq T \leq$ 150°C) to strongly increase the laser degradation and thereby speed up the device failure. Depending on the operating conditions, the duration of the aging tests is varied (i.e., 2000h, 3000h, 3500h, 15000h). The output power (i.e., degradation parameter) is monitored under constant operating current I. The failure criterion of the device is defined as the decrease of the output power by 1 dB (20%) of its initial value. Figure 3 shows examples of aging tests results of semiconductor lasers conducted under different operating conditions. As depicted in Fig. 3, few VCSELs are degraded or failed during the aging tests, whereas more tunable DFB lasers exhibit degradation.

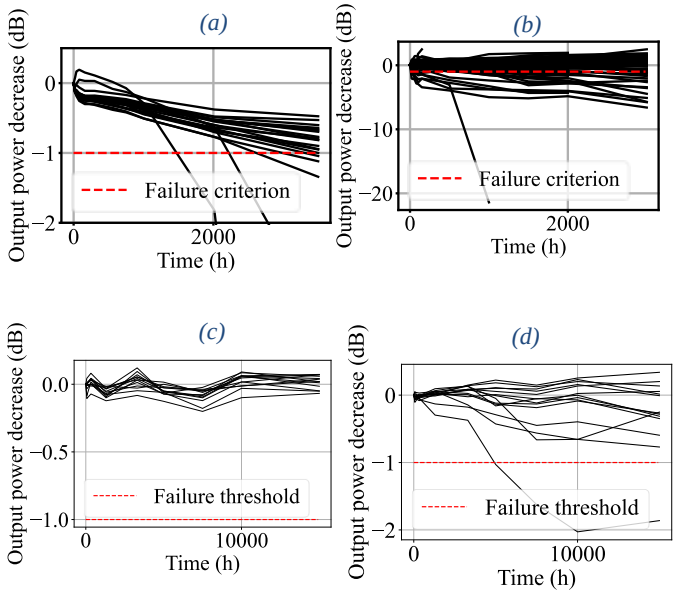

Fig. 3. Experimental aging test data of semiconductor lasers conducted under different operating conditions: (a) aging tests of VCSELs with different oxide aperture sizes performed at 85°C, (b) aging tests of tunable DFB lasers conducted at 90°C, (c) aging tests of VCSELs performed at 50°C, and (d) aging tests of VCSELs conducted at 70° C.

In total, a dataset of 6,564 samples incorporating 8-length sequences composed of monitored output power measurements of length 6 combined with the operating conditions namely T and I, is built. We assign to each sample the state of the device (normal or degraded). For training and testing the ML model to early predict the laser degradation, we consider the samples of early degraded devices (i.e., during the first stage of degradation, exhibiting a decrease of output power of value between 5% and 10%). The said data is then normalized and randomly divided into a training data (comprising of 80% of the samples) and a test dataset (the remaining 20% for testing). The training data is split then into N = 5 clients with different parts. Note that each client owns a data of either different types of lasers than the other clients or same type of lasers but manufactured by different laser manufacturers (i.e., different materials and structure) and tested on different wafers, leading to heterogeneous federated setting.

Global model

The adopted ML model to predict the degradation of the semiconductor laser is a GRU-based model as GRU is good at processing sequential data and to capture the relevant features underlying the laser degradation trend under different operating conditions. The architecture of the GRU model is composed of one GRU layer containing 25 cells. The GRU model takes as an input the sequence of length 8 including the output power measurement values collected till time t combined with T and I, and outputs the state of the device ("normal" or "degraded") at the prediction time t. The training of the global model is carried out in an iterative process as follows:

- The server distributes the global model ${w^t}_G$ to N clients.
- Each client k trains the model locally using its local data D_k and updates the weight ${w^t}_k$ for α epochs of Adam with mini-batch size of β to compute ${w^{t+1}}_k$.
- The server securely aggregates each client's ${w^{t+1}}_k$ using MPC.
- An autoencoder-based anomaly detection method is used to detect anomalous weights sent by the clients.
- The update of the global model ${w^{t+1}}_G$.is computed by a weighted averaging of only normal weights.

The above-described process is repeated for multiple communication rounds N_{round} (e.g., number of aggregation) to improve the performance of the global model. For our experiments, α, β and N_{round} are set to 8, 10 and 20, respectively.

Anomalous weight detection Method

A convolutional autoencoder implemented at the server is used to identify the anomalous weights and thereby detects the potentially malicious clients. The model contains an encoder and a decoder sub-model with 5 layers. The encoder takes an input

takes as an input a vector of length 75. It encodes the input into low dimensional features through a series of 2 convolutional layers containing 64 and 32 filters of size 3 x 1 with a stride of 1. The decoder is inversely symmetric to the encoder part. It consists of 3 transposed convolutional layers used to up-sample the feature maps. The last transposed convolutional layer with 1 filter of size 3 x 1 selected as an activation function for the hidden layers of the model. The loss function is set to the MSE, which is adjusted by using the Adam optimizer.

Experimental Results

Prediction Capability Evaluation

The performance of the proposed FL framework is compared to two baseline models including a model trained by applying a traditional centralized approach and a locally trained model without participating in the FL approach (i.e., localized model). The centralized approach is trained with the data from all the clients, which is collected and stored at a single server. The localized model is trained on the client's premises without model sharing during the training procedure. The different approaches are evaluated using as evaluation metrics the accuracy, the precision, quantifying the relevance of the predictions made by the ML model, the recall (i.e sensitivity), providing the total relevant results correctly classified by the ML model, and the F1 score, the harmonic mean of precision and recall. The results of the comparison shown in Fig. 4 demonstrate that first the FL framework outperforms the localized model by providing 11.4%, 9.62%, 14.7% and 12.24% improvements in accuracy, precision, recall and F1 score metrics, respectively, and second that the FL approach achieves similar performance as the centralized approach while ensuring data privacy.

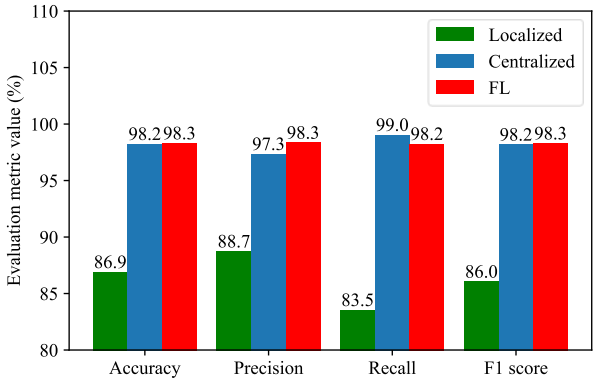

Fig. 4. Comparison of the federated (FL), centralized and localized approaches.

Figure 5 shows the states of some tested devices predicted by the ML model trained using the FL approach by giving it as input the output power measurements monitored till 5,000 h (i.e., time of prediction). As depicted in Fig. 5, the ML model accurately and early predicts the degraded devices before reaching the failure criterion, which proves the usefulness of the adopted ML model in early predicting the degraded/failed devices.

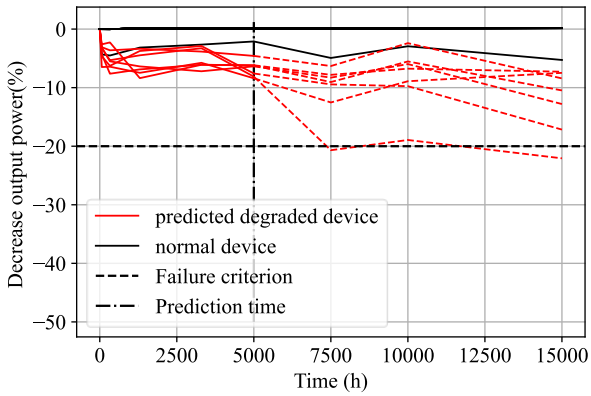

Fig. 5. Assessment of early degradation prediction capability of ML model.

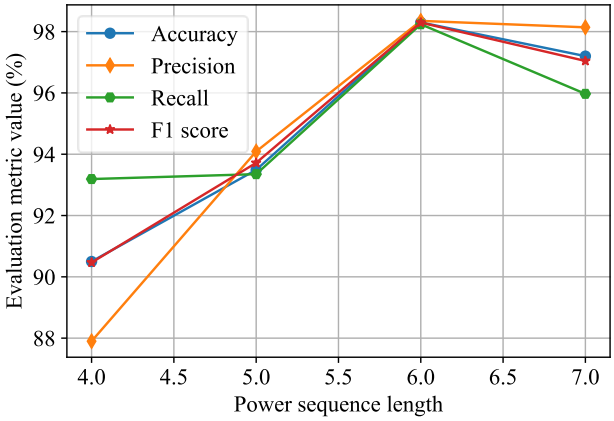

Fig. 6. Impact of the output power sequence length on the performance of the ML model.

The length of the input sequence, specifically the length of the sequence of the output power measurements, has a significant impact on the degradation prediction capability of the ML model. As shown in Fig. 6, increasing the length of the sequence of the output power measurements helps the ML model to capture more information about the degradation trend and thereby to achieve better degradation prediction capability performance (i.e., yielding better accuracy, precision, recall and F1 scores). However, rising the length of the sequence too much (higher than 6) can lead to overfitting and thus reduces the performance of the ML model.

Robustness to model poisoning attacks

The anomalous weight detection model is compared to defense-based methods namely krum (Blanchard, El Mhamdi, Guerraoui, & Stainer, 2017), Trimmed Mean (Yin, Chen, Kannan, & Bartlett, 2018) and Median. The testing accuracy

achieved by the global model for each communication round is adopted as evaluation metric. We consider the following three adversarial attacks launched by 20% of clients (i.e., one compromised or malicious vendor) for each communication round N_{round}:

- **Additive noise attack:** the compromised vendor k adds a Gaussian noise to the local model update and set it as $w_k = \bar{w}_k + \varepsilon$, where $\bar{w}_k$ denotes the original local model update or weight. ε is a vector derived from a gaussian distribution of mean 0 and standard deviation of 2 (i.e., standard deviation of the normal weights).
- **Sign flipping attack:** the malicious vendor k flips the sign of the local model weight as $w_k = \delta\, \bar{w}_k$, where δ is a constant selected randomly from a range from 1 to 5.
- **Same value attack**: the compromised vendor k sets its local model weight as $w_k = \beta\, \vec{1}$, where β is a constant set to 2 and $\vec{1}$ denotes all-one vector.

The results shown in Fig. 7 demonstrate that the proposed method significantly outperforms the defense-based approaches for the considered attack scenarios. It can be seen that the proposed method converges faster under all the settings and achieves similar performance as the FedAvg algorithm without attack, which proves the effectiveness of the anomaly detection model in detecting the anomalous weights and in mitigating the impact of launched attacks. As depicted in Fig. 7, the considered baselines are more robust to the additive noise attack and not effective against same value attack. The performances of the defense-based methods are worse as they are not effective in defending against attacks for not identically and independently distributed (iid) settings, and the fraction of the malicious clients which is required by Krum and Trimmed Mean cannot be known a priori in FL.

Let A denote the testing accuracy achieved by the global model trained under no attack setting for all the communication rounds, and $\bar{A}$ present the testing accuracy obtained by the global model trained under an attack launched each communication round. The impact of an attack (Δ) is defined as the reduction of the accuracy of the global model due to the attack. It is expressed as $\Delta\,(\%) = A - \bar{A}$.

Figure 8 illustrates that the proposed method is most robust to the different attacks by achieving the smallest attack impacts under all the considered attacks.

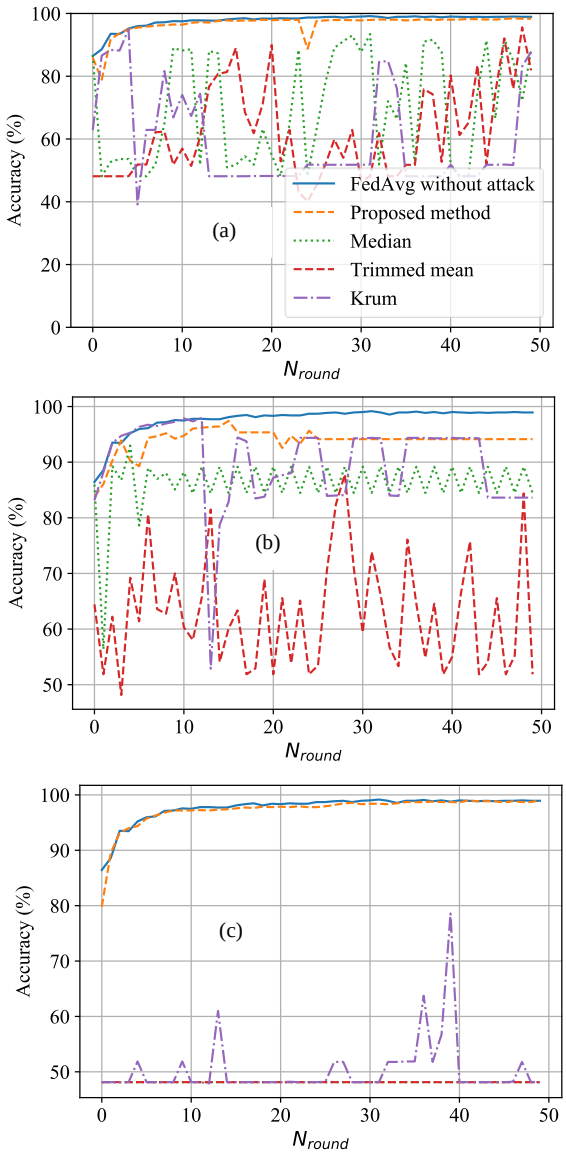

Fig. 7. Testing accuracy under different attack scenarios: (a) additive noise attack, (b) sign-flipping attack, and (c) same value attack.

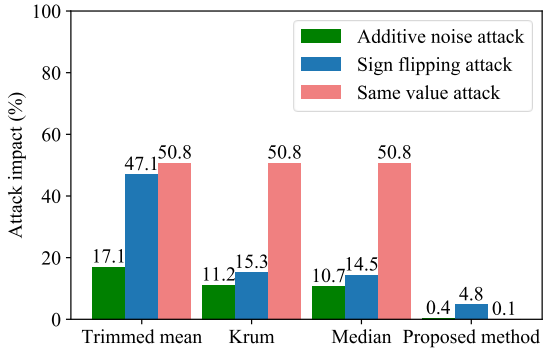

Fig. 8. Attack impacts of different model poisoning attacks on FL system with defense-based methods.

Conclusion

Optical networks require a high level of reliability and sustainability. Machine learning techniques are expected to improve maintaining such networks efficiently. We showed that an accurate and reliable ML model could be developed in collaborative learning without the disclosure of the clients' sensitive datasets even in a malicious setting. Our experiments confirm that (i) the presented FL approach achieves a good prediction capability similar to the one yielded by the centralized approach, and (ii) the proposed autoencoder based anomaly detection model is efficient in recognizing the anomalous weights potentially sent by malicious clients and outperforms the defense-based methods.

Acknowledgments

This work has been performed in the framework of the CELTIC-NEXT project AI-NET-PROTECT (Project ID C2019/3-4), and it is partly funded by the German Federal Ministry of Education and Research (FKZ16KIS1279K).

References

Abdelli, K., Griesser, H., & Pachnicke, S. (2020). Machine Learning Based Data Driven Diagnostic and Prognostic Approach for Laser Reliability Enhancement. 22nd International Conference on Transparent Optical Networks (ICTON).

Abdelli, K., Griesser, H., & Pachnicke, S. (2021). A Hybrid CNN-LSTM Approach for Laser Remaining Useful Life Prediction. Proc. of the Annual Conf. of the Prognostics and Health Management Society (PHM).

Bell, J., Bonawitz, K. A., Gascón, A., Lepoint, T., & Raykova, M. (2020). Secure Single-Server Aggregation with (Poly)Logarithmic Overhead. Cryptology ePrint Archive, Report 2020/704.

Blanchard, P., El Mhamdi, E. M., Guerraoui, R., & Stainer, J. (2017). Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent. Advances in Neural Information Processing Systems.

Bonawitz, K. A., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., . . . Seth, K. (2016). Practical Secure Aggregation for Federated Learning on User-Held Data. http://arxiv.org/abs/1611.04482.

Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., … Seth, K. (2017). Practical Secure Aggregation for Privacy-Preserving Machine Learning. (pp. 1175–1191). Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security.

Burkhart, M., Strasser, M., Many, D., & Dimitropoulos, X. (2010). SEPIA: Privacy-Preserving Aggregation of Multi-Domain Network Events and Statistics. 19th USENIX Security Symposium Security.

Cho, K., Merrienboer, B. v., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation.

Corrigan-Gibbs, H., & Boneh, D. (2017). Prio: Private, Robust, and Scalable Computation of Aggregate Statistics. Proceedings of the 14th USENIX Conference on Networked Systems Design and Implementation.

Fredrikson, M., Jha, S., & Ristenpart, T. (2015). Model Inversion Attacks That Exploit Confidence Information and Basic Countermeasures. Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security.

Halevi, S., Lindell, Y., & Pinkas, B. (2011). Secure computation on the web: Computing without simultaneous interaction. (pp. 132--150). Advances in Cryptology -- CRYPTO 2011.

Juvekar, C., Vaikuntanathan, V., & Chandrakasan, A. (2018). Gazelle: A Low Latency Framework for Secure Neural Network Inference. http://arxiv.org/abs/1801.05507.

Kramer, M. A. (1991). Nonlinear principal component analysis using autoassociative neural networks. AIChE Journal.

Li, L., Xu, W., Chen, T., Giannakis, G. B., & Ling, Q. (2018). RSA: Byzantine-Robust Stochastic Aggregation-Methods for Distributed Learning from Heterogeneous Datasets. http://arxiv.org/abs/1811.03761.

McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & Arcas, B. A. (2017). Communication-Efficient Learning of Deep Networks from Decentralized Data.

Mishra, P., Lehmkuhl, R., Srinivasan, A., Zheng, W., & Popa, R. A. (2020). Delphi: A Cryptographic Inference Service for Neural Networks. 29th USENIX Security Symposium.

Mohr, M., Becker, C., Möller, R., & Richter, M. (2020). Towards Collaborative Predictive Maintenance Leveraging Private Cross-Company Data. INFORMATIK 2020.

ReportLinker. (2021). Global Predictive Maintenance Industry. https://www.reportlinker.com/p05799417/Global-Predictive-Maintenance-Industry.html?utm_source=GNW.

Simon, J. (2021). Predictive maintenance with machine learning on AWS.

Yin, D., Chen, Y., Kannan, R., & Bartlett, P. (2018). Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates. Proceedings of Machine Learning Research.

Zheng, W., Deng, R., Chen, W., Popa, R. A., Panda, A., & Stoica, I. (2021). Cerebro: A Platform for Multi-Party Cryptographic Collaborative Learning. 30th USENIX Security Symposium.