International Journal of Forecasting

Privacy-preserving probabilistic wind forecasting using personalized vertical split learning

--Manuscript Draft--

Manuscript Number:	
Full Title:	Privacy-preserving probabilistic wind forecasting using personalized vertical split learning
Short Title:	Privacy-preserving probabilistic wind forecasting using personalized vertical split learning
Article Type:	Full Length Article
Keywords:	Data sharing, Multivariate time series, Personalization, Probabilistic forecasting, Split learning, Wind forecasting.
Corresponding Author:	Jean-Francois Toubeau BELGIUM
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	
Corresponding Author's Secondary Institution:	
First Author:	Jean-Francois Toubeau
First Author Secondary Information:	
Order of Authors:	Jean-Francois Toubeau
	Yi Wang, Prof.
	Fei Teng, Prof.
Order of Authors Secondary Information:	
Abstract:	This paper presents a new privacy-preserving framework for collaboratively learning space-time dependencies in the multi-horizon probabilistic forecasting of wind power. The approach relies on vertical split neural networks, i.e., a distributed setting that offers privacy by design via splitting a global deep learning model between the collaborative wind parks, which avoids explicitly sharing any raw data or details about local models in both training and inference stages. To achieve the optimal balance between exploiting global data while complying with local data distribution, the approach is here tailored to enable personalization, in a framework that is end-to-end trainable. Moreover, the model is augmented with specific layers designed to improve the robustness in case of operational issues (such as inaccurate or missing data). We evaluate several model configurations in a case study of seven correlated wind farms. Outcomes reveal that our proposed solution leads to relative improvements (measured with the quantile loss) of around 3% in comparison with purely private solutions without data sharing.
Suggested Reviewers:	Akylas Stratigakos MINES ParisTech, Centre for Processes, Renewable Energies and Energy Systems akylas.strat@hotmail.gr His research interests include energy forecasting
	Ricardo Bessa University of Porto Faculty of Engineering ricardo.j.bessa@inesctec.pt He is co-author of 50 journal papers, 97 conference papers and 6 book chapters about renewable energy forecasting, decision-making under uncertainty and innovative digital solutions (mainly data-driven methods) for smart grids.

	He is a co-founder of a spin-off company called Prewind, which sells forecasting services for wind power producers.
	Zacharie De Grève University of Mons Zacharie.DEGREVE@umons.ac.be Specialist in Machine Learning
	Kedi Zheng Tsinghua University zkd17@mails.tsinghua.edu.cn He is currently a Postdoctoral Researcher with Tsinghua University. His research interests include data analytics in power systems and electricity markets.
Opposed Reviewers:	
Additional Information:	
Question	**Response**

Jean-François Toubeau
Researcher, PhD
Department of Mechanical Engineering,
Division Applied Mechanics and Energy Conversion, KU Leuven
Celestijnenlaan 300, Post box 2421
B-3001 Leuven, Belgium.
Jean-Francois.TOUBEAU@kuleuven.be

Date : 12/04/2023

To: *International Journal of Forecasting*

Dear Editor of the *International Journal of Forecasting*,

We are pleased to submit our manuscript entitled "*Privacy-preserving probabilistic wind forecasting using personalized vertical split learning*" in your prestigious journal. This work is the result of the project proposal that received the **IIF-SAS grant in 2021 in the category of applications** (https://forecasters.org/programs/research-awards/iif-sas/).

The paper is a fruitful collaboration between KU Leuven (Belgium), Imperial College London (United Kingdom) and The University of Hong Kong, and presents an innovative collaborative solution for privacy-preserving energy forecasting, which is a milestone to convince people to participate in the energy digital transition. In particular, the methodology is generic, scalable to large systems, and can be applied for the probabilistic forecasting of any space-time dependent variables.

I am looking forward to answer any question regarding this paper.

Kind regards,

On behalf of all authors,

Jean-François Toubeau

Declaration of interests

☒The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Title Page (with Author Details)

Privacy-preserving probabilistic wind forecasting using personalized vertical split learning

Jean-François Toubeau[a,*], Yi Wang[b], Fei Teng[c]

[a]*Department of Mechanical Engineering, Division Applied Mechanics and Energy Conversion, KU Leuven, Celestijnenlaan 300, Post box 2421, Leuven (Heverlee), B-3001, Belgium*
[b]*Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong, China*
[c]*Department of Electrical and Electronic Engineering, Imperial College London, Exhibition Rd, South Kensington, London, SW7 2BX, United Kingdom*

Abstract

This paper presents a new privacy-preserving framework for collaboratively learning space-time dependencies in the multi-horizon probabilistic forecasting of wind power. The approach relies on vertical split neural networks, i.e., a distributed setting that offers privacy by design via splitting a global deep learning model between the collaborative wind parks, which avoids explicitly sharing any raw data or details about local models in both training and inference stages. To achieve the optimal balance between exploiting global data while complying with local data distribution, the approach is here tailored to enable personalization, in a framework that is end-to-end trainable. Moreover, the model is augmented with specific layers designed to improve the robustness in case of operational issues (such as inaccurate or missing data). We evaluate several model configurations in a case study of seven correlated wind farms. Outcomes reveal that our proposed solution leads to relative improvements (measured with the quantile loss) of around 3% in comparison with purely private solutions without data sharing.

*Corresponding author
Email address: jean-francois.toubeau@kuleuven.be (Jean-François Toubeau)

Privacy-preserving probabilistic wind forecasting using personalized vertical split learning

Abstract

This paper presents a new privacy-preserving framework for collaboratively learning space-time dependencies in the multi-horizon probabilistic forecasting of wind power. The approach relies on vertical split neural networks, i.e., a distributed setting that offers privacy by design via splitting a global deep learning model between the collaborative wind parks, which avoids explicitly sharing any raw data or details about local models in both training and inference stages. To achieve the optimal balance between exploiting global data while complying with local data distribution, the approach is here tailored to enable personalization, in a framework that is end-to-end trainable. Moreover, the model is augmented with specific layers designed to improve the robustness in case of operational issues (such as inaccurate or missing data). We evaluate several model configurations in a case study of seven correlated wind farms. Outcomes reveal that our proposed solution leads to relative improvements (measured with the quantile loss) of around 3% in comparison with purely private solutions without data sharing.

Keywords: Data sharing, Multivariate time series, Personalization, Probabilistic forecasting, Split learning, Wind forecasting.

1. Introduction

Renewable energy forecasting is an important component of decision-making in modern energy systems since it can help reduce operation costs without requiring any new investment in physical assets. Traditionally, forecasting tasks are carried out centrally by gathering information from all individual generators (e.g., wind parks) in a single database (Arrieta-Prieto and Schell, 2022; Hong et al., 2016). However, in the current liberalized environment, generators are typically owned by different entities, which may be reluctant to share their private data because it contains sensitive infor-

mation about their business operation (which could be exploited by market competitors (Cui et al., 2018)). If generators use only their own local data, they inevitably lose the explanatory power contained in measurements from neighboring sites, which may lead to opportunity costs and higher electricity costs for end-users (Webborn and Oreszczyn, 2019).

To address users' privacy and data security concerns of traditional (centralized) forecasters and thus promote collaboration among different data owners (Véliz and Grunewald, 2018), an efficient solution is provided by distributed learning. In particular, the alternating direction method of multipliers (ADMM) has been investigated due to its ability to collaboratively learn a global model by sharing only aggregate information instead of the raw data (Zhang et al., 2018; Zhang and Wang, 2018; Gonçalves et al., 2021). To alleviate shortcomings of such ADMM-based approaches that are tailored to convex models (Boyd et al., 2011), federated learning (FL) was recently proposed (McMahan et al., 2017) to accommodate non-linear, e.g., tree-based (Cheng et al., 2021) or deep learning (McMahan et al., 2017) structures.

However, traditional FL algorithms focus on horizontally distributed data, where parties have access to different samples from the same variables. The goal is thus to augment the database with samples from many clients to feed (and train) a global model (Toubeau et al., 2023). Unfortunately, such settings are unable to explicitly combine the information from several entities to form a complete feature set for each sample, which may reduce their capacity to learn cross-entities dependencies (Liu et al., 2022; Abuadbba et al., 2020). Hence, several approaches for vertically partitioned machine learning have been developed, dedicated for, e.g., linear and logistic regression (He et al., 2022), decision trees (Cheng et al., 2021), support vector machines (Shen et al., 2020), and feedforward neural networks (Zhang et al., 2018). These conventional solutions rely on cryptographic schemes such as secure multiparty computation (that enables parties to privately compute a function over their non-shareable inputs) and homomorphic encryption (that enables performing simple mathematical operations on encrypted values), which suffer from high computation and communication costs, such that current solutions are not yet scalable (Lloret-Talavera et al., 2021).

Alternatively, split learning was proposed in (Gupta and Raskar, 2018), wherein involved parties have access only to a portion of the global model, and only the output of these local models are shared with a central server (in every iteration) to make the predictions, such that a low communication bandwidth is needed. Split learning is highly versatile, which allows

it to accommodate different collaborative configurations for both horizontal and vertical partitioned data (Vepakomma et al., 2018). In contrast to FL, entities do not share their local model architecture nor its parameters, which provides an additional level of privacy. This, moreover, reduces the computational burden of the clients who need to run only a few local computations rather than the whole model (in both training and inference stages). However, similar to FL, split learning struggles in dealing with entities with heterogeneous data distributions. Moreover, the sequential nature of time series models requires clients-server communication for each individual time step, making vanilla split learning impractical for real-life applications (Abedi and Khan, 2020).

In light of this context, the key idea of this paper is to propose a new communication-efficient vertical split neural network for the collaborative multi-horizon probabilistic forecast of wind parks. The privacy-preserving model is tailored to capture all space-time dependencies among parks while enabling personalization to ensure that predictions comply with all individual data distributions. Overall, the contributions are threefold.

First, we adapt vertical split learning to accommodate advanced architectures for time series regression without adding any communication overhead. To that end, the proposed approach exploits the ability of sequential models to extract the relevant temporal context at each time step, such that time patterns can be decomposed over the prediction horizon. In this way, each wind park can share all its temporal features (in a single instance) with a central server, wherein a temporal alignment (between the wind parks) can then be achieved to capture space-time dependencies.

Second, we propose a personalization strategy to achieve the optimal trade-off between leveraging the relevant information from all wind parks while respecting local characteristics. This is achieved by sending the server's output back to individual wind parks that can hence recombine this global vector with their individual data using personalized processing layers for making the predictions of interest.

Third, we investigate different aggregation mechanisms to combine the outputs of the partial networks on the server side. Indeed, simply concatenating the client-side outputs (as done in traditional models) is not robust to missing or even outlier data, and other pooling layers (that can accommodate such bad data) are thus tested.

The resulting model is end-to-end differentiable (and can be learned using gradient descent) while innately protecting data privacy without encryption

algorithms or secure computation schemes. Outcomes from a case study composed of seven correlated wind parks show the advantages of the proposed method in comparison with other privacy-preserving techniques (such as federated learning) and purely private models (without data sharing) in terms of both forecasting performance and model robustness.

The rest of this paper is organized as follows. The formulation of the collaborative probabilistic forecasting problem and the concepts of (vertical) split learning are presented in Section 2. The proposed robust privacy-preserving time series forecaster is presented in Section 3, while section 4 discusses implementation details. Simulation results of the proposed method are provided in section 5, and the main conclusions and perspectives are drawn in Section 6.

2. Problem formulation and background

We consider a collaborative setting consisting of correlated wind parks, wherein the goal is to predict, at the forecast creation time t_0, the conditional distribution of wind power realizations $\mathbf{y}_{t_{1:T}}^{c_{1:C}} = \{y_{t_1}^{c_1}, ..., y_{t_T}^{c_C}\}$ across locations $c \in \mathcal{C} = \{c_1, ..., c_C\}$ and future time steps $t \in \mathcal{T} = \{t_1, ..., t_T\}$:

$$f_\theta = p\Big(\mathbf{y}_{t_{1:T}}^{c_{1:C}} \big| \underbrace{\mathbf{y}_{:t_0}^{c_{1:C}}, \mathbf{x}_{:t_0}^{(p),c_{1:C}}, \mathbf{x}_{t_{1:T}}^{(f),c_{1:C}}}_{\mathbf{x}_{t_0}^{\text{all}}}\Big) \tag{1}$$

where $\mathbf{x}_{t_0}^{\text{all}} = \left\{\mathbf{x}_{t_0}^{\text{all},c_1}, ..., \mathbf{x}_{t_0}^{\text{all},c_C}\right\}$ aggregates the input features from all wind parks, and is composed of the past wind measurements $\mathbf{y}_{:t_0}^{c_{1:C}}$ (before t_0), the past time-varying covariates $\mathbf{x}_{:t_0}^{(p),c_{1:C}}$ (before t_0), and the known future covariates $\mathbf{x}_{t_{1:T}}^{(f),c_{1:C}}$ (over the horizon $t_{1:T}$). Here, the covariates used for the proposed collaborative forecasting task are summarized in Table 1, and consist of historical power measurements, along with calendar-based and weather information for all wind parks.

Table 1: Input features of the wind power forecaster.

Past data $\mathbf{x}_{:t_0}^{(p),c_{1:C}}$	past measurements of wind speed and direction, calendar information (period of the day)
Known future data $\mathbf{x}_{t_{1:T}}^{(f),c_{1:C}}$	deterministic forecasted values of wind speed and direction (done by the system operator), calendar information (period of the day)

The target distribution (1) is here approximated by a set of conditional quantiles $\hat{y}_t^{(q),c}$, i.e., $p\left(y_t^c \leq \hat{y}_t^{(q),c} | \mathbf{x}_{t_0}^{\text{all}}\right) = q$, which are defined for Q relevant probability levels q (Toubeau et al., 2019). The model f_θ thus yields:

$$f_\theta\left(\mathbf{x}_{t_0}^{\text{all}}\right) = \hat{\mathbf{y}}_{t_{1:T}}^{c_{1:C}} = \left\{\hat{\mathbf{y}}_{t_{1:T}}^{(q),c_{1:C}}\right\}_{q \in \mathcal{Q}} \tag{2}$$

where the output $\hat{\mathbf{y}}_t^c$ of each client c for each time step t is thus Q-dimensional.

Due to data privacy considerations, individual data cannot be shared between wind parks or any other entity. The overall goal of the work is, therefore, to learn a shared model f_θ, while preserving data privacy, which is achieved by relying on vertical split learning (Gupta and Raskar, 2018).

2.1. Vertical split learning

In vanilla vertical split learning (represented in Figure 1), each client c has a (private and self-tailored) part $f_{\theta_x^c}$ of a shared model, and the outputs $\mathbf{a}^c = f_{\theta_x^c}\left(\mathbf{x}_{t_0}^{\text{all},c}\right)$ of these local models are fed to a central server. There, local outputs are aggregated $\mathbf{a} = f_{\text{agg}}\left(\mathbf{a}^{c_1}, ..., \mathbf{a}^{c_C}\right)$ and processed to make the predictions $\hat{\mathbf{y}}_{t_{1:T}}^{c_{1:C}}$ of interest (Ceballos et al., 2020). By combining information from different entities (carrying specific modalities of data) without sharing any raw data, the resulting model can capture cross-entities dependencies in a privacy-compliant way.

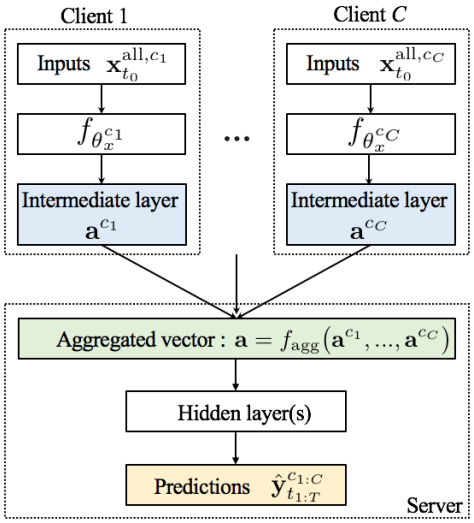

Figure 1: Architecture of vanilla vertical split learning.

2.2. Distance correlation

When all clients $c \in \mathcal{C}$ send their intermediate feature map $\mathbf{a}^c$ to the central server, an adversary may be able to reconstruct original raw data from these activations $\mathbf{a}^c$ (Yang et al., 2019). Inspired by (Vepakomma et al., 2020), we further robustify the procedure against honest-but-curious agents (i.e., that follow the communication and computation protocols while attempting to uncover private information from other entities) by minimizing the distance correlation between raw data $\mathbf{x}_{t_0}^{\text{all},c}$ and $\mathbf{a}^c$. This provides an extra level of security on top of the obfuscation given by local models $f_{\theta_x^c}$.

In statistics, the distance correlation is a measure of dependence between two random vectors (Székely et al., 2007), and is included in the interval $[0, 1]$, where a value of 0 means that the vectors are independent. Distance correlation is selected because of its unique set of advantages, i.e., i) it can be computed for two vectors (i.e., $\mathbf{x}_{t_0}^{\text{all},c}$ and $\mathbf{a}^c$) of different dimensions, ii) it captures both linear and nonlinear dependencies, iii) it has a closed-form that is easily computable (without requiring any tuning of additional parameters), and iv) it is fully differentiable (and can thus be used as a loss function in training).

3. Personalized split learning with vertically partitioned data

Vanilla (vertical) split learning models are developed for traditional feed-forward neural networks. However, such networks do not perform well for times series regression tasks (Toubeau et al., 2019). Here, split learning is thus adapted to leverage the temporal nature and modeling power of sequence-to-sequence (seq2seq) deep learning networks without requiring communication at each time step of the horizon $t_{1:T}$. Practically, each wind park has its own local (seq2seq) network $f_{\theta_x^c}$ that encodes its data $\mathbf{x}_{t_0}^{\text{all},c}$ into an output sequence $\mathbf{a}_{t_{1:T}}^c$. Local outputs from all parks are then transmitted (in a single communication step) on a shared server network that performs a temporal alignment of those outputs, hence capturing all underlying space-time dependencies.

In addition, although vanilla split learning avoids sharing any raw input data $\mathbf{x}_{t_0}^{\text{all},c}$, it involves sharing the labels $\mathbf{y}_{t_{1:T}}^c$. Since the labels contain sensitive information (i.e., actual wind realizations), we adopt a U-shaped configuration (Figure 2), wherein the server's output $\mathbf{z}_{t_{1:T}}^c$ are sent back to the wind park entities to make the final predictions $\hat{\mathbf{y}}_{t_{1:T}}^c$. In addition to preventing data leakages, this configuration can be used for personalization,

where each individual wind park c can tailor the last processing layers $f_{\theta_y^c}$ of the model to better fit its local data.

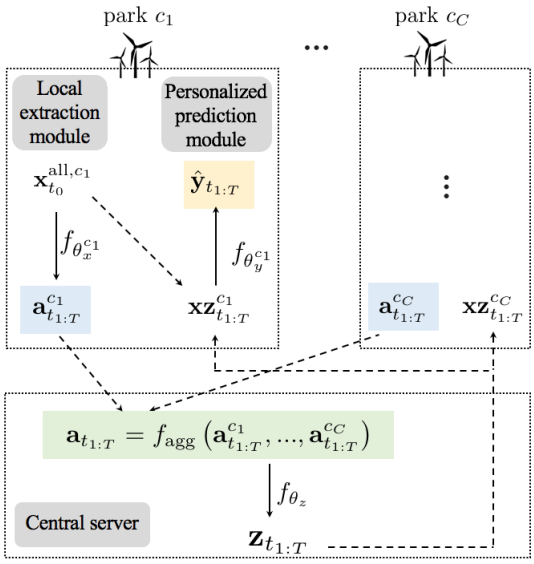

Figure 2: Forward pass of the personalized vertical split learning architecture, where the local extraction module and the personalized prediction module are only depicted for c_1.

The proposed collaborative model is composed of three main modules:

1. Feature extraction modules (section 3.1) that extract the relevant information to share with other collaborative entities, in a format that prevents reconstructing raw data.
2. A central server (section 3.2) that aggregates and processes the resulting space-time information using a strategy that is inherently robust to missing data.
3. Personalized prediction modules (section 3.3) that combine the shared server's output with local data to obtain a park-specific prediction.

The resulting model can be learned entirely end-to-end using gradient descent (section 3.4).

3.1. Local extraction modules

Each wind park processes its own data $\mathbf{x}_{t_0}^{\text{all},c}$, which is here carried out with seq2seq models $f_{\theta_x^c}$. In this architecture, an encoder processes past information $\left\{\mathbf{y}_{:t_0}^c, \mathbf{x}_{:t_0}^{(p),c}\right\}$ to convert it into a fixed-length vector, that is then used,

along with the known future data $\mathbf{x}_{t_{1:T}}^{(f),c}$, to generate the intermediate feature map $\mathbf{a}_{t_{1:T}}^{c}$. The goal is to decorrelate the output $\mathbf{a}_{t_{1:T}}^{c}$ from raw data $\mathbf{x}_{t_0}^{\text{all},c}$, while still containing explanatory power for the prediction task. Both encoder and decoder are modeled using Long Short Term Memory (LSTM) recurrent neural networks, but other architectures, e.g., Transformers (Vaswani et al., 2017), can be used.

It should be noted that both local data $\mathbf{x}_{t_{1:T}}^{\text{all},c}$ and the model design $f_{\theta_x^c}$ (architecture and parameters) are kept private, i.e., never shared with other wind park entities.

3.2. Central Server

The outputs $\mathbf{a}_{t_{1:T}}^{c}$ from all local extraction modules are then temporally aligned on the server. A naive approach consists in concatenating the local outputs along the dimension of features (left part of Figure 3), but this strategy requires having access to the intermediate outputs $\mathbf{a}_{t_{1:T}}^{c}$ from all entities on every iteration, which makes it very vulnerable in case of missing data (during the operational inference phase).

To tackle this problem, we rather propose element-wise operations across the local outputs $\mathbf{a}_{t_{1:T}}^{c}$ of all wind parks. As represented in the right part of Figure 3, this only involves that all local outputs have compatible shapes.

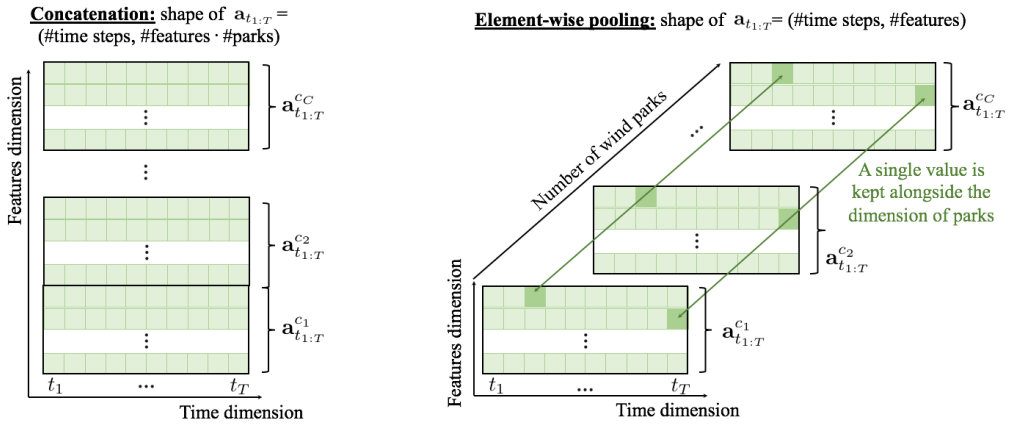

Figure 3: Aggregation of client-side outputs at the server side. Comparison between concatenation and element-wise pooling strategies.

Inspired by (Ceballos et al., 2020), we use the element-wise maximum, but alternatives such as element-wise average and element-wise sum are also

tested. The resulting matrix $\mathbf{a}_{t_{1:T}}$ is then processed by f_{θ_z} to extract the relevant space-time information $\mathbf{z}_{t_{1:T}}$ that will be returned back to individual wind park entities.

3.3. Personalized prediction modules

Since local wind power measurements may not be identically distributed, a personalization strategy composed of two components is adopted. First, each wind park c relies on its own local prediction module, which can thus be adapted (during training) to fit the park-specific wind power distribution. Second, a (data augmentation) procedure is carried out by concatenating (along the dimension of features) the server's output $\mathbf{z}_{t_{1:T}}$ with local data $\mathbf{x}^{\text{all},c}_{t_{1:T}}$ to form a new set of input features $\mathbf{xz}^c_{t_{1:T}}$. Overall, the augmented data $\mathbf{xz}^c_{t_{1:T}}$ are thus processed by park-specific layers for computing the quantiles of the forecast distribution, i.e., $\hat{\mathbf{y}}^c_{t_{1:T}} = f_{\theta^c_y}(\{\mathbf{xz}^c_{t_{1:T}}\})$.

To that end, a seq2seq model (different from the one in the extraction module) is used, where the encoder is fed with past information $\left\{\mathbf{y}^c_{:t_0}, \mathbf{x}^{(p),c}_{:t_0}\right\}$, while the decoder is fed with the augmented data $\mathbf{xz}^c_{t_{1:T}}$.

3.4. Training scheme

The goal of training is to find the best model f_θ in regards to two objectives, i) achieve the best forecasting performance on new future data (i.e., minimize the generalization error), and ii) protect the privacy of raw data. Since the true future data distribution is unknown, the training is carried out by minimizing the empirical risk, i.e., averaging the total loss $\ell(\cdot)$ over the training set composed of sequences $s \in \mathcal{S}$ of historical data $\{\mathbf{x}^{\text{all}}_{t_0}, \mathbf{y}^{c_{1:C}}_{t_{1:T}}\}_s$:

$$\min_\theta \sum_{s\in\mathcal{S}} \sum_{c\in\mathcal{C}} \sum_{t\in\mathcal{T}} \sum_{q\in\mathcal{Q}} \ell^c(\cdot) \tag{3}$$

where the total loss $\ell^c(\cdot)$ is composed of two different losses (to reflect both objectives), i.e., the quantile loss $QL^c(\cdot)$ that learns to predict the conditional quantiles, and the distance correlation $dCor^c(\cdot)$ between raw data $\mathbf{x}^{\text{all},c}_{t_0}$ and the intermediate feature map $\mathbf{a}^c_{t_{1:T}}$, i.e.,

$$\min_\theta \sum_{s\in\mathcal{S}} \sum_{c\in\mathcal{C}} \left(\alpha \cdot dCor^c(\mathbf{x}^{\text{all},c}_{t_0,s}, \mathbf{a}^c_{t_{1:T},s}) + \sum_{t\in\mathcal{T}} \sum_{q\in\mathcal{Q}} (1-\alpha) \cdot QL^c\left(\hat{y}^{(q),c}_{s,t}, y^c_{s,t}\right) \right) \tag{4}$$

where the α-value controls the level of privacy, by decreasing the dependence between the shared arrays $\mathbf{a}^c_{t_{1:T}}$ and raw local data $\mathbf{x}^{\text{all},c}_{t_0}$. The distance correlation $dCor(\cdot)$ is defined in section 2.2, while QL is given by:

$$QL(\hat{y}^{(q)}, y) = q \cdot \max(0, y - \hat{y}^{(q)}) + (1 - q) \cdot \max(0, \hat{y}^{(q)} - y) \tag{5}$$

Solving (4) is a non-convex task, and the model is thus trained using gradient descent method, i.e., an iterative procedure wherein we first compute the derivative of the loss function (4) with respect to parameters θ, and then adjust those parameters in the direction of the negative gradient $\theta \leftarrow \theta - \eta \nabla_\theta \ell(\cdot)$ towards a local minimum.

All gradients $\nabla_\theta \ell(\cdot)$ can be sequentially computed through backpropagation, as seen in Figure 4.

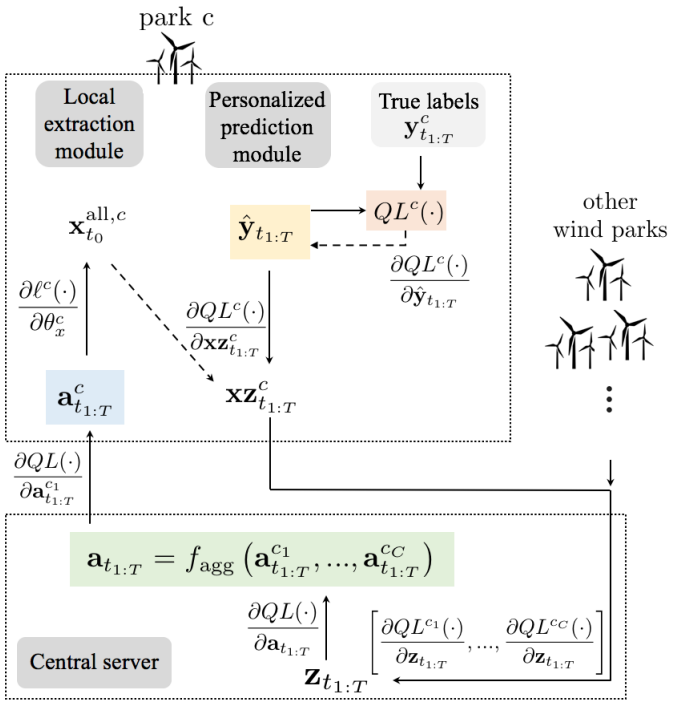

Figure 4: Backward pass (here depicted for a single wind park).

The training is initiated by individual wind parks c, each one calculating the derivatives of their own quantile loss with respect to predicted quantiles, i.e., $\frac{\partial QL^c(\cdot)}{\partial \hat{\mathbf{y}}^c_{t_{1:T}}}$. The training continues by applying the chain rule, working backwards to sequentially derive $\frac{\partial QL^c(\cdot)}{\partial \theta^c_y}$ and $\frac{\partial QL^c(\cdot)}{\partial \mathbf{xz}^c_{t_{1:T}}}$.

Then, each park shares $\frac{\partial QL^c(\cdot)}{\partial \mathbf{z}}$ with the central server that, in turn, computes the derivatives with respect to the server's parameters, i.e., $\frac{\partial QL(\cdot)}{\partial \theta_z}$.

The derivatives associated with the pooling layers, i.e., $\frac{\partial QL(\cdot)}{\partial \mathbf{a}_{t_{1:T}}}$, are also computed at the server side. It should be noted that each pooling layer leads to a different outcome. In this way, the element-wise maximum pooling returns zero gradients for non-maximum values and simply propagates $\frac{\partial QL(\cdot)}{\partial \mathbf{a}_{t_{1:T}}}$ for the input that actually corresponds to the maximum. On the other hand, the concatenation-based aggregation layer involves that the gradient values are split between their respective source layers, i.e., $\frac{\partial QL(\cdot)}{\partial \mathbf{a}_{t_{1:T}}} = \left[\frac{\partial QL(\cdot)}{\partial \mathbf{a}^{c1}_{t_{1:T}}}, \ldots, \frac{\partial QL(\cdot)}{\partial \mathbf{a}^{cC}_{t_{1:T}}}\right]$, where each component is sent to its corresponding wind park.

Finally, the backpropagated errors are transmitted back to the wind parks that finalize the training. This is achieved by computing the gradient of the total loss $\ell^c(\cdot)$ with respect to local parameters θ^c_x, hence jointly minimizing the distance correlation $dCor^c(\cdot)$ and the total quantile loss $QL(\cdot)$.

4. Implementation details

Before starting the collaborative learning, it should be noted that data samples from all clients need to be properly matched for training, i.e., (cleaned) samples need to be aligned across calendar information, which can be efficiently done using private set intersection (Pinkas et al., 2018).

Then, each wind park performs an offline local training to minimize the distance correlation $dCor^c(\mathbf{x}^{\text{all},c}_{t_0}, \mathbf{a}^c_{t_{1:T}})$. In this way, the amount of information shared with the server during the first iterations of the collaborative training is already obfuscated, thus preventing data leakage during early training stages.

The target distribution is estimated for $Q = 7$ probability levels q, i.e., the 5th, 15th, 25th, 50th, 75th, 85th and 95th percentiles. The training is carried out (according to section 3.4) using mini-batch stochastic gradient descent. Batches of 96 sequences $\{\mathbf{x}^{\text{all}}_{t_0}, \mathbf{y}^{c_{1:C}}_{t_{1:T}}\}_s$ are used during training. To ensure both generalization and unbiased model estimation, the historical sequences are divided into training, validation, and test sets via a (60%, 20%, 20%) allocation. The validation set is used for early stopping, while the test set enables estimating the performance of the resulting (trained) model on unseen conditions.

In the proposed collaborative setting, the wind parks need to agree on the optimization algorithm (here, the Adam algorithm with a starting learning rate of 0.001 is selected).

5. Case studies

The proposed privacy-preserving collaborative wind power forecasting method is tested on a publicly available dataset (Hong et al., 2014) composed of seven neighboring wind farms, recorded in hourly intervals over three years. As depicted in the correlation matrix in Figure 5, the seven wind parks are significantly correlated, with the exception of park 2 (and park 5 to a lesser extent) that exhibit weaker links with other parks.

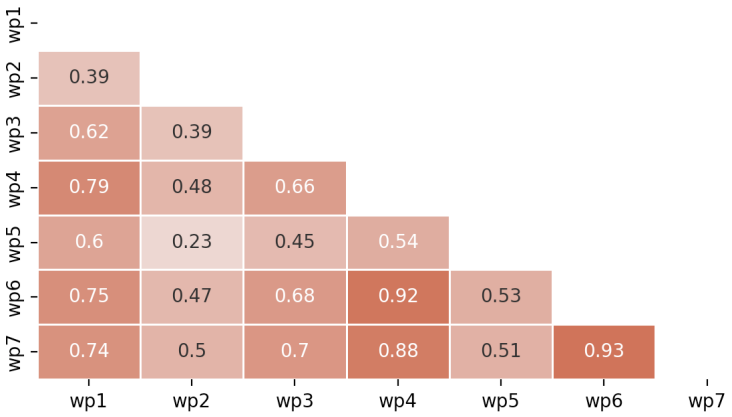

Figure 5: Correlation matrix yielding the linear dependencies among the seven wind parks.

The wind powers are predicted over a multi-horizon of $T = 6$ hourly intervals. A look-back window of $k = 8$ hours (used by all local models $f_{\theta_x^c}$) is selected to capture past dynamics. Other input features were presented in Table 1. After preliminary tests, we found that a value of $\alpha = 0.92$ in the loss function (4) leads to a very robust and consistent trade-off between forecasting performance and data privacy, and this value is thus used across all simulations. All features are individually scaled (using the training set) in the range [-1, 1].

For all tested models, the search for the optimal architecture (i.e., hyperparameter tuning) is done using random search (Bergstra and Bengio, 2012). The forecast performance is evaluated using two different metrics, with results given in p.u. (since wind powers are also used in p.u.).

First, we use the quantile loss, i.e., the same function that is minimized during training in (5), averaged over all time steps of the test set, all wind parks $c \in \mathcal{C}$ and all quantiles $q \in \mathcal{Q}$.

We complement the quantile loss with the Winkler score, which quantifies the forecast quality for different prediction intervals. For a prediction interval covering $(1-\beta)\cdot 100\%$, the Winkler score WS^{β} is defined as:

$$WS^{\beta} = \begin{cases} \delta & \hat{y}^{(\beta/2)} \leq y \leq \hat{y}^{(1-\beta/2)} \\ \delta + 2(\hat{y}^{(\beta/2)} - y)/\beta & y < \hat{y}^{(\beta/2)} \\ \delta + 2(y - \hat{y}^{(1-\beta/2)})/\beta & y > \hat{y}^{(1-\beta/2)} \end{cases} \tag{6}$$

where $\hat{y}^{(\beta/2)}$ and $\hat{y}^{(1-\beta/2)}$ are respectively the lower and upper bounds of the prediction interval $\delta = \hat{y}^{(\beta/2)} - \hat{y}^{(1-\beta/2)}$ (defined by the confidence level β). Hence, the Winkler score increases in case of large uncertainty and when the actual observation falls outside the predicted interval. It is thus preferable to have lower Winkler scores.

The data privacy of each wind park is measured using the distance correlation (defined in section 2.2) between the raw data $\mathbf{x}_{t_0}^{\text{all},c}$ and the information $\mathbf{a}_{t_{1:T}}^{c}$ shared with the server.

5.1. Ablation study

In this part, we perform an ablation study by measuring the change in performance of the proposed collaborative model (denoted by Ref) when removing/adapting important components of its architecture. In particular, four different aspects are investigated:

- *ref-dCor*, i.e., the *ref* model without the integration of the *dCor* minimization into the training procedure.
- *ref-noPers*, i.e., the *ref* model without the personalization strategy. Hence, the prediction modules $f_{\theta_y^c}$ rely only on $\mathbf{z}_{t_{1:T}}$.
- *ref-noSeq*, the *ref* model wherein the seq2seq models are replaced by feedforward networks. This topology reduces to a U-shaped vanilla (vertical) split neural network.
- *ref-concat*, i.e., the *ref* model using the concatenation as the aggregation layer f_{agg} (instead of the element-wise maximum). In addition, the element-wise mean (*ref-mean*) and the element-wise sum (*ref-sum*) are also tested.

The prediction performance and privacy measure (through the distance correlation) of all models are provided in Table 2. The training times are not reported since they are quite similar and consistently lower than 10 minutes. Typically, the training phase lasts around 10 seconds per epoch (for a convergence after 40 epochs).

Table 2: Forecasting performance and privacy preservation of different architectures of split learning-based models.

Model	QL [pu]	Winkler score [pu]			$dCor$	
		$\beta = 0.5$	$\beta = 0.3$	$\beta = 0.1$	mean	max
ref	0.176	0.260	0.321	0.451	0.15	0.24
ref-dCor	0.175	0.257	0.320	0.449	0.85	0.89
ref-noPers	0.187	0.276	0.343	0.473	0.18	0.37
ref-noSeq	0.190	0.281	0.350	0.480	0.32	0.36
ref-concat	0.178	0.263	0.327	0.453	0.23	0.37
ref-mean	0.177	0.262	0.326	0.451	0.25	0.32
ref-sum	0.178	0.262	0.326	0.453	0.31	0.44

First, we observe that the best forecasting accuracy is provided by the *ref-dCor* model, but at the expense of the privacy of the raw data. Thus is reflected by high values of the distance correlation between such individual raw data $\mathbf{x}_{t_0}^{\text{all},c}$ and the information $\mathbf{a}_{t_{1:T}}^{c}$ shared with the server. In this way, adding the distance correlation minimization in the training procedure enables decreasing the average distance correlation from 0.85 to 0.15. However, it is interesting to notice that our *ref* model achieves a performance close to the *ref-dCor* model, which shows that data privacy can be improved without significantly degrading the prediction accuracy.

In Figure 6, we represent the evolution of the distance correlation over the whole training for wind park 1, which is well representative of all parks. It is divided into two steps, starting from training local extraction modules $f_{\theta_x^c}^{c}$ with the goal of solely minimizing the distance correlation. Then, the collaborative model is trained to minimize the total loss defined in (4). We see that minimizing the local distance correlation ensures that raw data $\mathbf{x}_{t_0}^{\text{all},c}$ are well obfuscated before starting the collaborative training (i.e., $dCor < 0.1$). Data remain remains well protected thereafter (i.e., $dCor < 0.2$). The slight increase during the collaborative phase enables us to keep the correlation necessary to successfully achieve the forecasting task.

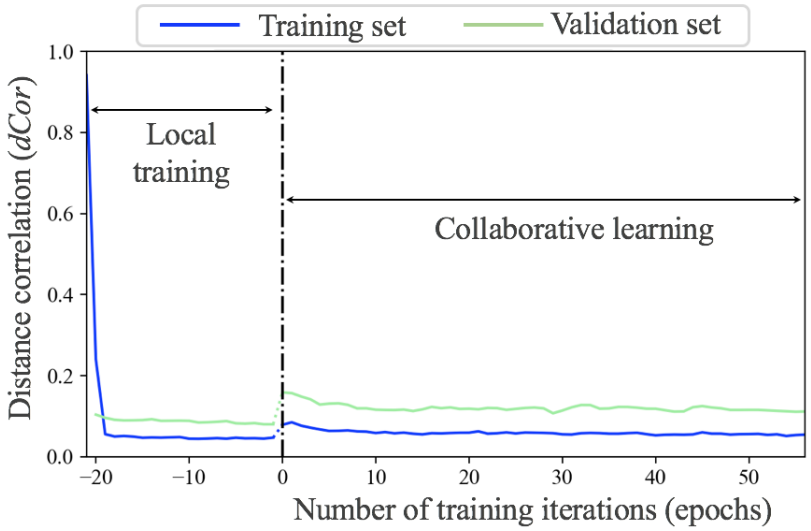

Figure 6: Evolution of the distance correlation with raw data over the training set (blue) and the validation set (green) for wind park 1.

The highest loss of prediction accuracy is given by the *ref-noSeq* model, which confirms the importance of properly capturing time dependencies in the wind power forecasting task. In particular, the *ref* model relies on seq2seq models for both extraction and prediction modules. The encoder and decoder networks of the feature extraction are composed of a single LSTM-based layer with 10 neurons. The same architecture is used for the prediction module but with 20 neurons in both encoder and decoder networks.

For illustrating the quality of the *ref* model, the probabilistic wind power forecasts (given in pu) of 3 parks (i.e., 1, 2, and 5) during two days (i.e., a summer and a winter afternoon) are shown in Figure 7. The gray areas cover the prediction intervals, while the red line represents the actual hourly wind power realizations. For both days, we observe that wind power patterns are well captured by the *ref* model, even for park 2 that exhibits a slightly different behavior (in line with its low correlation with other parks, see Figure 5).

In Table 2, we also see that the personalization strategy is also instrumental for the prediction performance. Only relying on the shared server's output $\mathbf{z}_{t_{1:T}}$ leads to significant losses, which tends to demonstrate that the server struggles in its dual mission to generalize from all wind parks (by grasping space-time dependencies), while capturing local data distributions.

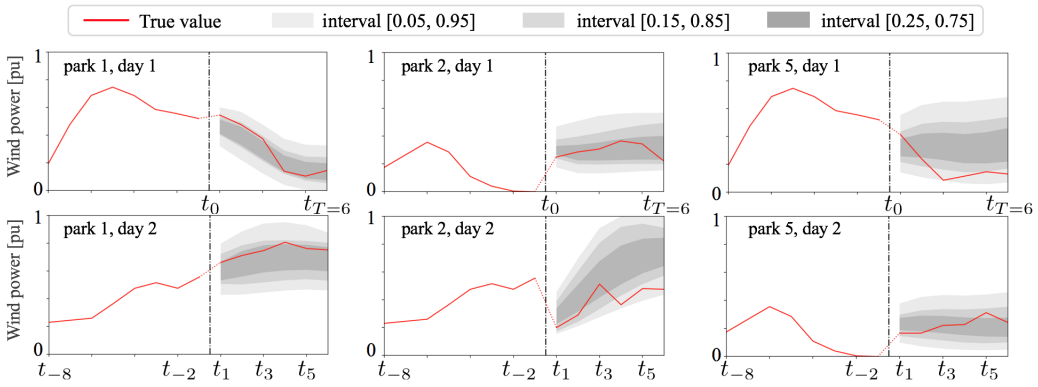

Figure 7: Multi-horizon probabilistic forecasts of wind power for parks 1, 2, and 5 for two different days.

Finally, simulations reveal that the max pooling layer outperforms other topologies, including the concatenation layer. This suggests that, in addition, to provide a robust framework in case of client dropout or inaccurate data (as further investigated in section 5.4), those pooling also serve of efficient downsampling, i.e., creating a lower resolution version of the input signals that disregards the elements useless for the forecasting task.

5.2. Comparison with centralized and purely private models

In this part, we compare the forecasting performance of the ref model with centralized and fully private models.

On the one hand, the purely private (non-cooperating) setting consists in training a different forecasting model for each wind park individually, assuming there is no collaboration. On the other hand, the (non-private) central models rely on complete information, i.e., an ideal (but impractical) case where each wind park builds its own model using all private data from other parks (Toubeau et al., 2021).

Both settings are tested with a traditional seq2seq model. In complement, we also implement a local multilayer perceptron (MLP), or feedforward neural network, to investigate the importance of explicitly capturing time dependencies. All these models are compared in Figure 8, where the forecasting performance (measured with the quantile loss) is given for each of the seven wind parks.

First, there is a significant performance gap between seq2seq centralized and seq2seq local models, which shows the practical interest of data shar-

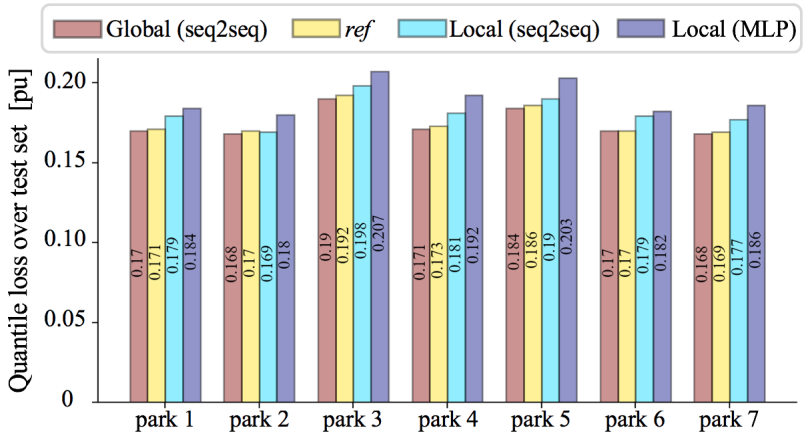

Figure 8: Comparison of the forecast accuracy (given by the quantile loss) of different centralized and fully-private models for the seven wind parks.

ing for correlated wind parks. Interestingly, this can be efficiently achieved through collaboration since our proposed *ref* model leads to a relative improvement of the quantile loss (averaged over the seven parks) with respect to local seq2seq models equal to around 3%. It should, however, be noted that, even with the personalization strategy, the *ref* model is slightly worse than the local model for wind park 2, which demonstrates the importance of relying on significant inter-park dependencies to participate to the collaborative learning task.

Second, we also confirm the importance of properly capturing time dependencies since local seq2seq models outperform local MLPs by 4.5% (regarding the quantile loss QL averaged over the seven wind parks). This highlights the inherent limitations of feedforward neural networks and by extension, of vanilla split neural networks for time series forecasting tasks.

5.3. Comparison with other collaborative models

Then, we consider two alternative collaborative methods: the Federated Averaging (FedAv) learning algorithm (McMahan et al., 2017) and horizontal split learning (HSplit). The outcomes are compared with our proposed *ref* model for the seven wind parks in Figure 9.

In FedAv, a global model is fully shared among wind parks. During the iterative training procedure, parks perform local computations on their private dataset to improve the model parameters θ. These local updates are then sent back to the server, which calculates the global average until

convergence. In HSplit, each wind park has its own local extraction module. The intermediate output is then shared with the global (shared) server that sends the information back to the park for the final prediction. In this setting (that aims at enriching sample diversity), the server is thus responsible to learn from sequences of all wind parks to improve generalization.

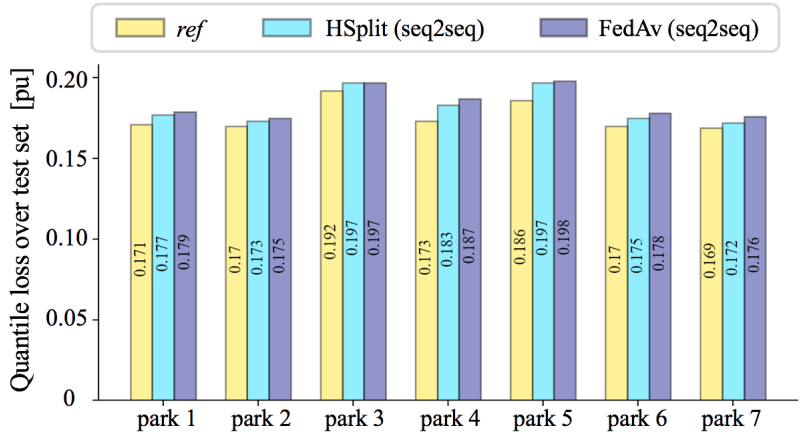

Figure 9: Comparison of the forecast accuracy (given by the quantile loss) of different collaborative models for the seven wind parks.

Although federated learning (FedAv) augments the database (by training a single shared model on the samples from all wind parks), we see that it fails to properly capture space dependencies among correlated variables.

Simulations also revealed that Hsplit is very sensitive to local overfitting. In particular, when the training is performed sequentially for each wind park, the results are dramatically biased in favor of the last trained parks. In particular, training from park 1 to 7, leads to a quantile loss of 0.29 for park 1 and 0.168 for park 7. Such discrepancies are unacceptable for collaborative entities, and we therefore mitigate this issue by shuffling all training sequences, which stabilizes the variability across parks.

5.4. Robustness to data errors

To investigate the robustness of the proposed *ref* model, we study the forecasting performance in case of an unexpected drop of clients (during the inference stage), which can be seen as an extreme case of data poisoning. A sensitivity analysis on the ratio of defaulting wind parks is carried out, and results are summarized in Table 3.

Table 3: Comparison of pooling strategies of the server on the forecasting performance (measured with the quantile loss) for different rates of clients dropping during testing.

Pooling	Quantile loss QL [pu]		
	$r = 0.1$	$r = 1$	$r = 2$
element-wise max	0.176	0.177	0.181
element-wise mean	0.177	0.180	0.185
element-wise sum	0.178	0.181	0.187

As expected, the performance suffers as a consequence of the clients dropping, which arises from the loss of the explanatory power of the underlying features $\mathbf{x}_{t_0}^{all,c}$. However, the performance loss is less severe for the element-wise maximum pooling strategy.

Moreover, we also observe that the collaborative setting enables obtaining prediction for all wind parks, even when some of them suffer from missing data (e.g., due to local sensor failures), which is achieved by leveraging the information given by other parks. This is a strong advantage with respect to fully private solutions, wherein the loss of input information cannot be mitigated by correlated variables from other entities.

6. Conclusion and perspectives

This paper presented a general collaborative framework for capturing space-time dependencies in privacy-preserving regression tasks and is here applied to the probabilistic forecast of wind power among neighboring parks. Outcomes revealed that the combination of (vertical) split learning with personalization is instrumental to unlock the value of collaboration. However, it is important to properly select the pooling layer (wherein local data are aggregated on a central server) to ensure robustness in case of missing information.

An important perspective of this work is to investigate how do we select the entities participating in collaboration since it has been shown that the model can struggle for weaker correlations. Also, we should further study how do we select the best model to ensure fairness among all collaborative entities (i.e., ensure that the gains of each entity are reflective of the added value provided to other agents).

References

M. Arrieta-Prieto, K. R. Schell, Spatio-temporal probabilistic forecasting of wind power for multiple farms: A copula-based hybrid model, International Journal of Forecasting 38 (2022) 300–320. doi:https://doi.org/10.1016/j.ijforecast.2021.05.013.

T. Hong, P. Pinson, S. Fan, H. Zareipour, A. Troccoli, R. J. Hyndman, Probabilistic energy forecasting: Global energy forecasting competition 2014 and beyond, International Journal of Forecasting 32 (2016) 896–913. doi:https://doi.org/10.1016/j.ijforecast.2016.02.001.

L. Cui, G. Xie, Y. Qu, L. Gao, Y. Yang, Security and privacy in smart cities: Challenges and opportunities, IEEE Access 6 (2018) 46134–46145. doi:10.1109/ACCESS.2018.2853985.

E. Webborn, T. Oreszczyn, Champion the energy data revolution, Nature Energy 4 (2019) 624–626. doi:https://doi.org/10.1038/s41560-019-0432-0.

C. Véliz, P. Grunewald, Protecting data privacy is key to a smart energy future, Nature Energy 3 (2018) 702–704. doi:https://doi.org/10.1038/s41560-018-0203-3.

X. Zhang, M. M. Khalili, M. Liu, Improving the privacy and accuracy of ADMM-based distributed algorithms, in: International Conference on Machine Learning, volume 80, 2018, pp. 5796–5805.

Y. Zhang, J. Wang, A distributed approach for wind power probabilistic forecasting considering spatio-temporal correlation without direct access to off-site information, IEEE Transactions on Power Systems 33 (2018) 5714–5726. doi:10.1109/TPWRS.2018.2822784.

C. Gonçalves, R. J. Bessa, P. Pinson, Privacy-preserving distributed learning for renewable energy forecasting, IEEE Transactions on Sustainable Energy 12 (2021) 1777–1787. doi:10.1109/TSTE.2021.3065117.

S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, Found. Trends Mach. Learn. 3 (2011) 1–122. doi:10.1561/2200000016.

B. McMahan, E. Moore, D. Ramage, S. Hampson, B. A. y. Arcas, Communication-Efficient Learning of Deep Networks from Decentralized Data, in: A. Singh, J. Zhu (Eds.), Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, volume 54 of *Proceedings of Machine Learning Research*, PMLR, 2017, pp. 1273–1282.

K. Cheng, T. Fan, Y. Jin, Y. Liu, T. Chen, D. Papadopoulos, Q. Yang, Secureboost: A lossless federated learning framework, IEEE Intelligent Systems 36 (2021) 87–98. doi:10.1109/MIS.2021.3082561.

H. B. McMahan, D. Ramage, K. Talwar, L. Zhang, Learning differentially private recurrent language models, in: International Conference on Learning Representations, 2017.

J.-F. Toubeau, F. Teng, T. Morstyn, L. V. Krannichfeldt, Y. Wang, Privacy-preserving probabilistic voltage forecasting in local energy communities, IEEE Transactions on Smart Grid 14 (2023) 798–809. doi:10.1109/TSG.2022.3187557.

Y. Liu, X. Zhang, Y. Kang, L. Li, T. Chen, M. Hong, Q. Yang, Fedbcd: A communication-efficient collaborative learning framework for distributed features, IEEE Transactions on Signal Processing 70 (2022) 4277–4290. doi:10.1109/TSP.2022.3198176.

S. Abuadbba, K. Kim, M. Kim, C. Thapa, S. A. Camtepe, Y. Gao, H. Kim, S. Nepal, Can we use split learning on 1d cnn models for privacy preserving training?, in: The 15th ACM ASIA Conference on Computer and Communications Security (ACM ASIACCS 2020), 2020.

D. He, R. Du, S. Zhu, M. Zhang, K. Liang, S. Chan, Secure logistic regression for vertical federated learning, IEEE Internet Computing 26 (2022) 61–68. doi:10.1109/MIC.2021.3138853.

M. Shen, J. Zhang, L. Zhu, K. Xu, X. Tang, Secure svm training over vertically-partitioned datasets using consortium blockchain for vehicular social networks, IEEE Transactions on Vehicular Technology 69 (2020) 5773–5783. doi:10.1109/TVT.2019.2957425.

Q. Zhang, C. Wang, H. Wu, C. Xin, T. V. Phuong, Gelu-net: A globally encrypted, locally unencrypted deep neural network for privacy-preserved learning, in: Proceedings of the Twenty-Seventh International

Joint Conference on Artificial Intelligence, IJCAI-18, International Joint Conferences on Artificial Intelligence Organization, 2018, pp. 3933–3939. doi:10.24963/ijcai.2018/547.

G. Lloret-Talavera, M. Jorda, H. Servat, F. Boemer, C. Chauhan, S. Tomishima, N. N. Shah, A. J. Pena, Enabling homomorphically encrypted inference for large dnn models, IEEE Transactions on Computers (2021) 1–1. doi:10.1109/TC.2021.3076123.

O. Gupta, R. Raskar, Distributed learning of deep neural network over multiple agents, Journal of Network and Computer Applications 116 (2018) 1–8. doi:https://doi.org/10.1016/j.jnca.2018.05.003.

P. Vepakomma, O. Gupta, T. Swedish, R. Raskar, Split learning for health: Distributed deep learning without sharing raw patient data, CoRR abs/1812.00564 (2018). arXiv:1812.00564.

A. Abedi, S. S. Khan, Fedsl: Federated split learning on distributed sequential data in recurrent neural networks, 2020. doi:10.48550/ARXIV.2011.03180.

J.-F. Toubeau, J. Bottieau, F. Vallée, Z. De Grève, Deep learning-based multivariate probabilistic forecasting for short-term scheduling in power markets, IEEE Transactions on Power Systems 34 (2019) 1203–1215. doi:10.1109/TPWRS.2018.2870041.

I. Ceballos, V. Sharma, E. Mugica, A. Singh, A. Roman, P. Vepakomma, R. Raskar, Splitnn-driven vertical partitioning, CoRR abs/2008.04137 (2020). URL: https://arxiv.org/abs/2008.04137.

Z. Yang, J. Zhang, E.-C. Chang, Z. Liang, Neural network inversion in adversarial setting via background knowledge alignment, Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (2019).

P. Vepakomma, A. Singh, O. Gupta, R. Raskar, Nopeek: Information leakage reduction to share activations in distributed deep learning, in: 2020 International Conference on Data Mining Workshops (ICDMW), IEEE Computer Society, Los Alamitos, CA, USA, 2020, pp. 933–942. doi:10.1109/ICDMW51313.2020.00134.

G. J. Székely, M. L. Rizzo, N. K. Bakirov, Measuring and testing dependence by correlation of distances, The Annals of Statistics 35 (2007) 2769 – 2794. doi:10.1214/009053607000000505.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, volume 30, 2017.

B. Pinkas, T. Schneider, M. Zohner, Scalable private set intersection based on ot extension, ACM Transactions on Privacy and Security (TOPS) 21 (2018) 1 – 35.

T. Hong, P. Pinson, S. Fan, Global energy forecasting competition 2012, International Journal of Forecasting 30 (2014) 357–363. doi:https://doi.org/10.1016/j.ijforecast.2013.07.001.

J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization, J. Mach. Learn. Res. 13 (2012) 281–305.

J.-F. Toubeau, T. Morstyn, J. Bottieau, K. Zheng, D. Apostolopoulou, Z. De Grève, Y. Wang, F. Vallée, Capturing spatio-temporal dependencies in the probabilistic forecasting of distribution locational marginal prices, IEEE Transactions on Smart Grid 12 (2021) 2663–2674. doi:10.1109/TSG.2020.3047863.