

ORIGINAL RESEARCH

Open access

A Data-Driven Evaluation of Federated Learning vs. Model Personalization for Electric Load Prediction

Mohamed Salah^{1*}, Youssef Karim¹, Ahmed Nabil², Mahmoud Adel¹, Karim Hassan²

Abstract

Accurate short-term forecasting of household electricity consumption is vital to contemporary distributed energy networks. Most approaches to load prediction rely on deep neural architectures and require access to confidential, user-specific consumption records during training. Traditional neural network training typically collects all available data at a single central location to develop one unified model. Nevertheless, pooling individual user information in this manner raises serious concerns regarding data security and personal privacy. To address these issues, the present research explores federated learning techniques combined with model personalization as promising alternatives. Through a comprehensive simulation-based evaluation, these approaches are benchmarked against the standard centralized training strategy and a simple pre-trained reference model. The analysis reveals that the intrinsic characteristics and heterogeneity of the underlying electricity consumption patterns significantly affect the resulting model error levels. Consequently, directly contrasting error distributions across methods largely reflects differences in data composition rather than true differences in algorithmic effectiveness. To overcome this limitation, the study introduces a novel evaluation technique termed “differential comparison.” This approach focuses exclusively on pairwise loss differences across identical data instances, thereby enabling a fairer assessment of relative model improvements via the resulting difference distributions. Applying this method demonstrates that personalized model training outperforms all other strategies examined for residential load prediction, achieving better results in 59.1% of the evaluated scenarios.

Keywords Federated learning, Machine learning, Model personalization, Temporal convolutional network, Electric load forecast, Differential comparison

*Correspondence:

Mohamed Salah
mohamed.salah@cu.edu.eg

¹ Department of Electrical Grid Systems, Faculty of Engineering, Cairo University, Cairo, Egypt

² Department of Energy Analytics and Control, Faculty of Engineering, Ain Shams University, Cairo, Egypt

Introduction

The large-scale deployment of smart metering systems is progressing rapidly for both residential and commercial electricity consumers as part of the global shift toward low-carbon energy infrastructures. In Switzerland, national policy under the ‘Energy Strategy 2050’ requires that 80% of final consumers be equipped with smart meters by the close of 2027 [1]. These advanced metering devices unlock extensive, high-resolution data streams that enhance grid observability, optimize system operations, and facilitate greater incorporation of variable renewable generation sources, thereby advancing worldwide efforts to mitigate climate change and achieve decarbonization goals.

A major application made feasible by smart meter proliferation is individual-level electricity demand forecasting. This task involves estimating future power draw for a single consumer or group of consumers. Precise demand predictions are

crucial for maintaining balance between supply and demand, thereby bolstering the stability and cost-effectiveness of power networks [2]. They also support strategies for handling peak loads, designing time-varying tariff structures, and delivering ancillary grid services [3].

Demand forecasting typically leverages state-of-the-art machine learning algorithms applied to historical time-series information, including previous consumption patterns, meteorological predictions, ambient temperature readings, and related variables. The fine-grained temporal and locational detail provided by these datasets enables sophisticated prediction techniques that forecast at the household level while simultaneously improving the accuracy of aggregated demand profiles [4, 5]. An illustrative example appears in **Figure 1**, where five preceding days of consumption data (shown in blue) serve as input to forecast the subsequent two days of demand (green), alongside the observed actual values (orange).

The specific requirements for load forecasting vary depending on its application, particularly with respect to the prediction time horizon [6]. Broadly, load forecasting approaches are categorized into several types [2]:

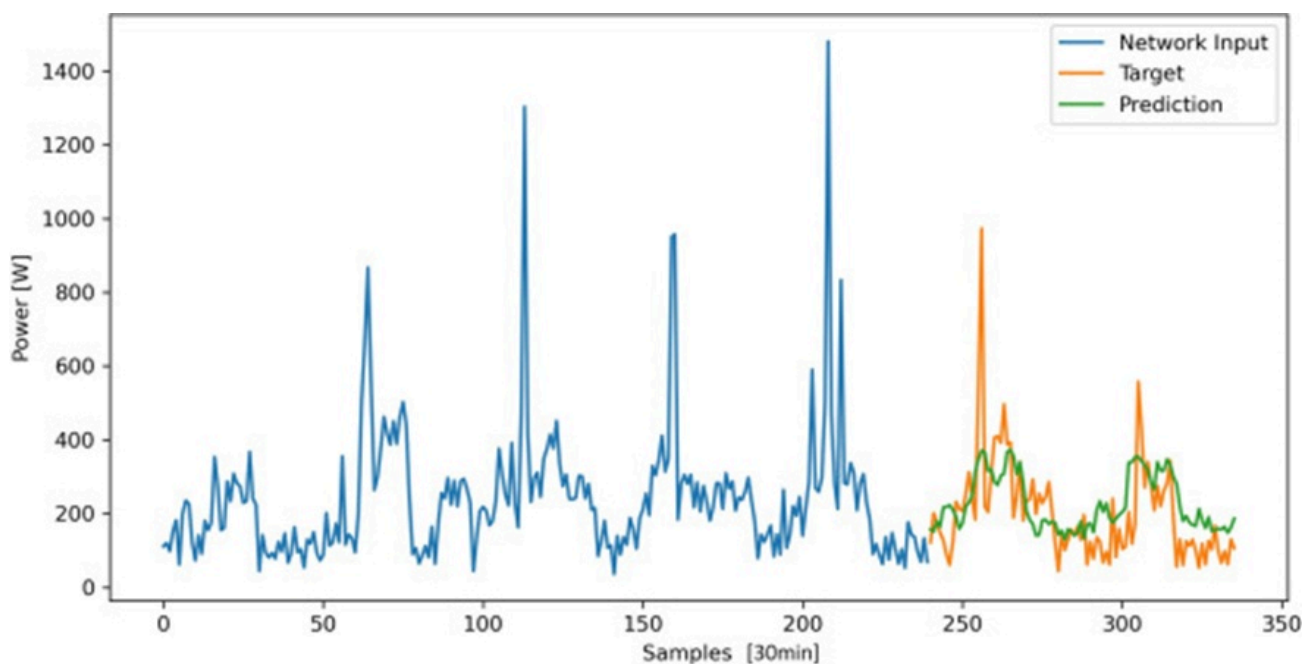


Figure 1. Illustration of short-term electricity demand prediction (Readers should consult the online version of this paper for the color key in the figure caption).

- Long-term forecasting (extending beyond one year)
- Medium-term forecasting (ranging from one week up to one year)
- Short-term forecasting (covering periods of up to one week)
- Very short-term forecasting (spanning less than 24 hours)

In most power system planning and real-time optimization tasks, forecasts with a one-day lookahead are essential [7]. For this reason, short-term load forecasting in electricity grid applications is commonly understood to mean day-ahead predictions.

Load forecasting methods can also be classified by voltage level and degree of data aggregation. Forecasts at the transmission and distribution levels (high- and medium-voltage) typically rely on aggregated load profiles from many customers. In contrast, forecasting at the low-voltage level—especially for individual households or end users—depends on disaggregated, customer-specific profiles. These single-user predictions exhibit much greater variability and are strongly influenced by each person's unique consumption habits [5].

Extensive recent research has focused on enhancing the accuracy of load profile predictions, both for aggregated groups and for individual consumers. Nonetheless, studies have consistently shown that conventional forecasting models optimized for aggregated demand often perform poorly when directly applied to single-household time series [5].

Main challenge I: Protecting data privacy

Standard procedures for processing smart meter readings involve gathering all data into a central repository for analysis and model building. This centralized approach creates substantial risks to both data security and individual privacy [8]. As privacy-preserving alternatives, federated learning (FL) and personalized model training (PS) have emerged. These techniques enable the development of neural network predictors using decentralized consumption time series without transferring raw user data to a central server, thereby complying with strict privacy frameworks such as the European Union's General Data Protection Regulation (GDPR).

Main challenge II: Heterogeneity across datasets

Although numerous investigations have evaluated the effectiveness of FL and PS approaches, most prior comparisons rely on overall distributions of error metrics computed with p-norm losses, such as mean absolute error (MAE). Such evaluations overlook the considerable structural and magnitude differences that exist among individual load profiles (i.e., across different end users). Because consumption patterns vary widely in shape, scale, and periodicity across households, a single global model struggles to adequately capture the full dataset's diverse characteristics [9]. As a result, error values can differ dramatically between models even when their underlying predictive quality is similar. When determining which training strategy performs best, this phenomenon risks leading to misleading conclusions driven by data heterogeneity rather than genuine differences in model capability.

Related Work

This section reviews earlier publications and empirical studies relevant to electricity load forecasting. It first outlines the current state-of-the-art (SOTA) approaches in neural network design for this domain, including their architectural features and core mechanisms. It then summarizes recent advances in decentralized and personalized training, including federated learning and model personalization.

Model architectures

In earlier decades, load forecasting relied primarily on classical machine learning algorithms, such as support vector machines (SVMs), decision trees (DT), and random forests (RFs). Advances in processing power, together with expanded availability of computational capacity (including edge devices), have driven a rapid shift toward neural network-based methods. These models have demonstrated strong potential to capture complex patterns in load time series and have therefore attracted growing attention in both research and practical development [4].

For sequence prediction tasks such as time series forecasting, contemporary neural architectures incorporate mechanisms explicitly designed to handle temporal dependencies, making it easier to detect cyclic, seasonal, or recurring behaviors and thereby boosting overall accuracy. This is most frequently accomplished through either convolutional layers or recurrent layer designs [10].

Several comparative studies have assessed various model families for load forecasting and are directly pertinent to the present research. Gasparin *et al.* [4], Bai *et al.* [10], and Kuo and Huang [11] examined conventional machine learning techniques (SVM, RF, DT, and fully connected neural networks—FCNN) alongside modern deep learning approaches, including long short-term memory networks (LSTM) and one-dimensional convolutional architectures such as the temporal convolutional network (TCN) introduced by Gasparin *et al.* [4] or the DeepEnergy architecture proposed by Kuo and Huang [11].

All three studies [4, 10, 11] highlight the strong performance of convolutional-based models for electricity demand prediction. Each independently reported that the TCN or DeepEnergy network outperformed competing architectures. Furthermore, Gasparin *et al.* [4] emphasized that the TCN not only delivers excellent forecasting results but also exhibits favorable training dynamics, being far less susceptible to vanishing or exploding gradients, which are common in recurrent neural networks. A schematic representation of the TCN structure is presented in Figure 2.

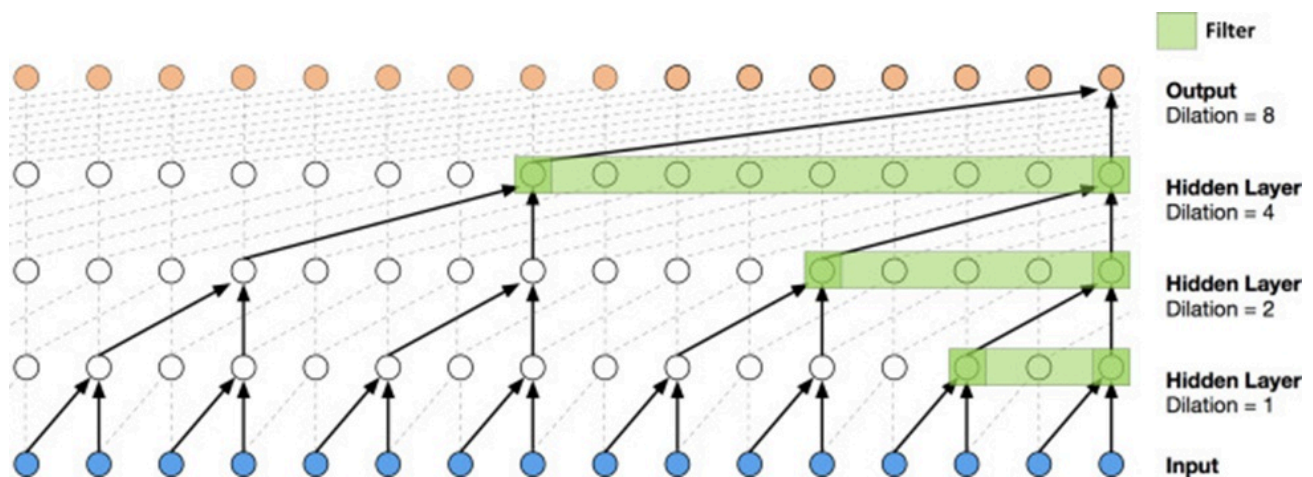


Figure 2. Diagram illustrating the layer structure of a temporal convolutional network (TCN) [12].

Federated learning

Several studies have explored techniques for developing predictive models using distributed datasets rather than consolidated ones, thereby making model training feasible under strict data protection requirements. The predominant technique examined in this body of work is federated learning (FL). In FL, a shared global model is refined by collecting only the updated model parameters (not the raw consumption records) from participating devices or clients. Each client performs local training steps using its own user-specific data. The locally computed parameter updates are then sent to a central coordinator, where they are combined through a simple per-parameter averaging procedure to produce an updated global model. This iterative cycle continues until the model stabilizes.

In the study by Taik and Cherkaoui [13], the application of FL to electricity demand forecasting was examined. The authors experimented with smart meter readings obtained from the Pecan Street Inc. Dataport repository [14], covering roughly 800 residential households. They assessed the effectiveness of both globally trained and individually adapted LSTM-based predictors within a federated framework. Their findings indicate that federated learning offers a viable path toward high-quality models, achieving comparable accuracy to fully centralized training while drastically lowering data transmission requirements and fully safeguarding the confidentiality of individual usage patterns.

Fekri *et al.* [9] evaluated two prominent FL variants—federated averaging (FedAvg) and federated stochastic gradient descent (FedSGD)—against a traditional centralized machine learning baseline. FedAvg aggregates model weights across clients to form the global update, while FedSGD aggregates client-computed gradients to perform a single central

gradient descent step. The comparison revealed that FedAvg consistently delivered lower mean absolute percentage error (MAPE) values than FedSGD.

In a different domain, Zhang and Li [15] applied FL to enhance sensor data integration for fault detection in industrial equipment while maintaining privacy. They confirmed FL's value as a privacy-respecting alternative but noted a common simplifying assumption in FL research: that client datasets are independent and identically distributed (IID). Real-world industrial datasets often violate this IID condition—a challenge known as domain shift or data heterogeneity. To mitigate this, the authors introduced a federated transfer learning strategy that shares statistical representations of local data distributions across clients, enabling effective knowledge transfer without exchanging raw sensitive samples.

Beyond energy systems, federated learning finds applications in diverse fields. For instance, Zhang and Li [15] employed it to fuse sensor signals for intelligent machinery diagnostics. In medicine, Boughorbel *et al.* [16] utilized FL to analyze electronic health records, and Thapa *et al.* [17] applied it to strengthen the detection of phishing emails.

Model personalization

Both Taik and Cherkaoui [13] and Sim *et al.* [18] examined model personalization (PS) as a complementary or alternative strategy to federated learning. Personalization involves taking an initial (often globally trained) model and further fine-tuning it exclusively on the data available at a single client or device. These studies independently found that performing local adaptation markedly improves predictive accuracy for that specific user or device, while ensuring that no private data ever leaves the local environment. Moreover, because personalization operates solely on local data without any global aggregation assumptions, it naturally sidesteps the non-IID / domain shift difficulties that commonly hamper standard FL.

Within the electricity sector, Taik and Cherkaoui [13] demonstrated that personalizing models significantly enhances the accuracy of very short-term load predictions compared with both purely centralized (CENT) and standard federated approaches. Likewise, Sim *et al.* [18] successfully applied personalization to develop privacy-preserving, user-specific end-to-end speech recognition systems on smartphones, ensuring that training data remains confined to the device and is never uploaded or stored centrally.

Loss functions

Loss functions serve as differentiable measures of prediction quality, guiding gradient-based optimization. Different loss formulations emphasize different aspects of model performance, ultimately leading to distinct optimal solutions. The most widely used loss functions in load forecasting belong to the family of p-norm errors, which compute some form of average deviation between predicted and actual values—for example, mean absolute error (MAE).

When applied to electricity demand forecasting, however, p-norm losses exhibit a well-known drawback: they are highly sensitive to extreme values and sharp peaks within the time series [19]. This sensitivity arises because large deviations (such as under- or over-predicting a demand spike) incur disproportionately heavy penalties. Additionally, accurate peak prediction requires some tolerance for small temporal misalignments, yet standard p-norm metrics harshly penalize any shift. As a result, models trained with these losses often converge toward smoothed, averaged profiles rather than faithfully reproducing distinct load spikes. Nevertheless, correctly anticipating the occurrence (and approximate timing) of peak demand periods remains essential for secure and efficient grid management.

To overcome these limitations, Haben *et al.* [20] proposed the adjusted error—a tailored loss function designed for noisy, peak-heavy, and temporally variable signals. The adjusted error introduces controlled flexibility by permitting limited temporal shifts (permutations within a small window) when aligning predicted and target sequences. This mechanism reduces the excessive penalty for slightly misaligned peaks, thereby encouraging the model to prioritize detecting peak events over rigid sample-by-sample matching.

Contributions of this paper

The primary goal of this research is to assess the practical effectiveness of federated learning (FL) and model personalization (PS) relative to two established neural network training paradigms: the fully centralized approach (CENT) and a simple pre-trained baseline (BASE). The CENT strategy trains a single model on pooled data from all users. In contrast, the BASE strategy applies an existing, off-the-shelf pre-trained model directly to individual decentralized datasets without further adaptation. A high-level comparison of the operational principles of these four training strategies is depicted in **Figure 3**.

Furthermore, this work seeks to determine, on a statistically robust basis, which training approach is best suited for individual household electricity load forecasting. To enable a fair and unbiased comparison despite substantial inter-user data heterogeneity, the study introduces a new evaluation framework called “Differential Comparison.” This technique isolates the relative performance differences between pairs of models by focusing only on loss deltas computed on identical data samples, thereby minimizing the confounding influence of varying data structures.

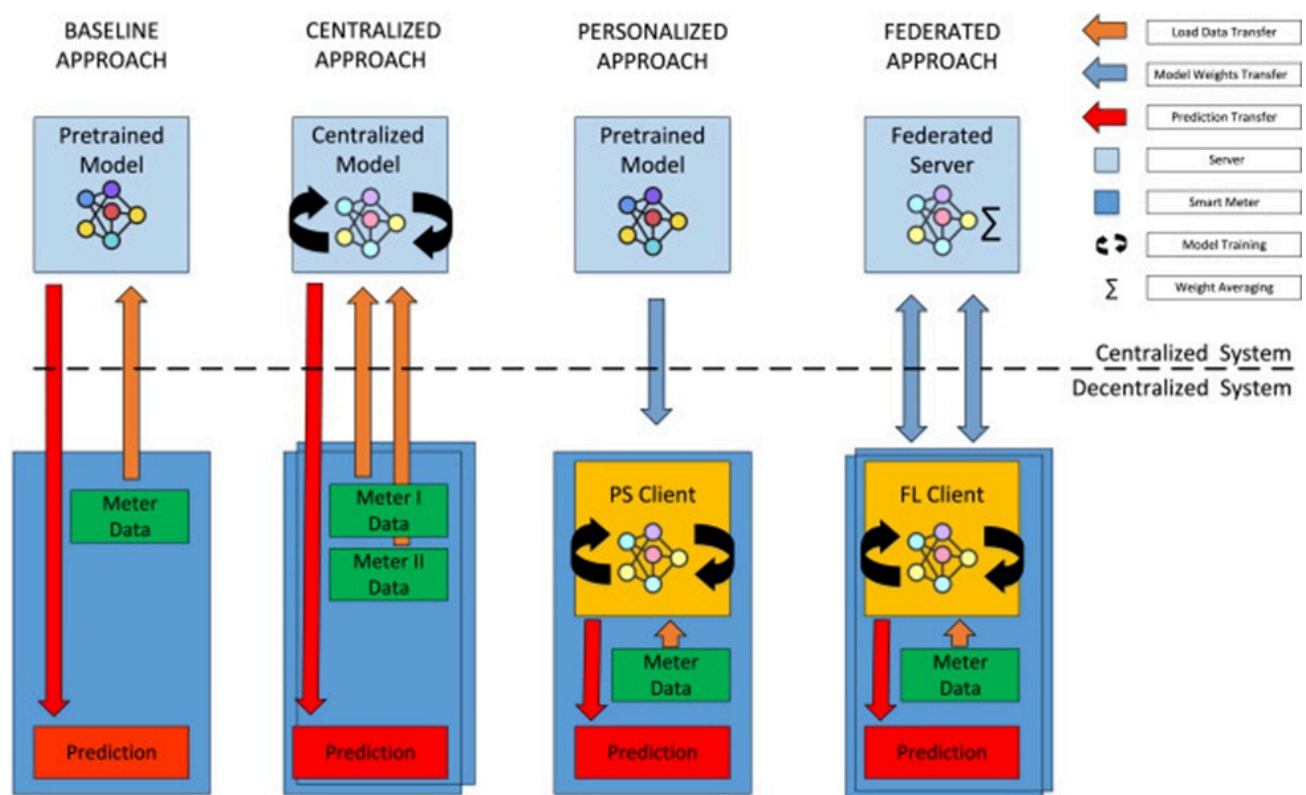


Figure 3. Schematic overview of the operational mechanisms behind the evaluated training strategies (BASE, CENT, PS, FL).

More precisely, the present work makes four primary contributions:

- First, it integrates key strengths from prior load forecasting research presented by Gasparin *et al.* [4], Bai *et al.* [10], and Haben *et al.* [20]. Specifically, the study adopts the neural network design outlined by Gasparin *et al.* [4] and Bai *et al.* [10], while incorporating the “adjusted error” loss function introduced by Haben *et al.* [20] to enhance the temporal convolutional network’s (TCN) ability to capture and predict irregularly occurring demand peaks.
- Second, the paper builds upon and extends the investigations reported by Fekri *et al.* [9] and Taik and Cherkaoui [13] by combining their federated learning frameworks with the advanced neural architecture proposed by Gasparin *et al.* [4] and

Bai et al. [10].

- Third, it performs a large-scale, simulation-driven comparison of federated learning (FL) and model personalization (PS) against the traditional centralized (CENT) and baseline (BASE) training approaches, ensuring conclusions are supported by statistically meaningful evidence.
- Fourth, the work tackles the challenge of fairly evaluating model performance when datasets exhibit substantial structural differences. To achieve this, it proposes “differential comparison”—a more reliable evaluation strategy than simply averaging loss values across samples. The technique isolates performance differences by examining pairwise loss contrasts on matching data instances, thereby removing the confounding influence of varying data characteristics.

Model Training

This section reviews the model training approaches investigated in the current research.

Rationale for adopting advanced training techniques

Training neural networks for load forecasting requires exposure to user-specific consumption records. Conventional approaches typically gather these records from multiple sources, consolidate them in a central location, and train a single shared model. When applied to forecasting at the individual household level, this centralized paradigm presents two significant limitations:

- The first concerns the highly sensitive and personal nature of household load profile data. Transferring such information to a central server creates notable risks to data security and user privacy [13]. Moreover, such data transfers may violate applicable privacy regulations, including the General Data Protection Regulation (GDPR) enforced across the European Union.
- The second stems from the considerable heterogeneity in consumption patterns across individual households. This diversity heightens the risk that a globally trained model will overfit to patterns prevalent in only a portion of the user base. A unified model often struggles to adequately represent the full range of structural variations present in the data [9]. At the same time, differences in magnitude (scale) between households can produce widely varying loss values even for predictions of comparable quality—a phenomenon commonly referred to as the domain shift or non-IID data challenge.

In response to these drawbacks, the present study examines federated learning and model personalization as viable alternatives that can mitigate or eliminate the limitations inherent in standard centralized neural network training.

Alternative I: Federated learning

Federated learning (FL) supports both the ongoing refinement of already-deployed models and the initial training of a shared global model using data that remains distributed across many devices or clients. Crucially, FL achieves this without ever requiring the transmission of raw source data. Instead, each participating client conducts local training on its own dataset and forwards only the resulting model parameter updates (weights) to a central server (Figure 4).

At the server (denoted FL-S), these locally optimized updates from multiple clients (FL-C) are aggregated — typically through simple averaging — to produce an improved version of the global model. The updated global model now incorporates knowledge derived from previously unseen data distributions, while all original consumption records stay local to their respective clients. This process iteratively refines the shared model.

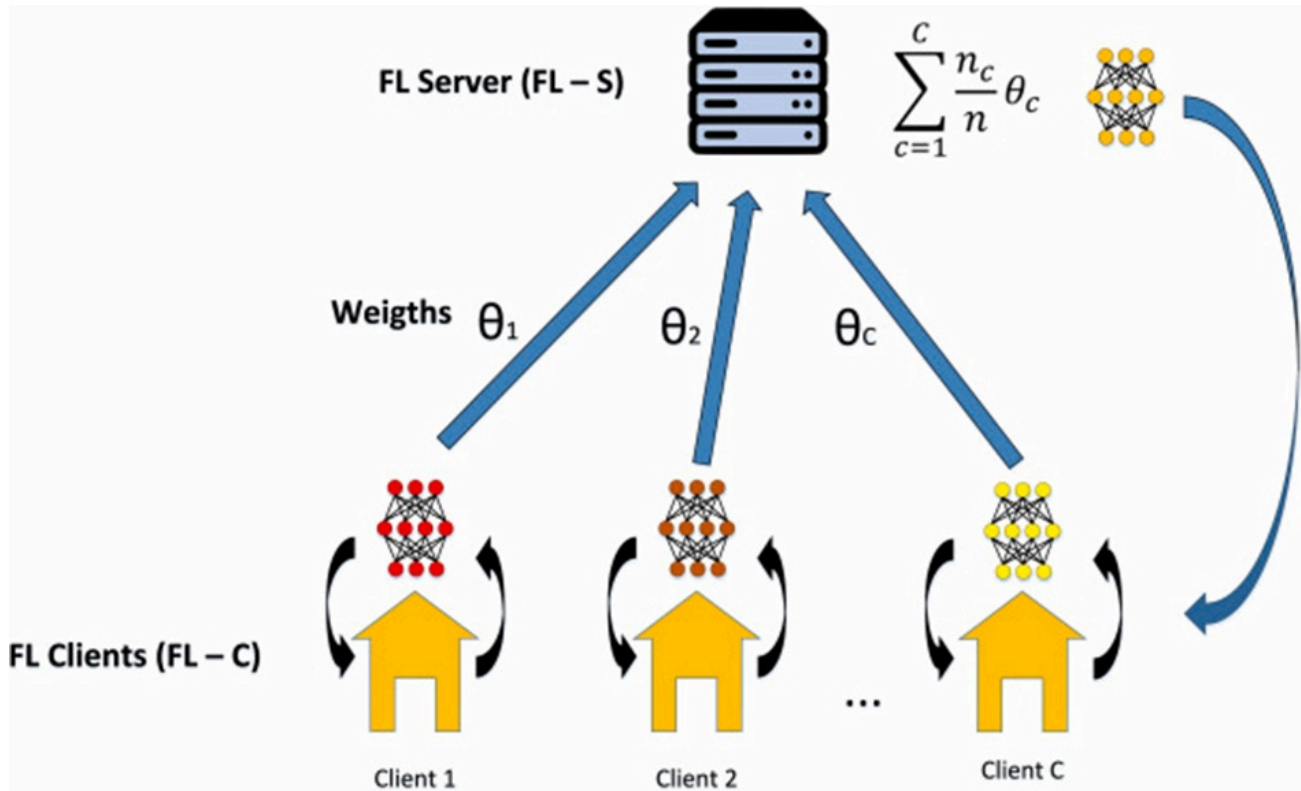


Figure 4. Illustration of the parameter (weight) update mechanism in the federated learning process.

Privacy and security advantages of federated learning

Federated learning enables each participating client to benefit from knowledge patterns learned across the entire user population while ensuring that all raw, private consumption data remains stored exclusively on the local device or at the edge. This represents a fundamental improvement over traditional centralized training, in which sensitive user records must be uploaded to a single server. In real-world deployments, the security of federated learning can be further strengthened by encrypting the exchanged model parameters before transmission [21].

Model aggregation strategies in federated learning

A variety of aggregation techniques have been developed for combining local updates in federated learning systems. Among the most widely adopted are federated averaging (FedAvg) [9] and federated stochastic gradient descent (FedSGD) [9, 22–24].

In FedAvg, the central server constructs an updated global model for communication round r by performing a straightforward element-wise average of the model weights received from all participating clients (Eq. 1).

$$\theta_{r+1}^S \leftarrow \sum_{c=1}^C \frac{n_c}{C} \theta_r^c \quad (1)$$

By contrast, FedSGD collects the gradient updates computed locally on each client, averages those gradients across clients, and applies the averaged gradient to adjust the server-side parameters θ^S . Different implementations of FedSGD exist, but the most prevalent version effectively performs mini-batch stochastic gradient descent. In each iteration, gradients ∇r from a randomly selected subset of clients are computed in parallel and averaged to produce a single update step with learning rate α (Eq. 2).

$$\theta_{r+1}^S \leftarrow \theta_r^S - \alpha * \sum_{i=1}^n \frac{n_i}{n} \nabla_{r,i} \quad (2)$$

Drawing on empirical findings [9, 22–24], the present study adopts the FedAvg aggregation rule for its federated learning implementation. A detailed description of the complete federated learning procedure using FedAvg is provided in pseudocode form in **Algorithm 1**.

r_{max} : max. round number, C : all clients in network, S : FL Server,
 n : number of randomly selected clients per FL round, E : epochs,
 J : error, L_{Adj} : adjusted loss, θ : neural network weights, ∇ : gradient,
 α : learning rate, θ_0 : saved model weights

```

S.init() // Initialize server S
for each client in C do
    client.load_data()
    client.init_model( $\theta_0$ )
    client.connect(S)
end
while round  $r < r_{max}$  rounds do
    Select  $n$  random clients  $C_n$ 
    for  $e$  in  $E$  Epochs do
         $\theta_r^c \leftarrow \theta_r^S$  // Receive server weights
        for client  $c$  in  $C_n$  Clients do
             $X, Y = c.data.get\_batch(e)$ 
             $\hat{Y} = c.model.predict(X)$ 
             $J(\theta_r^c) = L_{Adj}(Y, \hat{Y})$  // Loss function
             $\theta_{r+1}^c = \theta_r^c - \alpha * \nabla(J(\theta_r^c))$  // Back propagation
        end
         $S \leftarrow \theta_{r+1}^c$  // Send weights to server
    end
     $\theta_{r+1}^S = \sum_{c=0}^n \frac{n_c}{n} \theta_{r+1}^c$  // Node wise averaging of all weights of  $n$  client
     $r = r + 1$ 
end

```

Algorithm 1. Federated learning with FedAvg

Federated learning and the issue of domain shift

A key limitation of federated learning is its core assumption that data across all participating clients follow the same independent and identically distributed (i.i.d.) distribution, largely because of the model averaging step in each update round. In practice, however, certain use cases—such as electric load forecasting—often involve clients whose data distributions differ significantly. This mismatch is commonly referred to as the domain shift problem [15].

Approaches based on transfer learning have been applied to mitigate domain shift. Nevertheless, most traditional transfer learning methods require sharing data from the target domain [15], which violates privacy regulations and is therefore unsuitable for applications such as load forecasting, where data protection is critical.

Domain shift also carries another drawback: like fully centralized training, the federated model must accommodate the diverse data patterns across clients, or else overall predictive performance will suffer.

Studies by Taik and Cherkaoui [13] and Savi and Olivadese [25] have demonstrated that federated learning remains viable for short-term electric load forecasting and can achieve accuracy comparable to that of state-of-the-art centralized models. Building on these findings, the present work assumes in all subsequent experiments that sufficient shared patterns exist among client datasets, enabling federated learning to extract meaningful general features.

Alternative II: Model personalization

An alternative to federated learning is personalized training (PS), in which each client develops its own local model. In conventional personalized approaches, training of these individual models typically occurs centrally, requiring substantial computational resources at the central server to handle multiple models simultaneously. Moreover, this setup inherits the same privacy and security vulnerabilities as centralized training, since sensitive user data must still be uploaded.

The motivation for on-device personalized training stems from the increasing computational capabilities of modern smart meters, which now support local model training directly at the edge (Figure 3). This approach eliminates the need to transmit user data, thereby addressing privacy and security concerns while completely bypassing centralized training.

A further advantage is that the learning task becomes simpler: each model only needs to capture the specific characteristics of a single data source. This inherently resolves the domain shift issue encountered in federated learning, as each personalized model can independently converge to the optimum suited to its client's data distribution.

Research by Sim et al. [18] explored personalization to handle heterogeneous data sources, successfully fine-tuning centrally pretrained models on individual user data without requiring ongoing communication with the central server, thereby tailoring the model to each user's unique patterns. Similarly, Taik and Cherkaoui [13] reported superior performance for personalized models relative to standard centralized training.

Consequently, this study examines personalized training as a promising decentralized alternative to federated learning.

Conventional method I: Centralized training

As outlined earlier, centralized training (CENT) involves gathering load data from all sources into a single location to train one unified global model. Because transmitting private user data in this manner is generally restricted without explicit user consent, the CENT approach is included in this work solely as a performance benchmark against which federated learning and personalization methods are compared.

Conventional method II: Pretrained baseline

For additional comparison, this study incorporates a baseline method (BASE) (Figure 3). The BASE model employs the same temporal convolutional network (TCN) architecture as the other approaches but mimics the conventional practice of pretraining on historical data followed by deployment for inference on unseen households. To replicate this scenario, the BASE model is trained exclusively on one subset of households (S_{BASE_TRAIN}) and then tested on a completely separate subset (S_{BASE_VAL}). To ensure fair evaluation, the test subset for BASE matches the evaluation set used across all other methods.

$$S_{BASE_TRAIN} = \{H_1, H_2, \dots, H_n\} \notin S_{BASE_VAL} \quad (3)$$

$$S_{BASE_VAL} = S_{VAL} \quad (4)$$

Comparison of Training Methods

The primary objective of this paper is to evaluate and contrast the training paradigms. A simulation framework was developed to enable controlled, reproducible experiments across varied data sources, facilitating a statistically meaningful comparison of the different approaches.

Simulation preprocessing

Preparation of the simulation involved two main stages:

- Dataset preprocessing , and
- Selection of model hyperparameters .

Data preparation steps

All experiments utilized the publicly available Low Carbon London (LCL) dataset [26], which comprises smart meter electricity readings collected from households in the London region between 2012 and 2014. The dataset includes 30-minute resolution load profiles from 5566 households, with an average recording duration of 1.73 years per household. Several preprocessing operations were performed to minimize overfitting to seasonal patterns, prevent unintended bias toward specific households, and ensure no leakage of test data into training:

- Elimination of duplicate records,
- Forward imputation for isolated missing timestamps,
- Retention of only those households with a continuous 13-month period (365 + 31 days) and at least 99.5% data availability (resulting in 5082 households),
- Sorting of the retained households by their starting date,
- Division into 13 groups of 366 households each, ensuring different annual starting points per group to minimize seasonal bias within any single group,
- Setting aside three groups as a hold-out set for potential future evaluation (leaving 3660 households),
- Partitioning of each remaining household's time series into training, validation, and test portions following the scheme shown in the lower part of Figure 5, allowing a consistent data structure across training, validation, and testing phases, and
- Application of per-household standardization (z-score normalization) to the data.

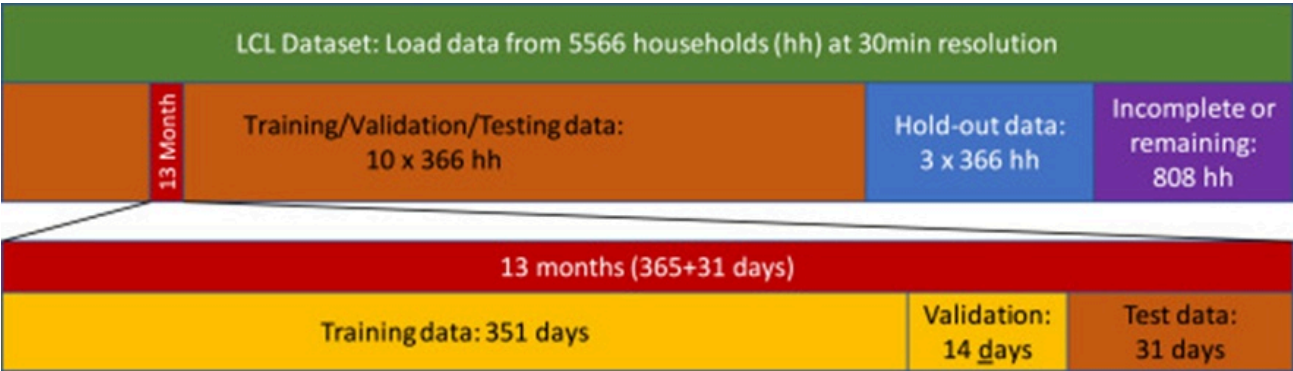


Figure 5. Division of the LCL dataset into training, validation, and test sets.

Hyperparameter optimization

To establish a strong baseline configuration for the simulation, a systematic hyperparameter tuning process was conducted for the Temporal Convolutional Network (TCN) architecture. This tuning procedure utilized Bayesian optimization combined with HyperBand scheduling (TuneBOHB) [27], paired with the associated HyperBandForBOHB scheduler provided by the Ray Tune framework [28].

The objective of the optimization was to minimize the validation loss, measured via mean absolute error (MAE). For each trial in the search process, a different randomly chosen subset of households served as the training and validation data, ensuring diverse sampling across the dataset.

The resulting optimal hyperparameter values, along with their best-performing configurations, are presented in **Table 1**. Except where explicitly noted otherwise, all experiments described in the following sections use the hyperparameter settings identified during this optimization phase.

Table 1. Optimal hyperparameters determined by hyperband search

Parameter	Optimum
Data Input Length I_w	5 days
Prediction Length O_w	1 day
Number of Parallel TCN Filters	64
Number of TCN Layers	8
Dilation of TCN Layers	2
TCN Dropout Rate	0.1
Regularization	L1 0.0001
Normalization Function	per LCL house, standardization scaling
Use Time of Day as Input	No
Use Day of Week as Input	No
Number of Parameters in Model	760,800

Setup of the simulation framework

To properly structure the simulation environment, several additional control parameters were established, including:

- The count of clients trained concurrently under each method,
- The total number of training epochs per run,
- The number of training iterations (steps) within each epoch, and
- The total number of independent experiment repetitions (using varied household groupings).

These parameter choices were deliberately aligned with an upcoming real-world pilot study connected to this research, in which load forecasting will be deployed across 30 residential sites. Accordingly, the following client configurations were adopted for every experimental run:

- 30 Baseline clients (BASE)
- 1 Centralized client (CENT)
- 30 Federated Learning clients (FL-C)
- 1 Federated Learning server (FL-S)
- 30 Personalized clients (PS)

It should be noted that, while 30 clients can be instantiated for both federated learning and personalized training, only one centralized model (CENT) and one federated server (FL-S) are created per run. This is because each FL client and each PS client is trained exclusively on data from a single household, whereas both the centralized model and the federated server aggregate information across all 30 households.

In contrast, although the baseline models are also trained on aggregated data from multiple households, their training sets consist of entirely separate groups of 30 households. This separation ensures that the BASE models receive no exposure to the 30 “new” households used for training and evaluation in the other methods, faithfully replicating a traditional pretrained model scenario with no knowledge of the target users. As a result, it remains feasible to train and subsequently evaluate 30 distinct BASE models, all tested on the identical 30-household evaluation set employed by the FL, PS, and CENT approaches.

The number of training steps per epoch was fixed at 10 to respect the communication constraints inherent to federated learning, which relies on relatively frequent model weight updates across participants.

The total number of epochs was set to 500, determined by observing the convergence curves of the models; all approaches reached stable performance well before the end of a single run.

To generate a sufficiently large sample size for robust statistical comparison of the methods' performance, the entire simulation was repeated 100 times, with each repetition using a freshly randomized household composition.

Execution of the simulation

To ensure fair and directly comparable performance metrics, every method must be assessed on the same set of Low Carbon London (LCL) households. Accordingly, the personalized (PS), federated (FL), and centralized (CENT) approaches are all trained, validated, and tested using the identical group of 30 households. The baseline (BASE) method, however, is trained on a completely disjoint group of 30 households and evaluated only on the shared 30-household test set, thereby mimicking the behavior of a conventionally pretrained model applied to previously unseen customers.

A comprehensive overview of the full simulation workflow is illustrated in **Figure 6**.

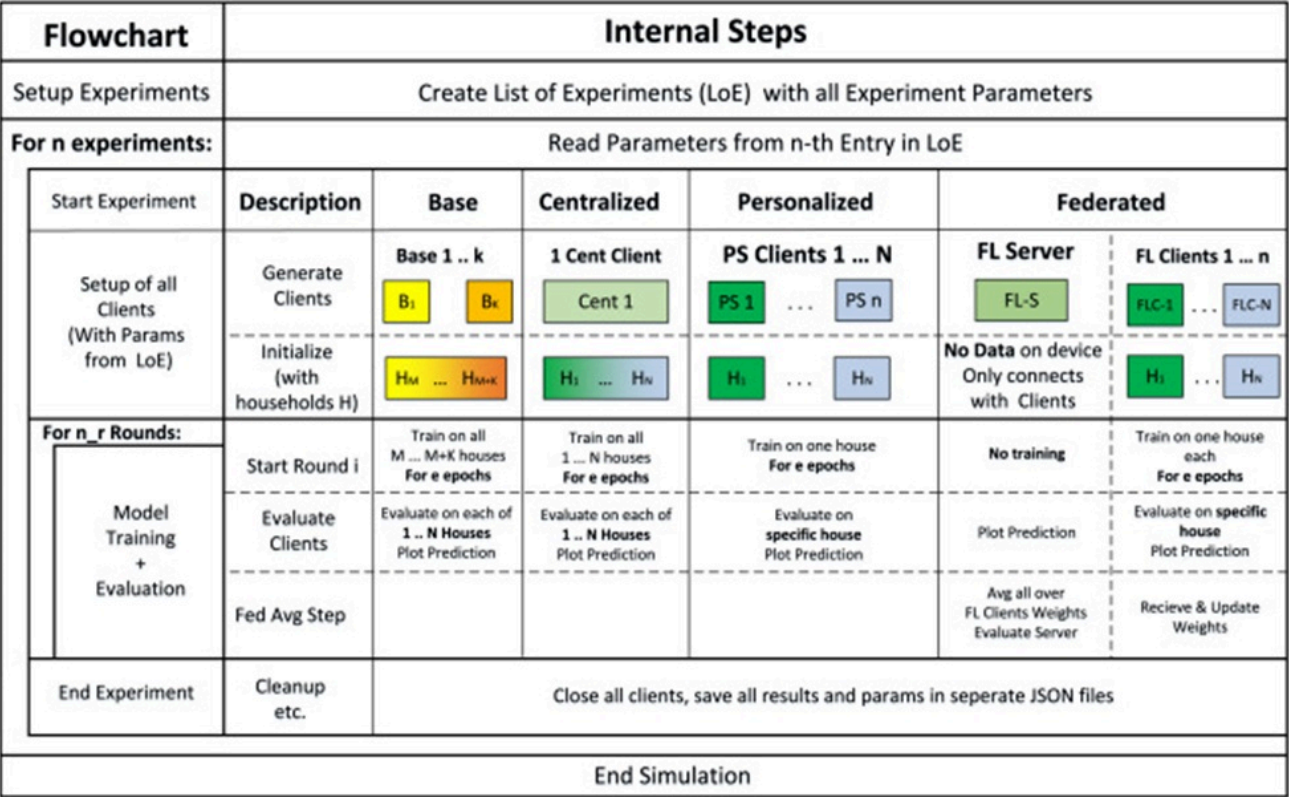


Figure 6. Schematic representation of the simulation workflow employed for comparing the various training strategies.

Formulated hypothesis

A hypothesis regarding the expected performance ordering of the different training approaches was established. This hypothesis assumes that all methods use the same model architecture and therefore have the same number of parameters. Under this condition, it follows that the method required to address the least complex learning task should theoretically achieve the highest performance. With this principle in mind, the anticipated ranking outlined in Table 2 was proposed.

Under the stated assumptions, the personalized training method (PS) was expected to yield the best overall results. The reasoning is that each PS model faces the simplest possible task: it must learn and represent only the consumption behavior of one individual household.

The centralized training method (CENT) was expected to rank second. Although it confronts a more difficult generalization problem than PS—having to model patterns across 30 diverse households—it benefits from unconstrained gradient descent. This allows it to more effectively locate a strong local minimum than the simpler parameter-averaging procedure (FedAvg) used in federated learning, which was therefore expected to produce slightly inferior performance.

It should be noted that although this study distinguishes between federated learning clients (FL-C) and the federated server (FL-S), both were expected to achieve essentially the same performance. This is due to the repeated application of the FedAvg aggregation step, which continuously aligns the parameters of clients and server toward a common solution.

Finally, the baseline pretrained method (BASE) was hypothesized to deliver the weakest performance. Because its training occurs on a completely separate group of households, it never encounters any data from the target evaluation set. It therefore cannot adapt as effectively to the specific characteristics of the test households as the other methods, which are at least partially trained on (or directly exposed to) data from those same households.

Table 2. Hypothesis of performance ranking of training methods.

Training Method	PS	CENT	FL-C	FL-S	BASE
Performance Rank	1	2	3 ^a	3 ^a	4

^a Too similar to be considered different.

Results From the Simulation

This section presents and examines the outcomes obtained from the simulations.

The correct functioning of the simulation framework and all implemented training approaches was confirmed through two main checks: consistent convergence of the loss during training, and visual inspection of predicted load curves against the actual target values (Figure 1).

Preliminary comparison: Impact of the underlying load data

When the absolute error distributions of the various training methods were initially compared, the overall loss values appeared quite comparable across all approaches (Figure 7). Nevertheless, the results also revealed substantial variability in predictive accuracy among the different households included in the analysis. For instance, the adjusted error range for the centralized model spanned 0 to approximately 6, while the corresponding range for the personalized models spanned 0 to approximately 13 (Figure 7).

This observation prompted the formulation of a working hypothesis: the dominant factor driving differences in model performance is likely the intrinsic characteristics of the training data assigned to each household, rather than the specific training methodology employed. Relying solely on the raw performance metrics—which appear to be heavily shaped by data-related factors—could therefore lead to misleading interpretations when attempting to identify the most effective training strategy.

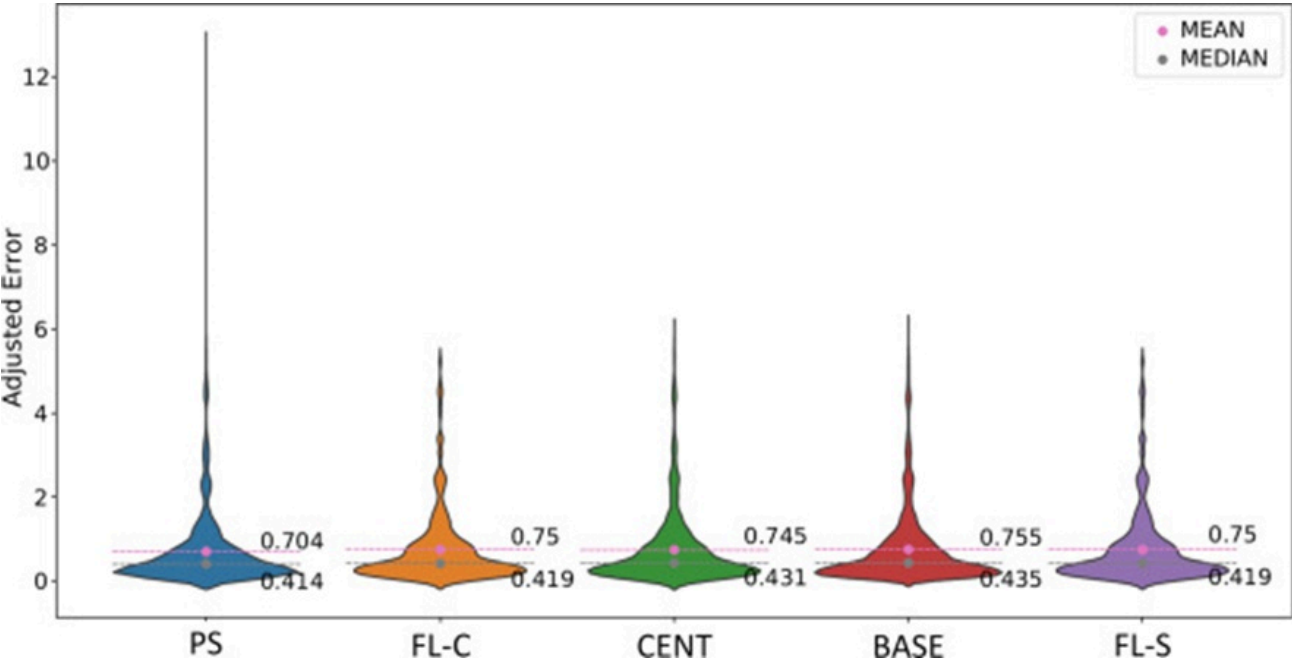


Figure 7. Comparison of the different distributions of test loss of each training method.

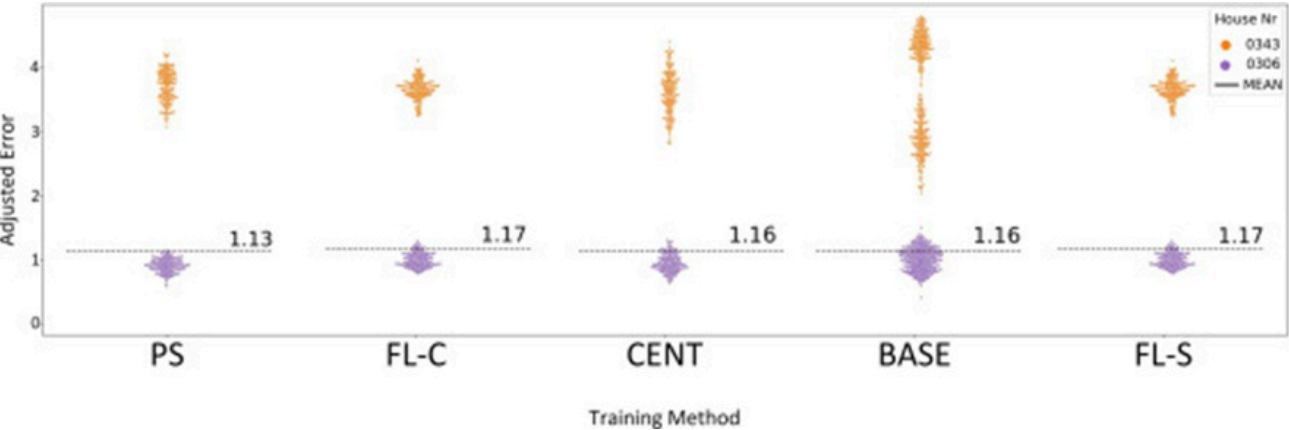


Figure 8. Comparison of the converged test loss of all training methods trained on the same data.

To confirm that variations in performance were primarily driven by differences in the underlying data sources rather than by the training approaches themselves, a more detailed examination was conducted. The findings are illustrated in **Figure 8**.

This visualization displays the loss distributions for a limited subset of households across all training methods, clearly revealing which data source is responsible for each recorded loss value. Each point in the plot represents the test loss of a single model instance. The color of each point indicates the household (data source) it originates from, its horizontal position (X-axis) corresponds to the specific training method, and its vertical position (Y-axis) reflects the magnitude of the loss.

For clarity, only the results from two households are displayed in the main figure. The complete version, including the full small-sample household set, is in **Figure 9**.

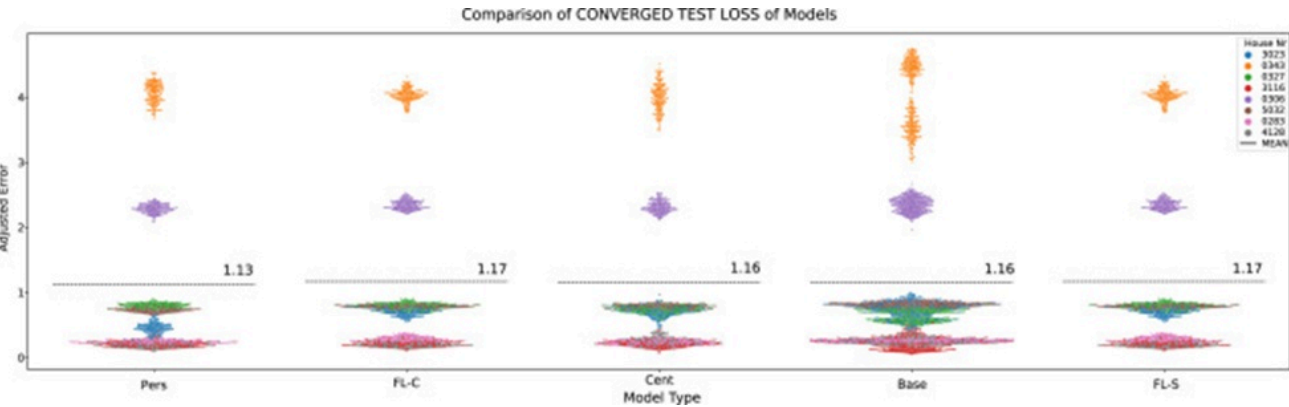


Figure 9. Comparison of all training methods when trained on identical source data.

From **Figure 8**, the following patterns are evident:

- Losses associated with the same individual household form tight clusters within each training method,

- These household-specific clusters remain clearly separated from one another, and
- For any given household, the spread and level of losses remain broadly similar regardless of the training method applied. In contrast, substantial differences in loss values appear between households even when using the same training method.

These observations strongly suggest that the primary source of variation in loss stems from differences among the data sources (households), rather than from meaningful differences in the effectiveness of the training methods. This conclusion is further reinforced by the fact that identical households produce comparable loss levels across all training strategies, whereas switching to a different household within the same method results in large performance shifts.

Consequently, simply comparing the overall loss distributions of the methods effectively amounts to comparing the inherent characteristics of the data distributions across households, rather than providing a fair assessment of which training method performs best. Such direct comparisons of loss distributions are therefore not suitable for determining the superior training approach.

It is important to note that these findings were obtained using only normalized load values. A comparable pattern was also observed when a relative error metric, such as Mean Absolute Percentage Error (MAPE), was applied instead.

Additional support for this interpretation emerges from a review of related prior work. For instance, Fekri *et al.* [9] evaluated federated learning, model personalization (referred to as “Individual LSTM” in their study), and a centralized model, comparing the MAPE values produced by each method (Figure 10). Their results similarly reveal that performance differences across training methods are heavily influenced by the specific household data being modeled. Notably, all methods achieved good accuracy on House 6 but poor performance on House 13, highlighting once again the overriding role of the underlying data source.

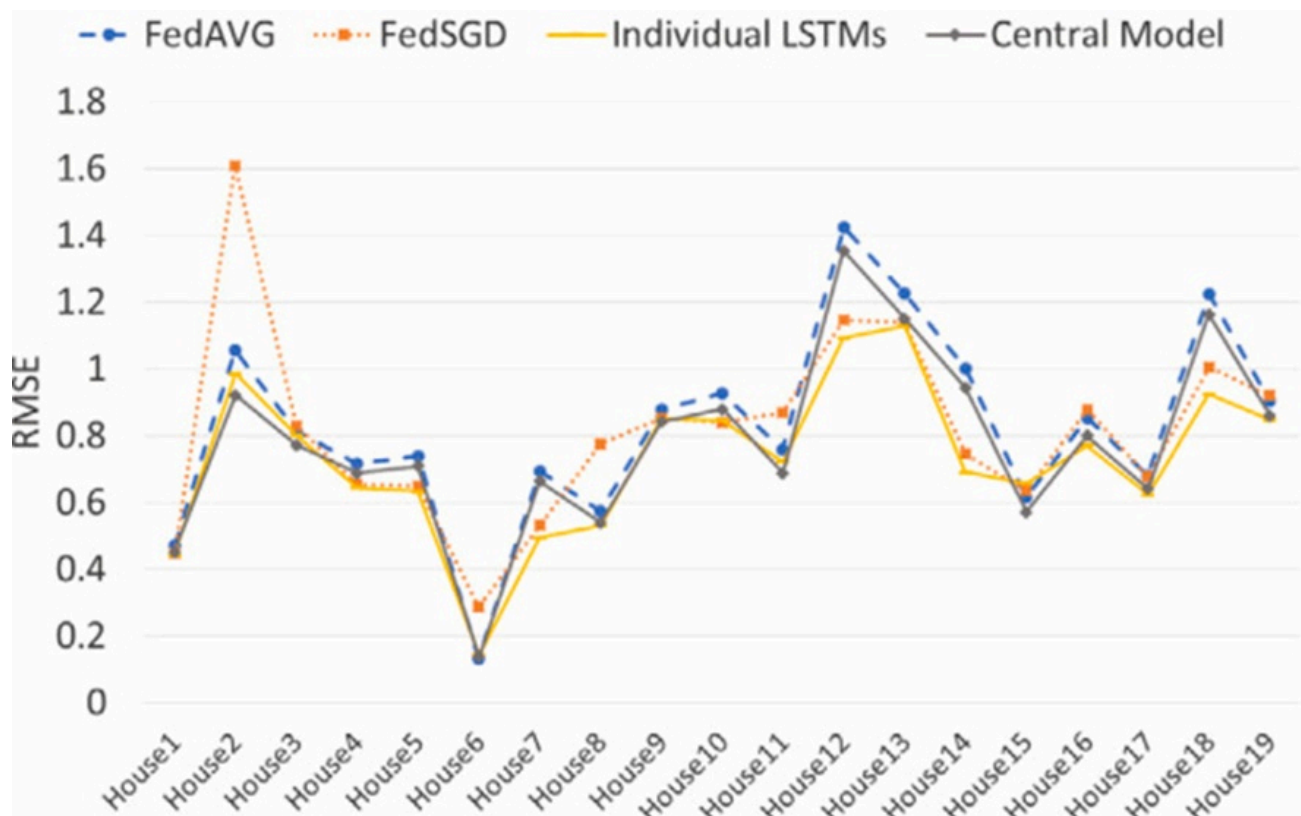


Figure 10. Test loss results of different training methods analyzed by Fekri *et al.* [9].

Mitigating the impact of load data through differential analysis

To fairly assess the relative effectiveness of the forecasting models, it was necessary to develop a strategy that minimized the confounding influence of the underlying data sources. This section outlines several possible approaches to address this issue and describes the method ultimately adopted for the subsequent analyses.

Multiple techniques can help reduce the extent to which differences in data sources affect model evaluation:

- One straightforward option is to apply normalization (or standardization) to both the input features and target values used during training, validation, and testing. While this step helps, it does not fully eliminate the problem, as large fluctuations and structural differences within each household's load profile remain.
- A second approach involves adopting a scale-invariant loss function, such as the Mean Absolute Percentage Error (MAPE), as Fekri et al. [9] did to support comparisons among training strategies. However, MAPE primarily corrects for absolute scale offsets and does not adequately account for internal variability or distributional differences within the data, so the influence of the source data can persist.
- A third possibility is to ensure that all models are trained and evaluated on the same collection of households. This reduces—but does not completely remove—the confounding effect of data variability.

To more effectively neutralize the impact of differing data sources, this work introduces a differential comparison approach. The core idea is to directly examine the pairwise differences in loss values produced by two training methods on the same individual data samples (i.e., the same household time steps).

This technique is particularly well-suited to electric load forecasting because subtracting the losses of two models trained and tested on identical samples effectively cancels out the household-specific characteristics (denoted as data source DS). The resulting difference isolates the contribution attributable to the training method itself rather than to the data (Eqs. 5–8).

$$L_{PS} = L_{DS} + L_{\Delta PS} \quad (5)$$

$$L_{FL-C} = L_{DS} + L_{\Delta FL-C} \quad (6)$$

$$[L_{FL-C} - L_{PS}] = L_{\Delta PS} - L_{\Delta FL-C} \quad (7)$$

Let LPS and LFL represent the losses obtained from the personalized (PS) and federated learning (FL) methods, respectively, on the same sample. The differential loss is then computed as:

$$\left| \frac{L_{\Delta FL-C}}{L_{\Delta PS}} \right| \gg \left| \frac{L_{FL-C}}{L_{PS}} \right| \quad (8)$$

Under the working assumption that $L_{PS} < L_{FL}$, the difference is positive and reflects PS's performance advantage.

By applying this subtraction consistently across all matched samples, a distribution of pairwise performance differences can be constructed (Figure 11), provided both methods were trained and evaluated on the identical set of data points.

To enable a comprehensive and unbiased ranking of all training methods, the following combined strategy was implemented in this study:

- Standardization of both model inputs and outputs (per-household z-score normalization)
- Training, validation, and testing of every model on the same subsets of households
- Pairwise differential comparison of loss values between all training methods for each household and sample (Figure 11)
- Aggregation of the mean differences across all pairwise comparisons and data sources, followed by summation to derive an overall performance ranking (Table 3)

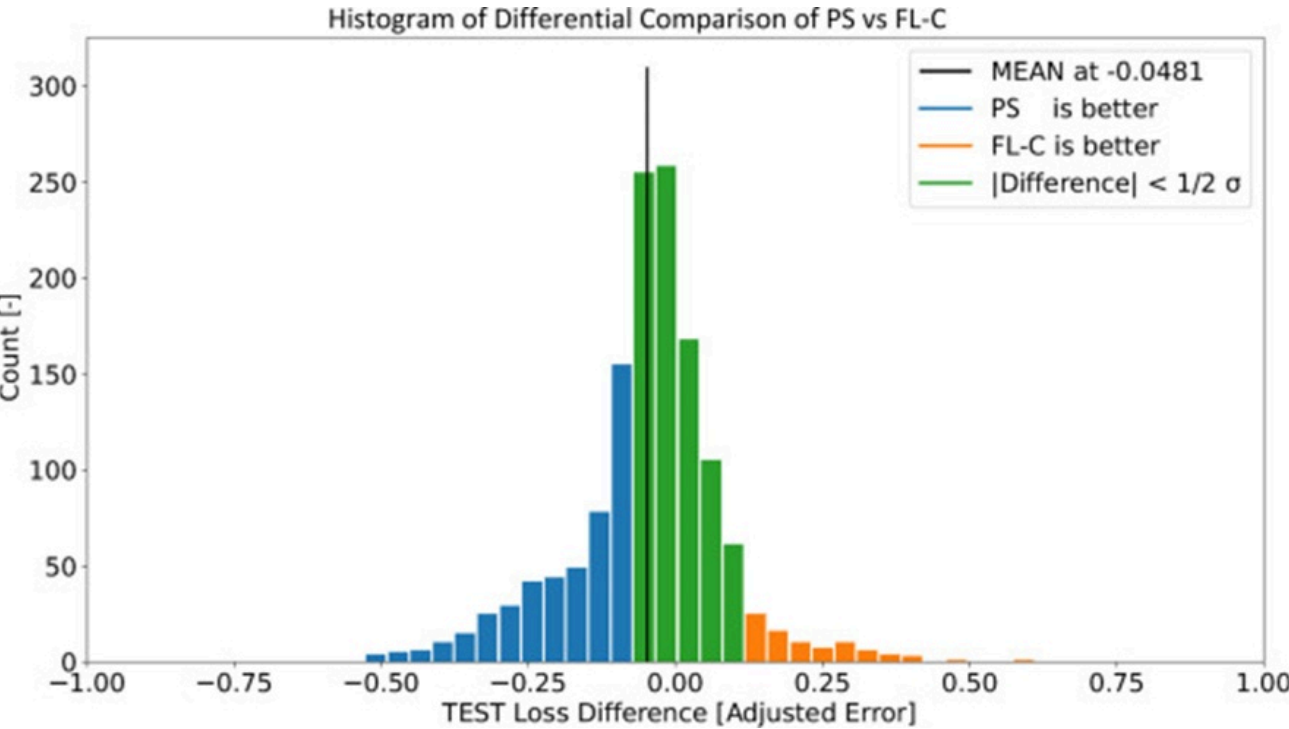


Figure 11. Differential comparison result of FL-C and PS.

Table 3. Mean of the distribution of differences from differential comparisons with adjusted error.

Loss Diff [y - x]	PS	CENT	FL-C	FL-S	BASE
PS	0	0.0421	0.0459	0.0459	0.0512
CENT	-0.0421	0	0.0038	0.0038	0.0091
FL-C	-0.0459	-0.0038	0	0.0001	0.0053
FL-S	-0.0459	-0.0038	0.0001	0	0.0053
BASE	-0.0512	-0.0091	-0.0053	-0.0053	0
Sum	-0.1851	0.0254	0.0443	0.0445	0.0709

Rank	1	2	3 ^a	3 ^a	4
------	---	---	----------------	----------------	---

Negative Loss Difference = Better Performance of the vertical Solution Method.

^a Difference too similar to be considered different ranks.

Statistical evaluation of the methods

The aggregated results presented in **Table 3** confirm that the personalized training approach (PS) achieves the highest overall performance among all evaluated strategies. It is followed in descending order by the centralized method (CENT), then the federated learning client and server models (FL-C and FL-S), with the pretrained baseline method (BASE) exhibiting the lowest performance.

To rigorously assess the reliability of this observed ranking and to formally test the hypothesis, a statistical examination of the differential comparisons was conducted. This analysis quantified how frequently one training method outperformed another across matched samples. For each pair of methods, the number of instances in which one produced a lower loss than the other was tallied. These pairwise counts were then summed (denoted as Sum) for every method, allowing calculation of a relative superiority rate.

In a scenario where all training approaches performed identically, the superiority rate would hover near 50%, and the corresponding Sum value would be close to zero.

The pairwise results summarized in **Table 4** show that, in head-to-head comparisons, the personalized method (PS) outperforms all other approaches in 59.1% of the evaluated cases. Given that the statistical confidence associated with this advantage exceeds the 3σ threshold, the observed performance difference is highly significant.

Based on these findings, this study concludes that model personalization (PS) is the most effective training strategy for electric load forecasting, particularly when dealing with previously unseen households or novel data patterns.

Table 4. Comparison of the superiority count from the differential comparison.

Sum Count [y vs. x]	PS	CENT	FL-C	FL-S	BASE
PS	0	−12344	−18304	−18348	−19444
CENT	12344	0	−8388	−8324	−12264
FL-C	18304	8388	0	200	−5864
FL-S	18348	8324	−200	0	−6180
BASE	19444	12264	5864	6180	0
Sum	68480	16632	−26692	−20292	−43752
Relative Superiority Rate	0.591	0.522	0.472	0.473	0.442
Rank	1	2	3 ^a	3 ^a	4
Hypothesis	1	2	3 ^a	3 ^a	4

Comparisons: Add +1 to count if method is better, −1 if method is worse. ^aDifference too similar to be considered different ranks

In a direct head-to-head comparison among the remaining training approaches, the CENT method outperforms both the FL and BASE methods in 55.1% of cases. This difference is statistically significant, exceeding the 3σ confidence level.

When comparing only the FL and BASE methods, the findings indicate that in 52% of cases, both FL-S and FL-C surpass the BASE method, again with statistical significance greater than 3σ . However, the performance gap between FL-C and FL-S is not large enough to be considered statistically significant.

Summary of results

The key findings from the analysis and performance evaluation of the training methods examined in this study can be summarized in the following four points:

- Recognition that data influence represents a major issue when comparing loss distributions across training methods that rely on different data source sets (with data influence being the primary driver).
- Introduction of a differential comparison approach that isolates the effects attributable solely to the training methods themselves.
- Confirmation that model personalization (PS) emerges as the highest-performing training method among the four methods evaluated.
- Demonstration that both alternative training strategies analyzed (PS and FL) deliver superior performance compared to the pretrained baseline method (BASE).

Conclusion

This study investigated alternative training approaches for neural network-based load forecasting. The research was driven by the hypothesis that distributed training methods improve privacy and security of end-user data by eliminating the need to centralize electric load information. It compared model personalization (PS) and federated learning (FL) with a conventional centralized predictor and a pretrained baseline, using a comprehensive simulation framework. In the simulation, a Temporal Convolutional Network was trained on individual household data and fine-tuned with a modified error loss function for short-term electric load forecasting.

The analysis revealed that the unique characteristics of load curve data—particularly the statistical and relative distribution of peak loads, along with differences in scale across samples—dominate performance comparisons in multi-client datasets. If overlooked, this can result in an unintended comparison of data characteristics rather than true model performance. To address this, the study introduces a “differential comparison” technique that isolates performance variations to the training method or model architecture alone. This is accomplished by comparing loss differences exclusively on the same data samples across all models, thereby removing offsets caused by variations in data structure.

Applying a differential comparison, the study found that PS is the best-performing method among those evaluated, achieving superiority in 59.1% of cases, with a significance level exceeding 3σ . Furthermore, all four training methods exhibit statistically significant performance differences (significance $> 3\sigma$), and their rankings are presented in [Table 5](#).

Since model personalization (PS) demonstrated the strongest performance among the evaluated approaches, aligns well with data privacy regulations, and avoids the challenges associated with domain shift, this study recommends model personalization as the preferred training method for short-term electric load forecasting.

Table 5. Performance ranking of all analyzed training methods.

Training method	PS	CENT	FL	BASE
Rank	1	2	3	4

Transfer learning as a potential future enhancement for federated learning

Federated learning performs best when client data is independent and identically distributed (IID), yet this assumption rarely holds in electric load forecasting applications. A widely adopted solution in the literature to mitigate non-IID data challenges is transfer learning. However, traditional transfer learning approaches typically require sharing target values (labels) between clients, which conflicts with strict data privacy regulations commonly applied to electric load data. For this reason, transfer learning was not explored in the present study.

Zhang and Li [15] introduced an innovative transfer learning technique specifically designed to preserve privacy by prohibiting the exchange of raw client data. Their work demonstrated encouraging outcomes when applied to federated transfer learning scenarios in industrial machine maintenance tasks.

The authors recognize that re-running the federated learning experiments while incorporating this privacy-preserving transfer learning method might yield improved performance for the FL approaches. Nevertheless, because model personalization (PS) is inherently immune to domain shift issues, this study continues to recommend it as the preferred training method for short-term electric load forecasting.

Abbreviations

- BASE = Baseline method
- CENT = Centralized method
- DT = Decision tree
- FCNN = Fully connected neural network
- FL = Federated learning
- FedAvg = Federated averaging
- FedSGD = Federated stochastic gradient descent
- GDPR = General Data Protection Regulations
- LCL = Low Carbon London
- LSTM = Long Short-Term Memory Network
- MAPE = Mean absolute percentage error
- PS = Personalized method
- RF = Random forest
- TCN = Temporal convolutional network
- SOTA = State of the art

SVM = Support vector machine

Acknowledgements

The writing, review, and editing process was further supported by Raimundo Daniel and Harris Josephine, two research associates of the Lucerne University of Applied Sciences and Arts (HSLU).

Conflict of interest

The authors declare the following financial interests/personal relationships, which may be considered as potential competing interests: Antonios Papaemmanouil reports that financial support was provided by the Swiss Federation of Energy (SFOE).

Financial support

This research was supported by the Swiss Federation of Energy (SFOE) [grant number [SI/502076](#)] for the project: “KnowLEDGE – Decentralized, secure and privacy-protecting AI to improve grid reliability, resilience and cost performance for DSOs”.

Ethics statement

None

Received: 15 Sep 2024

Revised: 22 Oct 2024

Accepted: 01 Dec 2024

Published online: 20 March 2025

Rights and permissions

Open Access The author(s) retain copyright. This article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. It may be shared and adapted for non-commercial purposes with appropriate attribution, an indication of changes, and distribution of adaptations under the same license. Third-party material may be subject to separate terms identified in its credit line. View the license at <https://creativecommons.org/licenses/by-nc-sa/4.0/>.

References

- European Commission, Directorate-General for Energy, Tounquet F, Linden M, Mandatova P, Knapp J, et al. Supporting country fiches accompanying the report Benchmarking smart metering deployment in the EU-28. Publications Office; 2020. <https://doi.org/10.2833/728829>.
- Alfares HK, Nazeeruddin M. Electric load forecasting: literature survey and classification of methods. *Int J Syst Sci*. 2002;33(1):23-34. <https://doi.org/10.1080/00207720110067421>.
- Pití A, Verticale G, Rottondi C, Capone A, Lo Schiavo L. The role of smart meters in enabling real-time energy services for households: The Italian case. *Energies*. 2017;10(2). <https://doi.org/10.3390/en10020199>.

Gasparin A, Lukovic S, Alippi C. Deep learning for time series forecasting: The electric load case. *arXiv:1907.09207*. 2019.

Wang Y, Chen Q, Hong T, Kang C. Review of smart meter data analytics: applications, methodologies, and challenges. *IEEE Trans Smart Grid*. 2018;10(3):3125-48.

Hahn H, Meyer-Nieberg S, Pickl S. Electric load forecasting methods: Tools for decision making. *Eur J Oper Res*. 2009;199(3):902-7.

Gonzalez-Romera E, Jaramillo-Moran MA, Carmona-Fernandez D. Monthly electric energy demand forecasting based on trend extraction. *IEEE Trans Power Syst*. 2006;21(4):1946-53.

Lee D, Hess DJ. Data privacy and residential smart meters: comparative analysis and harmonization potential. *Util Policy*. 2021;70:101188.
<https://doi.org/10.1016/j.jup.2021.101188>.

Fekri MN, Grolinger K, Mir S. Distributed load forecasting using smart meter data: Federated learning with recurrent neural networks. *Int J Electr Power Energy Syst*. 2022;137:107669.

Bai S, Kolter JZ, Koltun V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv:1803.01271*. 2018.

Kuo P-H, Huang C-J. A high precision artificial neural networks model for short-term energy load forecasting. *Energies*. 2018;11(1).
<https://doi.org/10.3390/en11010213>.

Remy P. Temporal convolutional networks for Keras. GitHub Repository. 2020. <https://github.com/philipperemy/keras-tcn>.

Taik A, Cherkaoui S. Electrical load forecasting using edge computing and federated learning. In: *ICC 2020-2020 IEEE International Conference on Communications*. IEEE; 2020. p. 1-6.

Pecan Street Inc. Pecan Street dataset. Dataport. 2019. <https://www.pecanstreet.org/dataport/>.

Zhang W, Li X. Data privacy preserving federated transfer learning in machinery fault diagnostics using prior distributions. *Struct Health Monit*. 2022;21(4):1329-44.

Boughorbel S, Jarray F, Venugopal N, Moosa S, Elhadi H, Makhlof M. Federated uncertainty-aware learning for distributed hospital EHR data. *arXiv:1910.12191*. 2019.

Thapa C, Tang JW, Abuadbbba A, Gao Y, Camtepe S, Nepal S, et al. Evaluation of federated learning in phishing email detection. *arXiv:2007.13300*. 2020.

Sim KC, Zadrazil P, Beaufays F. An investigation into on-device personalization of end-to-end automatic speech recognition models. *arXiv:1909.06678*. 2019.

Moreno JJM, Pol AP, Abad AS, Blasco BC. Using the R-MAPE index as a resistant measure of forecast accuracy. *Psicothema*. 2013;25(4):500-6.

Haben S, Ward J, Greetham DV, Singleton C, Grindrod P. A new error measure for forecasts of household-level, high resolution electrical energy consumption. *Int J Forecast*. 2014;30(2):246-56.
<https://doi.org/10.1016/j.ijforecast.2013.08.002>.

Aono Y, Hayashi T, Wang L, Moriai S. Privacy-preserving deep learning via additively homomorphic encryption. *IEEE Trans Inf Forensics Secur*. 2017;13(5):1333-45.

Wang J, Wang S, Chen R-R, Ji M. Local averaging helps: Hierarchical federated learning and convergence analysis. *arXiv:2010.12998*. 2020.

Lian X, Zhang C, Zhang H, Hsieh C-J, Zhang W, Liu J. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, et al., editors. *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc.; 2017. p. 1-13.

Chen J, Pan X, Monga R, Bengio S, Jozefowicz R. Revisiting distributed synchronous SGD. *arXiv:1604.00981*. 2016.

Savi M, Olivadese F. Short-term energy consumption forecasting at the edge: A federated learning approach. *IEEE Access*. 2021;9:95949-69.
<https://doi.org/10.1109/ACCESS.2021.3094089>.

UKPower Networks. SmartMeter energy consumption data in London households. London Datastore. 2014.
<https://data.london.gov.uk/dataset/smartmeter-energy-use-data-in-london-households>.

Li L, Jamieson K, DeSalvo G, Rostamizadeh A, Talwalkar A. Hyperband: A novel bandit-based approach to hyperparameter optimization. *J Mach Learn Res*. 2018;18(185):1-52. <http://jmlr.org/papers/v18/16-558.html>.

Liaw R, Liang E, Nishihara R, Moritz P, Gonzalez JE, Stoica I. Tune: A research platform for distributed model selection and training. *arXiv:1807.05118*. 2018.