# A Real-World Deployment of Federated Learning for Residential Solar PV Power Forecasting

Frederick Apina[1,3], Diogo Monteiro[2,4], Bruno Dias[5], Hugo Morais[2,4], and Lucas Pereira[1,4]

[1] Interactive Technologies Institute, LARSyS, Lisbon, Portugal
[2] Instituto de Engenharia de Sistemas e Computadores - Investigação e Desenvolvimento, Lisbon, Portugal
[3] Faculty of Exact Sciences and Engineering, University of Madeira, Funchal, Portugal
[4] Instituto Superior Técnico (IST), University of Lisbon, Lisbon, Portugal
[5] Department of Electrical Energy, Federal University of Juiz de Fora (UFJF), Juiz de Fora, MG, Brazil

**Abstract.** Accurate Solar Photovoltaics (PV) power forecasting is essential for the stability and efficiency of modern smart grids, but traditional centralized Machine Learning (ML) approaches raise significant data privacy concerns for prosumers. Federated Learning (FL) offers a privacy-preserving alternative by enabling collaborative model training without sharing raw data. This paper presents a comprehensive, real-world demonstration of a FL framework for PV power forecasting. We deployed a system using five prosumers from Madeira Island, with Raspberry Pi devices for local data collection and dedicated cloud servers for federated training. The system incorporates an automated pipeline for real-time inference, generating 24-hour forecasts with a sample rate of 15 minutes. Results from offline evaluations showed that the model achieved reasonable accuracy, with an average Normalised Root Mean Squared Error (NRMSE) of 0.08 and values reaching as low as 0.06. In the live deployment, real-time performance was slightly worse (NRMSE of 0.12), highlighting practical challenges. A relevant finding is the significant impact of data source consistency: using a different weather data source for inference than for training led to measurable performance degradation. This work validates the feasibility of FL for privacy-preserving energy forecasting and highlights the critical importance of maintaining consistent data pipelines in real-world deployments.

**Keywords:** Federated Learning · PV Power Forecasting · Smart Grids · Real-World Demonstration · Distributed Energy Resources.

## 1 Introduction

The increasing integration of Renewable Energy Sources (RESs), particularly distributed Solar Photovoltaics (PV) systems, is transforming modern power grids. While this transition is crucial for sustainability, it introduces significant

challenges in grid management due to the intermittent and decentralized nature of solar power generation [18]. Accurate forecasting of PV output is paramount for grid operators to ensure stability, optimize energy dispatch, and manage Distributed Energy Resources (DERs) effectively. However, traditional ML models for forecasting typically require centralizing large amounts of historical data from individual prosumers, which raises substantial privacy and data ownership concerns [8]. Consequently, there is a pressing need for a collaborative approach to PV forecasting in residential settings that can improve accuracy by leveraging data from multiple households, without compromising the privacy of individual prosumers [1, 6].

In this regard, Federated Learning (FL) has emerged as a powerful paradigm to address this problem. As a decentralized ML approach, FL allows multiple parties to collaboratively train a shared prediction model without exchanging their raw local data [11]. Instead, each participant trains a model on their own data, and only the resulting model parameters are sent to a central server for aggregation. This process enables the creation of a robust global model that learns from diverse datasets while preserving privacy [12].

While the principles of FL are well-established, comprehensive end-to-end demonstrations in the energy sector remain limited [3, 21]. This paper aims to bridge this gap by providing a detailed account of the design, implementation, and evaluation of a real-world FL system for PV power forecasting. The key contributions are as follows:

– The design and real-world deployment of a complete FL system, from data acquisition on edge devices (Raspberry Pi) to cloud-based federated training and real-time inference.
– A comprehensive performance analysis in both offline and live operational settings, providing insights into real-world effectiveness.
– A comparative analysis of forecast accuracy when using different weather data sources for training versus inference, quantifying the impact of data pipeline consistency.

This paper is organized as follows. Section 2 reviews related work in PV federated learning forecasting. Section 3 details the system design and methodology, including the forecasting model, the FL framework, and the forecasting algorithm. Section 4 describes the experimental setup, specifying the datasets and evaluation metrics. Section 5 presents and discusses the results from both offline and real-time evaluations. Finally, Section 6 concludes the paper and suggests directions for future work.

## 2   Related Works

FL is being actively explored in energy domains to forecast solar power and manage smart grids. For instance, Misra et al. proposed a framework to address the critical challenge of data imbalance in distributed load forecasting, demonstrating improved accuracy on real-world smart meter data while preserving

user privacy as the model was trained on-site, sharing only model updates with a central server for aggregation [12]. This star-topology architecture (clients ↔ server) is common. Riedel et al. [19] report a federated LSTM/GRU model for residential PV feed-in forecasting, achieving high accuracy while using differential privacy to protect household patterns. These works primarily use standard aggregation algorithms like Federated Averaging (FedAvg). More advanced approaches tackle the challenge of data heterogeneity. The FedCCL framework, for instance, uses location-based clustering before training to improve accuracy and stability [10].

Deploying FL systems from controlled simulations to real-world applications introduces significant practical hurdles. Experience from pioneering deployments in mobile systems and healthcare has highlighted key obstacles that must be overcome for a successful implementation [8, 20]. These well-documented challenges are broadly applicable and include statistical heterogeneity of data (Non-IID data), systems heterogeneity across clients, communication bottlenecks, and ensuring fairness for all participants [8]. In the context of residential PV forecasting, these issues are particularly pronounced due to vast differences in household PV installation sizes, panel orientation, and localized microclimates.

In summary, while many studies have explored FL for energy forecasting via simulation, comprehensive, end-to-end demonstrations that address personalization, fairness, and communication efficiency in a real-world setting are rare. This paper helps fill that gap by documenting the infrastructure, processes, and practical challenges of an end-to-end FL system for PV power forecasting, offering a valuable case study.

## 3   System Design and Methodology

This section presents the system architecture, data processing pipeline, the federated learning methodology used in this work, and the implemented privacy and security mechanisms.

### 3.1   System Architecture

We deployed a testbed using five prosumers from the Madeira Island Prosumers dataset [17]. The architecture is designed to handle real-world constraints, such as the limited computational power of edge devices.

– **Edge Layer:** Each prosumer's residence is equipped with a Raspberry Pi 3 Model B device (featuring an ARMv7 processor and 1GB of RAM) responsible for collecting and storing local PV operational data at 1-second intervals [18].
– **Client Server Layer:** Due to the 32-bit architecture of the Raspberry Pis being incompatible with the required 64-bit Python packages, model training is not performed at the extreme edge. Instead, data from the Raspberry Pis is periodically synchronized with five dedicated client servers hosted on the

Cloud, one for each prosumer. These 64-bit servers (each with an Arm64 processor and 4GB of RAM) execute the local model training.
– **Central Server Layer:** A sixth cloud server acts as the central Federated Learning server. It coordinates the training process, distributes the global model, and aggregates incoming model updates from the client servers.

This hybrid architecture is depicted in Figure 1. A fully automated pipeline was also developed for real-time inference, as shown in Figure 2. This pipeline handles real-time PV production data syncing, daily weather forecast downloads, and hourly forecast generation.
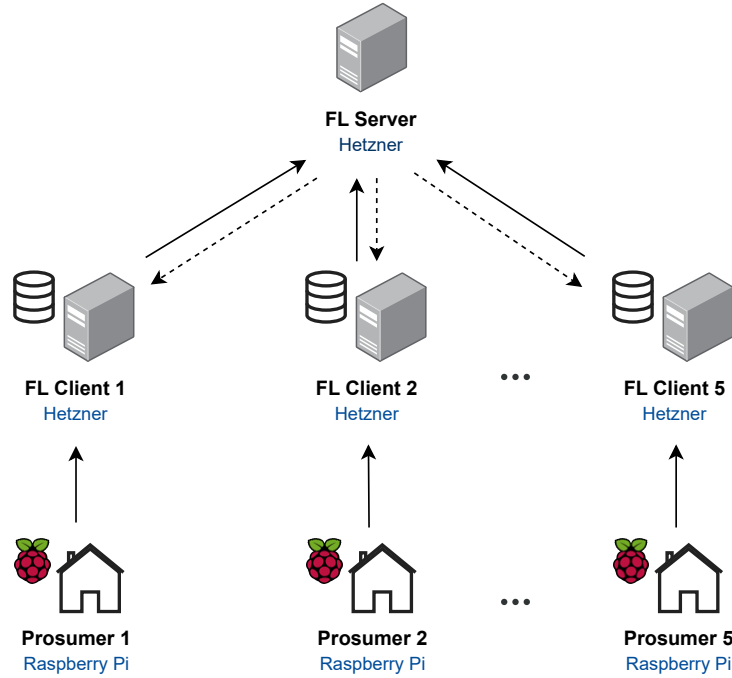


Fig. 1: Overview of the federated learning infrastructure, showing data flow from edge Raspberry Pi devices to dedicated cloud client servers for training, coordinated by a central FL server.
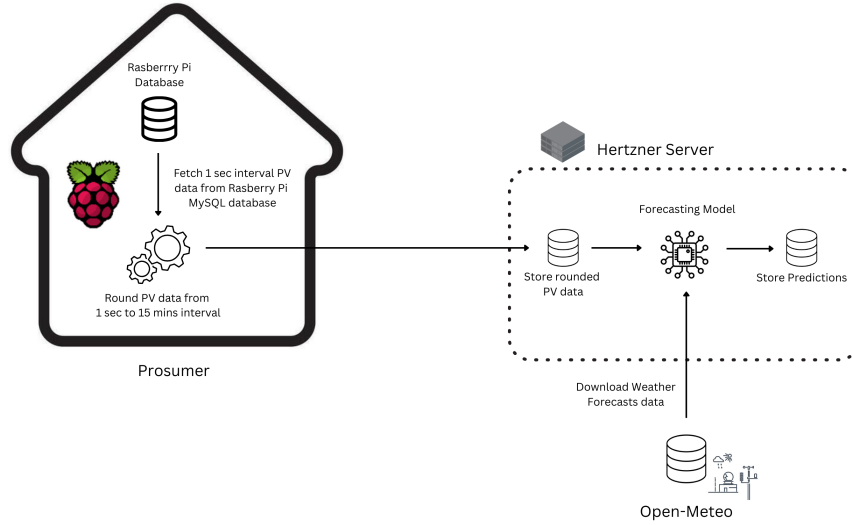
Fig. 2: The real-time inference architecture. Data is synced from the edge, combined with weather forecasts, and fed into the trained model on the client server to generate and store predictions.

## 3.2   Data Collection and Preprocessing

Data preprocessing and feature engineering are performed locally on each client server to preserve privacy. To ensure consistency and effective aggregation, standardized protocols were established.

- **Data Collection:** Raw PV power data is collected at 1-second intervals at the edge. This data was aggregated into 15-minute mean values to match the temporal resolution of the weather data, which was compiled from two distinct sources using the approximate coordinates of each prosumer. For model training, we used historical solar irradiation data from the Copernicus dataset [5], which offers high resolution of at least one-minute intervals for up to two days prior. For real-time inference, we used solar irradiance forecasts from the Open-Meteo API [14], which provides both the upcoming day and the two preceding days' forecasts at 15-minute resolution.
- **Feature Set:** A common set of input features was used by all clients, comprising two days of historical PV power and Global Horizontal Irradiation (GHI), along with one day of forecasted GHI and calendar features (day of year, week, hour, season).
- **Preprocessing Steps:** All clients applied a common preprocessing procedure consisting of min–max normalization and input sequence formation. For real-time inference, a key step was the conversion of weather forecast data.

The model was trained on historical solar *irradiation* ($Wh/m^2$), whereas the forecasted data is provided as solar *irradiance* ($W/m^2$). To enable the use of forecasted values as model inputs, we convert irradiance to irradiation using Equation (1).

$$Irradiation(Wh/m^2) = \frac{Irradiance(W/m^2)}{4} \tag{1}$$

### 3.3 Federated Learning Methodology

The FL process was implemented using the platform developed within the ALAMO project [15], which provided the infrastructure for orchestrating client-server interactions, managing data pipelines, and executing training tasks across distributed edge devices.

– **Forecasting Model:** The base algorithm employed is FPSeq2Q, a probabilistic sequence-to-quantile model based on Quantile Regression (QR) [7]. For this demonstration, we focused on deterministic forecasting by averaging the predicted quantiles to generate a single point estimate.
– **Flower Framework:** The FL process was executed using the Flower deployment engine [4], which manages orchestration between the clients and the central server.
– **FL Algorithm:** The federation was carried out over 15 communication rounds, using the standard FedAvg aggregation algorithm. In each round, two randomly selected clients participated in local training. Selecting two clients per round provided a balance between computational efficiency and statistical diversity in the aggregated updates, while maintaining manageable communication costs.
– **Model Training:** Local training at each client was configured to run for 50 epochs per round. This number was determined to be sufficient for substantial local model improvement while mitigating the risk of client drift, where local models diverge excessively from the global objective.

### 3.4 Privacy and Security Considerations

The core FL framework inherently preserves privacy by keeping raw data localized on each client's server. Data never leaves the client's control. To further enhance system security, two layers of encrypted communication were implemented:

– **Local Data Synchronization:** Data transfers from the edge device (Raspberry Pi) to the client server were secured via SSH tunnels using shared public keys, preventing unauthorized access during synchronization.
– **Federated Communication:** All interactions between client servers and the central FL server were protected using Transport Layer Security (TLS), as provided by the Flower framework built on gRPC. TLS ensures end-to-end encryption, maintaining the confidentiality and integrity of all model updates and metadata during transmission.

Together, these security mechanisms provide robust protection against threats like Man-in-the-Middle (MitM) attacks and eavesdropping, thereby establishing the fundamental security posture required for trustworthy, privacy-respecting collaboration.

## 4  Experimental Setup

This section outlines the case study and evaluation strategy used to validate the proposed framework.

### 4.1  The 5-Household Case Study

The demonstration was conducted with five prosumers from Madeira Island, Portugal. These prosumers exhibit heterogeneity in their PV installations and building types, offering a realistic and diverse testbed for evaluating the FL model. Detailed characteristics of each prosumer are presented in Table 1. Note that for Prosumer 1, who upgraded its PV installation during the data collection, PV power data was scaled during evaluation to match the capacity used during training. The training and offline evaluation datasets are summarized in Table 2, with time periods selected based on data availability, prioritizing one full year of consecutive data points.

Table 1: Details of the prosumers used in the real-world demonstration.

| ID | Type | Historical Data | Peak PV (kWp) |
|---|---|---|---|
| 1 | House | 23/06/2018 - 31/12/2023 | 1.50[a] |
| 2 | House | 01/05/2018 - 01/03/2022 | 2.70 |
| 3 | House | 22/11/2019 - 21/12/2023 | 0.75 |
| 4 | Apartment | 10/07/2018 - 31/12/2023 | 0.50 |
| 5 | Apartment | 10/07/2018 - 31/12/2023 | 0.50 |

[a]This prosumer upgraded the installation to 2.25kWp after 2019.

Table 2: Data periods used for training and offline evaluation.

| ID | Training Period | Offline Evaluation Period |
|---|---|---|
| 1 | 01/01/2019 - 31/12/2019 | 01/01/2022 - 31/12/2022 |
| 2 | 01/01/2019 - 31/12/2019 | 01/01/2021 - 31/12/2021 |
| 3 | 01/01/2020 - 31/12/2020 | 01/01/2023 - 31/12/2023 |
| 4 | 01/01/2019 - 31/12/2019 | 01/01/2023 - 31/12/2023 |
| 5 | 10/07/2018 - 20/04/2019 | 11/08/2022 - 28/06/2023 |

## 4.2   Performance Evaluation Metrics

To evaluate the model's predictive accuracy, we selected two metrics widely used in PV forecasting literature: the Root Mean Squared Error (RMSE) and the Normalised Root Mean Squared Error (NRMSE) [2]. The RMSE is a standard regression metric that effectively penalizes large prediction errors, which are particularly undesirable in grid management and energy scheduling applications.

To enable a fair and direct comparison of forecasting performance across households with different PV array sizes, we use the NRMSE. This metric is calculated by normalizing the RMSE of each household by its installed peak power ($P_{peak}$) [13]. For our analysis, both metrics are computed for each 24-hour forecast period, and aggregated results (e.g., per season or per week) are then reported.

## 5   Results and Discussion

This section reports offline and real-time results for the five-household case study. All data and analysis scripts are publicly available on a Open Sciente Framework (OSF) repository [16] to ensure reproducibility.

### 5.1   Offline Operation

**Offline Predictive Performance:** The FL forecasting model was initially evaluated offline using historical data from each prosumer.

As summarized in Table 3, the model's performance varied across the five prosumers. Prosumer 1 demonstrated the best performance, with the lowest NRMSE value of 0.06 ±0.02. Conversely, Prosumers 4 and 5 exhibited the highest NRMSE, both with values of 0.09 ±0.03.

Table 3: Total Performance Results for Offline Evaluation of Prosumers.

| ID | Total | |
| --- | --- | --- |
| | **RMSE** | **NRMSE** |
| 1 | 139.13 ±40.22 | 0.06 ±0.02 |
| 2 | 203.99 ±65.02 | 0.08 ±0.02 |
| 3 | 63.05 ±20.53 | 0.08 ±0.03 |
| 4 | 38.26 ±14.83 | 0.09 ±0.03 |
| 5 | 45.70 ±14.74 | 0.09 ±0.03 |

Table 4 and Figure 3 present a detailed comparison of the forecasting performance across seasons, highlighting the distinct data characteristics of each prosumer.

Forecasting performance varies across seasons and prosumers, with clear differences highlighted by the seasonal RMSE and NRMSE results. Prosumers 2,

3, and 5 exhibit higher errors during winter, consistent with increased weather variability and lower irradiance in this season. For example, Prosumer 2 shows the highest RMSE and NRMSE in winter (226.5 ± 72 and 0.08 ± 0.03), while Prosumer 3's error is also notably elevated (75.2 ± 23 and 0.10 ± 0.03).

Prosumer 5 maintains relatively high errors consistently across all seasons (NRMSE of 0.09), likely reflecting local factors such as shading that make it harder to predict. In contrast, Prosumer 1 consistently has the lowest errors across all seasons (NRMSE around 0.06 to 0.07), indicating it is the easiest to forecast.

Table 4: Seasonal Performance Results for Offline Evaluation of Prosumers.

| ID | Winter | | Spring | | Summer | | Autumn | |
|---|---|---|---|---|---|---|---|---|
| | RMSE | NRMSE | RMSE | NRMSE | RMSE | NRMSE | RMSE | NRMSE |
| 1 | 127.69 ±45.14 | 0.06 ±0.02 | 136.96 ±44.10 | 0.06 ±0.02 | 142.75 ±33.58 | 0.06 ±0.01 | 149.14 ±38.04 | 0.07 ±0.02 |
| 2 | 226.51 ±71.97 | 0.08 ±0.03 | 215.01 ±69.42 | 0.08 ±0.03 | 182.08 ±62.60 | 0.07 ±0.02 | 192.38 ±56.07 | 0.07 ±0.02 |
| 3 | 75.17 ±23.42 | 0.10 ±0.03 | 58.24 ±17.30 | 0.08 ±0.02 | 52.20 ±19.80 | 0.07 ±0.03 | 66.60 ±21.60 | 0.09 ±0.03 |
| 4 | 42.04 ±13.41 | 0.08 ±0.03 | 41.14 ±19.68 | 0.08 ±0.04 | 44.24 ±12.73 | 0.09 ±0.03 | 25.62 ±13.51 | 0.09 ±0.03 |
| 5 | 46.14 ±15.12 | 0.09 ±0.03 | 45.34 ±17.14 | 0.09 ±0.03 | 45.40 ±12.71 | 0.09 ±0.03 | 45.90 ±13.99 | 0.09 ±0.03 |
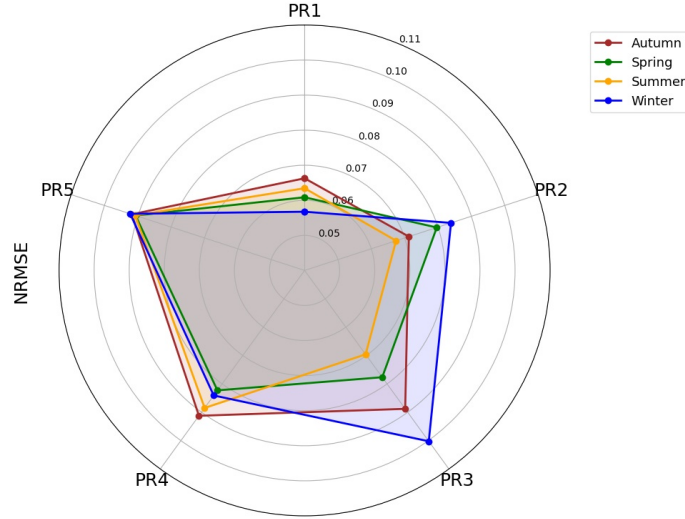


Fig. 3: Visual Comparison of NRMSE across Seasons for Prosumers.

**Communication and Computational Costs:** The deployment highlighted practical system constraints and communication overheads. The decision to of-

fload computationally intensive model training from the 32-bit Raspberry Pi devices to 64-bit cloud servers was critical for feasibility, establishing a tiered architecture.

The entire federated training process, encompassing 15 communication rounds, had a total duration of approximately 8 hours and 15 minutes. During each round, participating clients trained their local models for an average of 49 minutes. The heterogeneity in client participation was notable, with prosumers contributing varying numbers of local training rounds to the global model: Prosumer 1 participated 5 times, Prosumer 2 for 8 times, Prosumer 3 for 8 times, Prosumer 4 for 3 times, and Prosumer 5 for 6 times. After each round of training, the model weights were transferred between the client and the central server. The model's weights data for most cases were around 9 Megabytes.

Regarding data transfer, each Raspberry Pi (edge device) synchronized its local PV measurements with its respective client server hourly. Given that 15-minute mean power values were used, this translated to approximately 4 measurements per hour, plus associated timestamps and identifiers. While the exact payload size varies, this is a relatively small amount of data, as it is in the order of a few kilobytes per hour. Weather data, crucial for forecasting, was acquired daily from external APIs (Copernicus and Open-Meteo). These API calls, typically returning JSON data, also incurred relatively low communication overhead, usually in the range of tens of kilobytes per daily forecast. The communication overhead due to the SSH tunnels (for edge-to-client data synchronization) and TLS encryption (for federated communication via gRPC) is considered negligible compared to the data payload itself, primarily adding cryptographic handshake and header bytes.

### 5.2   Real-Time Operation

**Real-Time Predictive Performance:** During the project's development, several Raspberry Pis became inaccessible due to hardware or network issues. Consequently, real-time results are limited to prosumers 4 and 5, evaluated from late March to late April 2025. The results are presented in Table 5.

Overall, the real-time errors are higher than the offline errors, as exemplified by Prosumer 4, where the RMSE increases from 41.14 $\pm$ 19.68 W in Spring (offline) to 73.19 $\pm$ 25.64 W (real-time). To help understand these results, the predictions for both prosumers are plotted in Figure 4 and Figure 5.

The predicted PV power closely follows the general daily trends of the actual production, but it often underestimates the output, particularly during peak hours. This bias can be attributed to the nature of the GHI input data, which tends to smooth out irradiance peaks and therefore limits the model's ability to reflect the full dynamic range of solar production. On days with unusually low production, likely due to heavy cloud cover or poor weather, the model instead overestimates the output, revealing a limited sensitivity to such typical conditions. Additionally, the model struggles to capture short-term fluctuations, especially those caused by passing clouds, resulting in a mismatch between actual and predicted power during partially cloudy periods.

Table 5: Results for the Real-time Deployment.

| ID | Week | RMSE | NRMSE | Total | |
|----|------|------|-------|-------|---|
|    |      |      |       | **RMSE** | **NRMSE** |
| 4 | 1 | 73.22 ±15.27 | 0.15 ±0.03 | | |
|   | 2 | 46.86 ±6.88 | 0.09 ±0.01 | | |
|   | 3 | 77.06 ±27.58 | 0.15 ±0.06 | 73.19 ±25.64 | 0.15 ±0.05 |
|   | 4 | 85.24 ±28.02 | 0.17 ±0.06 | | |
|   | 5 | 82.48 ±50.45 | 0.16 ±0.10 | | |
| 5 | 1 | 71.75 ±9.41 | 0.14 ±0.02 | | |
|   | 2 | 54.49 ±17.07 | 0.11 ±0.03 | | |
|   | 3 | 73.01 ±21.05 | 0.15 ±0.04 | 74.96 ±28.72 | 0.15 ±0.06 |
|   | 4 | 87.40 ±28.03 | 0.17 ±0.06 | | |
|   | 5 | 91.99 ±68.06 | 0.18 ±0.14 | | |

Ultimately, the model's accuracy hinges on the quality of the forecasted GHI data. As illustrated in Figure 6, the GHI inputs themselves fail to reflect rapid variations, which propagate through to the PV predictions. These limitations suggest that improving GHI forecasting, or incorporating higher-resolution or hybrid data sources, could substantially enhance model performance.

Moreover, another factor contributing to these issues may be the use of different GHI sources during training and inference, which introduces inconsistencies in the learned irradiance–power relationship. This is addressed in the next subsection.
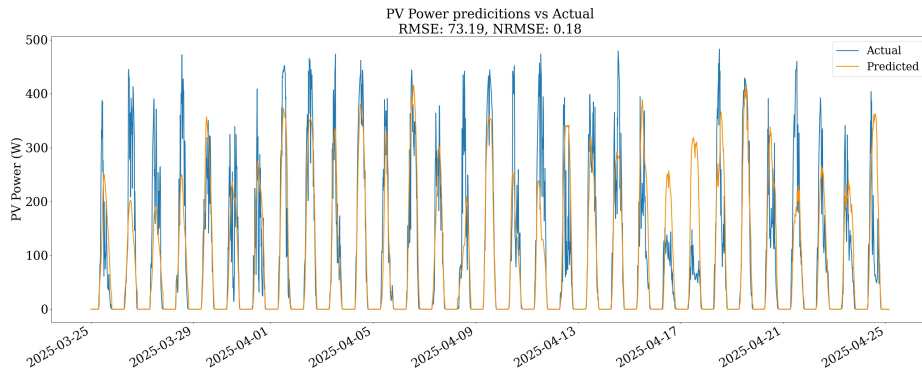


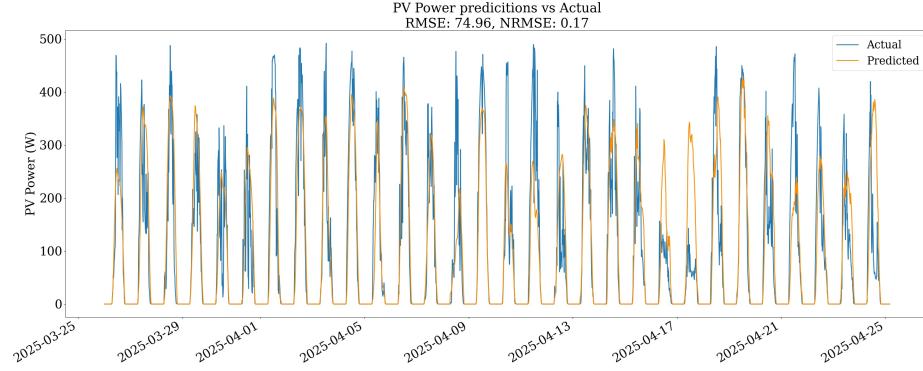Fig. 4: Visualization for the Real-time Deployment of Prosumer 4 (25/03/2025 - 26/04/2025).

PV Power predicitions vs Actual
RMSE: 74.96, NRMSE: 0.17

Fig. 5: Visualization for the Real-time Deployment of Prosumer 5 (26/03/2025 - 26/04/2025).



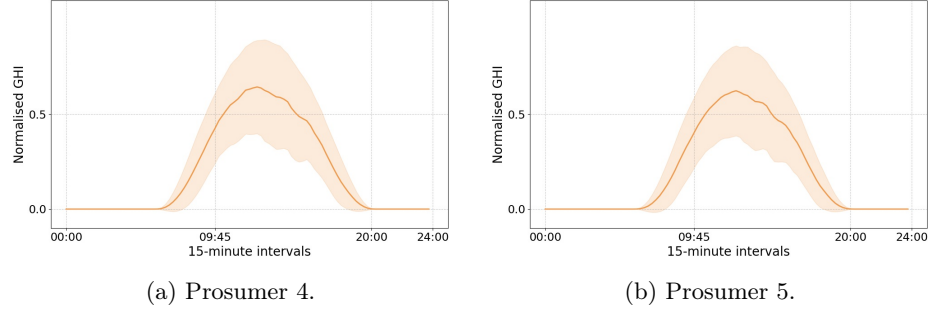(a) Prosumer 4.                          (b) Prosumer 5.

Fig. 6: Normalised GHI 15-minute intervals trends.

**Impact of Weather Data Sources on Forecast Accuracy:**  To quantify the performance impact of mismatched weather data sources between training and inference, we conducted an offline experiment. We compared two sets of model predictions for the same evaluation period: one generated using weather data from Copernicus (the source for historical training data) and the other using data from Open-Meteo (the source used for real-time inference).

 Figures 7 and 8 illustrate that the model using Copernicus GHI (magenta) better tracks the actual PV power (cyan), especially during periods of variable solar irradiance. In contrast, predictions based on Open-Meteo (black) are consistently lower and smoother, failing to capture rapid fluctuations caused by passing clouds.

These visual differences are reflected in the aggregate performance metrics, summarized per weekly performance in Tables 6 and 7. For both prosumers, Copernicus consistently outperforms Open-Meteo across all five weeks, with lower average RMSE and NRMSE values. For instance, in the case of Prosumer 4, Copernicus yields a total RMSE of 55.05±18.67 and NRMSE of 0.11±0.04, compared to 73.19±25.64 and 0.15±0.05 with Open-Meteo. Simi-
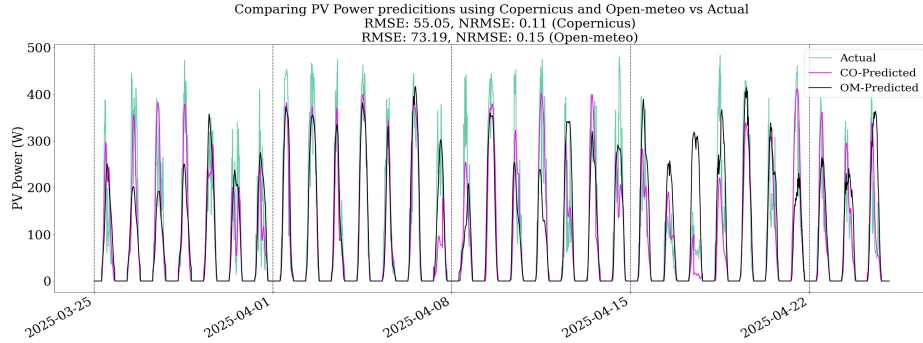
Fig. 7: Visualization for the Predictions made using Copernicus (CO-predicted) Vs. Open-Meteo (OM-predicted) for Prosumer 4.
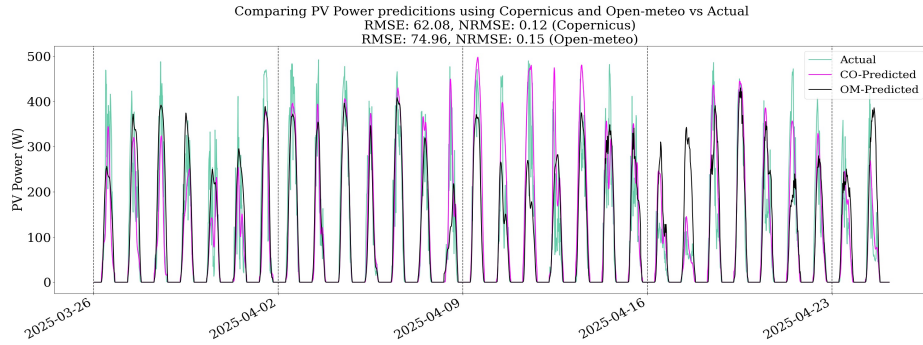


Fig. 8: Visualization for the Predictions made using Copernicus (CO-predicted) Vs. Open-Meteo (OM-predicted) for Prosumer 5.

lar trends are observed for Prosumer 5, where Copernicus achieves 62.08±17.25 RMSE and 0.13±0.03 NRMSE, while Open-Meteo reaches 74.96±28.72 RMSE and 0.15±0.06 NRMSE.

These results confirm that weather data characteristics significantly influence forecasting performance. Still, the superior performance observed with Copernicus is likely due primarily to the model having been trained on Copernicus data and evaluated on the same source; furthermore, since Copernicus provides reanalysis or observed values rather than actual forecasts, the inputs more closely reflect real conditions, particularly during periods of high variability, which further contributes to improved accuracy. Notably, the variability in performance, as captured by the standard deviations, is higher with Open-Meteo, which suggests less stable input conditions that translate into less reliable predictions.

Ultimately, these results suggest that models learn patterns and biases specific to their training data. Introducing a new data source at inference, even after careful unit conversion, can cause performance to degrade. This is a critical lesson for real-world machine learning deployments.

Table 6: PV Power Predictions Using Copernicus vs. Open-Meteo (Prosumer 4).

| Data Source | Week | RMSE | NRMSE | Total | |
|---|---|---|---|---|---|
| | | | | RMSE | NRMSE |
| Open-Meteo | 1 | 73.22 ±15.27 | 0.15 ±0.03 | 73.19 ±25.64 | 0.15 ±0.05 |
| | 2 | 46.86 ±6.88 | 0.09 ±0.01 | | |
| | 3 | 77.06 ±27.58 | 0.15 ±0.06 | | |
| | 4 | 85.24 ±28.02 | 0.17 ±0.06 | | |
| | 5 | 82.48 ±50.45 | 0.16 ±0.10 | | |
| Copernicus | 1 | 54.90 ±14.38 | 0.11 ±0.03 | 55.05 ±18.67 | 0.11 ±0.04 |
| | 2 | 54.23 ±21.84 | 0.11 ±0.04 | | |
| | 3 | 58.92 ±14.62 | 0.12 ±0.03 | | |
| | 4 | 56.58 ±19.53 | 0.11 ±0.04 | | |
| | 5 | 43.59 ±22.99 | 0.09 ±0.05 | | |

Table 7: PV Power Predictions Using Copernicus vs. Open-Meteo (Prosumer 5).

| Data Source | Week | RMSE | NRMSE | Total | |
|---|---|---|---|---|---|
| | | | | RMSE | NRMSE |
| Open-Meteo | 1 | 71.75 ±9.41 | 0.14 ±0.02 | 74.96 ±28.72 | 0.15 ±0.06 |
| | 2 | 54.49 ±17.07 | 0.11 ±0.03 | | |
| | 3 | 73.01 ±21.05 | 0.15 ±0.04 | | |
| | 4 | 87.40 ±28.03 | 0.17 ±0.06 | | |
| | 5 | 91.99 ±68.06 | 0.18 ±0.14 | | |
| Copernicus | 1 | 66.11 ±16.54 | 0.13 ±0.03 | 62.08 ±17.25 | 0.13 ±0.03 |
| | 2 | 54.47 ±7.21 | 0.11 ±0.01 | | |
| | 3 | 72.92 ±10.48 | 0.15 ±0.02 | | |
| | 4 | 57.07 ±19.38 | 0.11 ±0.04 | | |
| | 5 | 54.17 ±32.64 | 0.11 ±0.07 | | |

## 6   Conclusion, Limitations and Future Work

This paper has presented a successful end-to-end deployment of a real-time, cloud-based FL framework for residential PV power forecasting. By seamlessly integrating data acquisition from five prosumers via edge devices with a scalable cloud-based training and inference pipeline, we have demonstrated that FL is a practical, privacy-preserving, and robust approach for collaborative forecasting across distributed participants.

Our results confirm that federated models can achieve competitive predictive accuracy, comparable to state-of-the-art day-ahead PV forecasting approaches that rely on centralized data (e.g., [9,22]), while respecting user data privacy by avoiding centralized data aggregation. To the best of our knowledge, this study is among the first to report a real-world implementation of FL-based residential PV

power forecasting, bridging the gap between theoretical research and practical deployment.

Nonetheless, the study faces important limitations that open paths for future work. The limited number of prosumers constrains the generalizability of our findings, motivating future evaluations on larger, more heterogeneous cohorts. Real-time evaluations were considerably affected by edge hardware failures, highlighting the need for more robust deployment environments. Moreover, while different weather data sources were compared, domain adaptation techniques to address distribution shifts remain unexplored and represent a promising research direction.

Looking ahead, this work lays a foundation for advancing FL research in energy forecasting through several avenues. Periodic federated retraining can help the system adapt to evolving conditions and mitigate concept drift. More sophisticated FL algorithms that explicitly model inter-client relationships and heterogeneity may enhance robustness and personalization. Integrating personalized PV forecasts into home energy management systems to optimize battery operation or flexible load scheduling would unlock the practical value of privacy-preserving predictions. Further investigations into communication overheads, operation on diverse edge hardware, comparisons with centralized approaches, and the influence of forecast inputs such as GHI will deepen understanding and improve system design.

# References

1. Al-Shorbaji, A., et al.: Data sharing, privacy and security considerations in the energy sector: A review from technical landscape to regulatory specifics. arXiv preprint arXiv:2403.02476 (2024)
2. Antonanzas, J., et al.: Review of solar power forecasting. Solar Energy **136**, 78–109 (2016)
3. Ariff, M.F.M., et al.: Advancing power system services with privacy-preserving federated learning techniques: A review. UPCommons (2024), [Online]. Available: `https://upcommons.upc.edu/handle/2117/402809`
4. Beutel, D.J., Topal, T., Mathur, A., Qiu, X., Fernandez-Marques, J., Gao, Y., Sani, L., Kwing, H.L., Parcollet, T., Gusmão, P.P.d., Lane, N.D.: Flower: A friendly federated learning research framework. arXiv preprint arXiv:2007.14390 (2020)
5. Copernicus Atmosphere Monitoring Service (CAMS): CAMS solar radiation time-series (Feb 2020), `https://ads.atmosphere.copernicus.eu/datasets/cams-solar-radiation-timeseries`, accessed on 04 Jannuary 2025

6. EnergyREV Consortium: Privacy and data sharing in smart local energy systems: insights and recommendations. Tech. rep., EnergyREV (2020), `https://www.energyrev.org.uk/media/1586/privacy-and-data-sharing-in-sles.pdf`
7. Faustine, A., Pereira, L.: Fpseq2q: Fully parameterized sequence to quantile regression for net-load forecasting with uncertainty estimates. IEEE Transactions on Smart Grid **13**(3), 2440–2451 (2022). `https://doi.org/10.1109/TSG.2022.3148699`
8. Kairouz, A., et al.: Advances and open problems in federated learning. Foundations and Trends in Machine Learning **14**(1–2), 1–210 (2021)
9. Kumar, A., Rizwan, M., Nangia, U.: A Hybrid Intelligent Approach for Solar Photovoltaic Power Forecasting: Impact of Aerosol Data. Arabian Journal for Science and Engineering **45**(3), 1715–1732 (Mar 2020). `https://doi.org/10.1007/s13369-019-04183-0`
10. Lee, H., Yoon, J., Choi, S.: Fedccl: Federated learning with clustered client learning for solar power forecasting. Applied Energy **322**, 119492 (2022)
11. McMahan, H.B., et al.: Communication-efficient learning of deep networks from decentralized data. In: Proc. AISTATS (2017)
12. Misra, S., Ojha, T., Sakor, A.: Federated learning for distributed load forecasting: Addressing data imbalance in smart grids. IEEE Transactions on Industrial Informatics (2025)
13. Nespoli, L., et al.: A deep learning-based model for photovoltaic power forecasting. Applied Sciences **12**(10), 4887 (2022)
14. Open-Meteo: Open-Meteo weather forecast api, `https://open-meteo.com/`, accessed on 10 February 2025
15. Pereira, L., Nair, V., Annaswamy, A., Dias, B., Morais, H.: Accurate federated learning with uncertainty quantification for distributed energy resource forecasting applied to smart grids planning and operation: The alamo vision. In: IET Conference Proceedings. vol. 2024, pp. 1123–1126 (2024)
16. Pereira, L., Monteiro, D., Apina, F.: Supplementary Material for: A Real-World Deployment of Federated Learning for Residential Solar PV Power Forecasting (Aug 2025). `https://doi.org/10.17605/OSF.IO/J6NG5`
17. Prsma, M-ITI, EEM, ACIF-CCIM: Installation report of dsm demo (m36 version). Tech. Rep. 4.7, EUROPEAN COMMISSION, Funchal, Portugal (2020), `https://ec.europa.eu/research/participants/documents/downloadPublic?documentIds=080166e5cebf3967&appId=PPGMS`
18. Quintal, F., Garigali, D., Vasconcelos, D., Cavaleiro, J., Santos, W., Pereira, L.: Energy monitoring in the wild: Platform development and lessons learned from a real-world demonstrator. Energies **14**(18), 5786 (2021). `https://doi.org/10.3390/en14185786`
19. Riedel, T., Falkner, M., Fischer, A., Moslehi, K.: Residential load and pv feed-in forecasting using federated learning. IEEE Transactions on Smart Grid **15**(2), 1450–1462 (2024)
20. Rieke, N., et al.: The future of digital health with federated learning. NPJ Digital Medicine **3**(1), 119 (2020)
21. Savio, D.A., et al.: Federated learning for the power grid: A survey. arXiv preprint arXiv:2303.12093 (2023)
22. Theocharides, S., Theristis, M., Makrides, G., Kynigos, M., Spanias, C., Georghiou, G.E.: Comparative Analysis of Machine Learning Models for Day-Ahead Photovoltaic Power Production Forecasting. Energies **14**(4), 1081 (Jan 2021). `https://doi.org/10.3390/en14041081`